

Design of Computational Intelligence based Image Restoration and Classification Algorithms

**Thesis submitted in partial fulfillment of the
requirement**

for the degree of

Doctor of Philosophy (Ph.D.)

in

Engineering

By

Sriparna Banerjee

Department of Electronics and Telecommunication

Engineering

Jadavpur University, Kolkata

India

2021

JADAVPUR UNIVERSITY

KOKATA – 700032, INDIA

INDEX NO. 283/18/E

1. **Name of the Thesis:** Design of Computational Intelligence based Image Restoration and Classification Algorithms

2. **Name, Designation & Institution of the Supervisor:**

Dr. Sheli Sinha Chaudhuri, Professor, Dept. of Electronics and Telecommunication Engineering, Jadavpur University, Kolkata – 700032, West Bengal, India

3. **List of Publications:**

Conferences:

[1]. **Sriparna Banerjee**, Pranay Kumar Singha, Pritam Kumar Ghosh and Sheli Sinha Chaudhuri, “Fuzzy logic based image dehazing system”, IEEE WIECON 2021, Dhaka, Bangladesh, 4-5 December, 2021.

[2]. Madhusri Maity, **Sriparna Banerjee** and Sheli Sinha Chaudhuri, “Faster R-CNN and YOLO based Vehicle detection: A Survey”, 5th IEEE International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 8-10 April, 2021.

[3]. **Sriparna Banerjee**, Pritam Kumar Ghosh, Pranay Kumar Singha and Sheli Sinha Chaudhuri, “Relevance of Color spaces and Color channels in performing Image dehazing”, IEEE International Conference on Innovations in Energy Management and Renewable Resources, Kolkata, India, 5-7 February, 2021.

[4]. Gourab Adhikari, Sourav Halder, **Sriparna Banerjee** and Sheli Sinha Chaudhuri, “An Effective Use of Synthetic Aperture Radar Imaging Technique Over Optical Remote Sensing and Real Aperture Radar for Mapping of Earth Surface Features”, IEEE International Conference on Innovations in Energy Management and Renewable Resources, Kolkata, India, 5-7 February, 2021.

[5]. Gourab Adhikari, Sourav Halder, **Sriparna Banerjee** and Sheli Sinha Chaudhuri, “An Edge Aware Polarimetric Co-Variance Feature Matrix Generation of Geo-Coded and Quad-Polarized Single Look Complex Synthetic Aperture Radar Data”, IEEE International Conference on Innovations in Energy Management and Renewable Resources, Kolkata, India, 5-7 February, 2021.

[6]. **Sriparna Banerjee**, Sheli Sinha Chaudhuri, Raghav Mehra and Arundhati Misra, “A Comprehensive Survey on Frost Filter and its Proposed Variants”, 5th IEEE International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, 10-12 June, 2020.

[7]. **Sriparna Banerjee**, Sheli Sinha Chaudhuri, Raghav Mehra and Arundhati Misra, “A Survey on Lee Filter and Its Improved Variants”, 6th International Conference on Opto-Electronics and Applied Optics (Optronix), Springer, Kolkata, India, 8-10 June, 2020.

[8]. **Sriparna Banerjee**, Sheli Sinha Chaudhuri, Raghav Mehra and Arundhati Misra, “A Comprehensive Review of Median Filter, Hybrid Median Filter and Their Proposed Variants”, 6th International Conference on Opto-Electronics and Applied Optics (Optronix), Springer, Kolkata, India, 8-10 June, 2020.

[9]. **Sriparna Banerjee**, Shambhab Chaki, Soham Jana and Sheli Sinha Chaudhuri, “Fuzzy Logic- Refined Color Channel Synergism based Image Dehazing”, IEEE Region 10 Symposium Tensymp 2020, Dhaka, Bangladesh, 5-7 June, 2020.

[10]. **Sriparna Banerjee** and Sheli Sinha Chaudhuri, “S-HAZE: Dataset Consisting of Real World Ground Truth and Hazy Images of Varying Haze Density with No-Sky, Little Sky and Large Sky Regions”, IEEE Region 10 Symposium Tensymp 2019, Kolkata, India, 7-9 June, 2019.

[11]. Shibarjun Mandal, Mou Adhikari, **Sriparna Banerjee** and Sheli Sinha Chaudhuri, “Novel Adaptive Statistical Method-CNN Synergism Based Two-Step WCE Image Segmentation”, IEEE International Conference on Intelligent Computing and Control Systems, Chennai, India, 15-17 May, 2019.

[12]. Swati Chowdhuri, **Sriparna Banerjee** and Sheli Sinha Chaudhuri, “Implementation of a novel Minimum Energy Consume Secure Routing Protocol for reliable and energy efficient rescue operation of FANET”, IEEE Conference on Applied Signal Processing Conference (ASPCON), Kolkata, India, 7-9 December, 2018.

[13]. Shibarjun Mandal, Mou Adhikari, **Sriparna Banerjee** and Sheli Sinha Chaudhuri, “Survey on poyp and ulcer detection techniques in WCE images”, IEEE International Conference on Intelligent Computing and Sustainable System, Coimbatore, India, 20-21, September, 2018.

[14]. **Sriparna Banerjee**, Sheli Sinha Chaudhuri and Sangita Roy, “Fuzzy Logic and Log-Sigmoid Function Based Vision Enhancement of Hazy Images”, IEEE IntelliSys, London, United Kingdom, 6-7 September, 2018.

[15]. **Sriparna Banerjee** and Sheli Sinha Chaudhuri, “A Review on various Speckle Filters used for despeckling SAR images”, 2nd IEEE International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 15-16 February, 2018.

[16]. **Sriparna Banerjee** and Sheli Sinha Chaudhuri, “Log sigmoid function based patch independent image haze removal method”, International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 4-6 January, 2018.

[17]. **Sriparna Banerjee**, Sangita Roy and Sheli Sinha Chaudhuri, “Fuzzy logic based vision enhancement using sigmoid function”, IEEE Calcutta Conference (CALCON), Kolkata, India, 2-3 December, 2017.

[18]. **Sriparna Banerjee**, Sangita Roy and Sheli Sinha Chaudhuri, “Effect of patch size and haziness factor on visibility using DCP”, 2nd International Conference for Convergence in Technology (I2CT), Mumbai, India, 7-9 April, 2017.

Journals:

[1]. **Sriparna Banerjee** and Sheli Sinha Chaudhuri, “Bacterial Foraging-Fuzzy synergism based Image Dehazing”, Multimedia Tools and Applications, vol.80(11), 2021, pp.8377-8421.

[2]. **Sriparna Banerjee** and Sheli Sinha Chaudhuri, “Nighttime Image-Dehazing: A Review and Quantitative Benchmarking”, Archives of Computational Methods in Engineering, vol.28, 2021. pp. 2943–2975.

[3]. **Sriparna Banerjee** and Sheli Sinha Chaudhuri, “Total Contribution Score and Fuzzy Entropy based two-stage selection of FC, ReLU and InverseReLU features of multiple

Convolution Neural Networks for erythrocytes detection”, IET Computer Vision, vol. 13 (7), 2019, pp. 640-650.

3. List of Patents/Copyrights:

[1]. **Sriparna Banerjee**, Sourav Halder, Gourab Adhikari and Sheli Sinha Chaudhuri, “Sky Detection using Intensity based Fuzzy Controller,” Diary Number: 9016/2021-CO/SW. Registration Number: SW-14661/2021. (Registration Date: 21 June, 2021)

[2]. **Sriparna Banerjee**, Gourab Adhikari and Sheli Sinha Chaudhuri, “Degree of Purity & Scattering Diversity based Advanced Lee filter ,” Diary Number: 9006/2021-CO/SW. Registration Number: SW-14660/2021. (Registration Date: 21 June, 2021)

[3]. **Sriparna Banerjee**, Pranay Kumar Singha, Pritam Kumar Ghosh and Sheli Sinha Chaudhuri, “Daytime and Nighttime Dehazing Benchmarking Database”, Diary Number: 8997/2021-CO/L. Registration Number: L-103093/2021. (Registration Date: 11 May, 2021)

4. List of Presentations in National/ International Conferences/Workshops:

[1] Madhusri Maity, **Sriparna Banerjee** and Sheli Sinha Chaudhuri, “Faster R-CNN and YOLO based Vehicle detection: A Survey”, 5th IEEE International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 8-10 April,2021.

[2] **Sriparna Banerjee**, Sheli Sinha Chaudhuri, Raghav Mehra and Arundhati Misra, “A Comprehensive Survey on Frost Filter and its Proposed Variants”, 5th IEEE International Conference on Communication and Electronics Systems (ICES), Coimbatore, India,10-12 June, 2020.

[3] **Sriparna Banerjee**, Sheli Sinha Chaudhuri, Raghav Mehra and Arundhati Misra, “A Survey on Lee Filter and Its Improved Variants”, 6th International Conference on Opto-Electronics and Applied Optics (Optronix), Springer, Kolkata, India,8-10 June,2020.

[4] **Sriparna Banerjee**, Sheli Sinha Chaudhuri, Raghav Mehra and Arundhati Misra, “A Comprehensive Review of Median Filter, Hybrid Median Filter and Their Proposed Variants”, 6th International Conference on Opto-Electronics and Applied Optics (Optronix), Springer, Kolkata, India,8-10 June,2020.

[5] **Sriparna Banerjee**, Shambhab Chaki, Soham Jana and Sheli Sinha Chaudhuri, “Fuzzy Logic-Refined Color Channel Synergism based Image Dehazing”, IEEE Region 10 Symposium Tensymp 2020, Dhaka, Bangladesh, 5-7 June, 2020.

[6] **Sriparna Banerjee** and Sheli Sinha Chaudhuri, “S-HAZE: Dataset Consisting of Real World Ground Truth and Hazy Images of Varying Haze Density with No-Sky, Little Sky and Large Sky Regions”, IEEE Region 10 Symposium Tensymp 2019, Kolkata, India, 7-9 June, 2019.

[7] **Sriparna Banerjee** and Sheli Sinha Chaudhuri, “A Review on various Speckle Filters used for despeckling SAR images”, 2nd IEEE International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 15-16 February, 2018.

[8] **Sriparna Banerjee** and Sheli Sinha Chaudhuri, “Log sigmoid function based patch independent image haze removal method”, International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 4-6 January, 2018.

[9] **Sriparna Banerjee**, Sangita Roy and Sheli Sinha Chaudhuri, “Fuzzy logic based vision enhancement using sigmoid function”, IEEE Calcutta Conference (CALCON), Kolkata, India, 2-3 December, 2017.

[10] **Sriparna Banerjee**, Sangita Roy and Sheli Sinha Chaudhuri, “Effect of patch size and haziness factor on visibility using DCP”, 2nd International Conference for Convergence in Technology (I2CT), Mumbai, India, 7-9 April, 2017.

STATEMENT OF ORIGINALITY

I, **Sriparna Banerjee**, registered on 25/05/2018 do hereby declare that this thesis entitled, “**Design of Computational Intelligence based Image Restoration and Classification Algorithms**” contains literature survey and original research work done by the undersigned candidate as a part of doctoral studies.

All the information in this thesis are obtained and presented in accordance with existing academic rules and ethical conduct. I declare that, as required by these rules and conduct and ethical conduct, I have fully cited and referred all materials and results that are not original to this work.

I also declare that I have checked this thesis as per the “Policy on Anti-plagiarism, Jadavpur University,2019”, and the level of similarity as checked by iThenticate software is 1%.

Sriparna Banerjee

Signature of Candidate

Date:27/09/2021

Certified by Supervisor:



Dr. Sheli Sinha Chaudhuri

Professor

Jadavpur University, Kolkata-700032

CERTIFICATE FROM THE SUPERVISOR

This is to certify that the thesis entitled, “**Design of Computational Intelligence based Image Restoration and Classification Algorithms**” submitted by **Sriparna Banerjee** , who got her name registered on 25/05/2018 for the award of Ph.D (Engg.) degree of Jadavpur University is absolutely based on her work under the supervision of **Prof. Sheli Sinha Chaudhuri**, Dept. Of Electronics & Telecommunication Engineering, Jadavpur University, Kolkata-700032, West Bengal, India, and that neither her thesis nor any part of the thesis has been submitted for any degree/diploma or any other academic degree anywhere before.

27/09/2021 ,Kolkata



Dr. Sheli Sinha Chaudhuri

Professor, Dept. of ETCE

Jadavpur University, Kolkata-700032

Acknowledgement

This section of the thesis is very much special for any student as while writing this section he/she gets an opportunity to remember the entire journey once again.

With the blessings of **goddesses Mahakali** and constant encouragement and support of my father **Mr. Biswajit Banerjee**, my mother **Mrs. Lipika Banerjee**, my Ph.D journey is fulfilled with numerous good memories which I like to cherish all throughout my life.

I am really grateful to my supervisor, **Prof. Sheli Sinha Chaudhuri** for giving me the opportunity to pursue Ph.D under her supervision and help me to fulfill not only my dream but my parents dream too.

I would like to express my heartiest gratitude to **Prof. Sudipta Chattopadhyay**, Professor, Electronics & Telecommunication Engineering Department, Jadavpur University, Kolkata 700032 and **Dr.Sayan Chatterjee**, Associate Professor, Electronics & Telecommunication Engineering Department, Jadavpur University, Kolkata 700032 for giving me valuable advice during RAC meetings and thus help me to improve my work.

I am grateful to all my co-authors for their contribution in my work.

Sriparna Banerjee

SRIPARNA BANERJEE

Date: 27/09/2021

Contents

Abstract	1-3
Chapter 1	4-32
1.1. Daytime and Nighttime image dehazing	4-9
<i>1.1.1. Daytime and Nighttime images' properties</i>	6-7
<i>1.1.1.1. Sky Vs Non-sky regions of Daytime images</i>	7
<i>1.1.1.2. Sky Vs Non-sky regions of Nighttime images</i>	7
<i>1.1.2. Significance of image dehazing problem in real world</i>	8-9
1.2. Detection of structurally variant erythrocytes	9-13
<i>1.2.1. Brief introduction on blood components</i>	9-10
<i>1.2.1.1. Erythrocytes/ red blood cells</i>	9
<i>1.2.1.2. Abnormality detection in blood components</i>	9-10
<i>1.2.1.3. Basic functions of blood components</i>	10
<i>1.2.1.4. Poikilocytes</i>	10
<i>1.2.2. Significance of poikilocytes detection in real world</i>	10-13
1.3. Land cover classification using full-polarimetric image data	13- 16
<i>1.3.1. Land cover</i>	13
<i>1.3.2. Synthetic Aperture Radar</i>	13
<i>1.3.3. Advantages of SAR over conventional radars with physical antenna</i>	13-14
<i>1.3.4. Characteristics of images generated by SAR</i>	14-15
<i>1.3.5. Significance of land cover classification in real world</i>	16
1.4. Introduction to the principles of Computational Intelligence	16-33
<i>1.4.1. Computational Intelligence</i>	16
<i>1.4.2. Principles of Computational Intelligence</i>	17
<i>1.4.3. Basics of Conventional Logic and Fuzzy Logic</i>	17-23
<i>1.4.3.1. Membership functions used for representing Fuzzy sets</i>	18-20
<i>1.4.3.2. Fuzzy sets</i>	20
<i>1.4.3.2.1. Representation of Fuzzy linguistic variable</i>	20
<i>1.4.3.3. Intersection and Union operations on Fuzzy sets</i>	21
<i>1.4.3.3.1. Intersection (Fuzzy T-Norm) operation</i>	21
<i>1.4.3.3.2. Union (Fuzzy S-Norm) operation</i>	21
<i>1.4.3.4. Fuzzy Control Systems</i>	21-23
<i>1.4.3.4.1. Mamdani Type Fuzzy Control System</i>	21
<i>1.4.3.4.1.1. Steps of Mamdani Type Fuzzy Controller System</i>	22
<i>1.4.3.4.1.2. Centre of Area defuzzification method</i>	22

1.4.3.4.2. Takagi-Sugeno Type Fuzzy Control System	22-23
1.4.4. Evolutionary Computation	23-24
1.4.5. Neural Networks	24-33
1.4.5.1. Basic structure of Neural Network	25-28
1.4.5.2. Brief introduction to the architecture of CNN	28-32
1.5. Thesis Organization	32-33
Chapter 2	35-67
2.1. Three crucial aspects of image dehazing	35
2.2. Brief description of daytime image dehazing methods	36-40
2.2.1. Dark Channel Prior	36-38
2.2.1.1. Challenges which arise while performing nighttime image dehazing using DCP	37-38
2.2.2. Daytime image dehazing methods	38-40
2.2.3. Inefficiency of daytime image dehazing methods in performing nighttime image dehazing	40
2.2.4. Nighttime image models	40-42
2.2.4.1. New imaging model	40-41
2.2.4.2. Nighttime imaging models (including the glow characteristics)	41-42
2.2.4.2.1. Multiple light colors and glow model	41
2.2.4.2.2. Nighttime haze model model	41
2.2.4.2.3. Improved atmospheric scattering model	41-42
2.2.4.3. Relaxed atmospheric scattering model	42
2.2.4.4. Nighttime hazy image model with HDP function	42
2.2.4.5. Survey on existing nighttime image dehazing methods	42-67
2.2.4.5.1. Methods designed following Atmospheric scattering model	43-48
2.2.4.5.1.1. Image color transfer theory based methods	43-47
2.2.4.5.1.2. Multi-scale Fusion based method	47-48
2.2.4.5.2. Methods designed following New imaging model	48-51
2.2.4.5.2.1. Illumination compensation & color correction based methods	48-50
2.2.4.5.2.2. Maximum Reflectance Prior based method	50-51
2.2.4.5.3. Methods designed following Multiple Light Colors & Glow model	51-55
2.2.4.5.3.1. Filtering based method	51-52
2.2.4.5.3.2. Weighted Entropy based methods	52-54
2.2.4.5.3.3. Super-pixel based method	54-55
2.2.4.5.4. Method designed following Nighttime haze model	55-56
2.2.4.5.4.1. Deep neural network based method (DeGlow-DeHaze Network)	55-56
2.2.4.5.5. Method designed following Improved atmospheric scattering model	56-57
2.2.4.5.5.1. Fusion based method	56-57
2.2.4.5.6. Methods designed following Relaxed atmospheric scattering model	58-60

2.2.4.5.6.1. Colour line model based method	58-59
2.2.4.5.6.2. Pixel-wise alpha blending based method	59-60
2.2.4.5.7. Method designed following Nighttime Hazy Image Model with HDP Function	60-61
2.2.4.5.7.1. Deep neural network based method (HDP-Net)	60-61
Chapter 3	62-90
3.1. Novel Fuzzy Logic & Bacterial Foraging (BF) synergism based daytime image dehazing method	62-90
3.1.1. FI rules based contrast enhancement	64-67
3.1.1.1. Reasons behind choosing HC as an input linguistic variable of the designed FI rules	64
3.1.1.2. Reasons behind choosing LSF as an input linguistic variable of the designed FI rules	64
3.1.1.3. Details of designed FI rules based contrast enhancement method	65-66
3.1.1.3.1.1. Dynamic Range(DR) of linguistic variables used in these FI rules	65 -67
3.1.2. Novel BF-Fuzzy synergism based simultaneous edge & noise detection	67-88
3.1.2.1. Search space	68
3.1.2.2. Chemotaxis	68-79
3.1.2.3. Reproduction and Elimination step	79
3.1.2.4. Constraints	79
3.1.2.5. Comparative analyses of noise removal methods	79-84
3.1.2.6. Comparative analysis of edge detection methods	84-88
3.1.3. FI rules based edge-sharpening	88-90
3.1.4. Block diagram	90
Chapter 4	91-104
4.1. Databases	91-92
4.1.1. Waterloo IVC Dehazed Image Database	91
4.1.2. Middlebury Database	91
4.1.3. D-HAZY Database	91
4.1.4. I-HAZE Database	91
4.1.5. O-HAZE Database	91-92
4.2. Quantitative parameters	92
4.2.1. Full-reference quantitative parameters	92
4.2.2. No-reference quantitative parameters	92
4.3. Experimental results	93-104
Chapter 5	106-113
5.1. Fuzzy Logic based reference image generation technique	108-111
5.2. Experimental Results	111-113
Chapter 6	115-140
6.1. S-HAZE database	115-123
6.1.1. Images belonging to S-HAZE database	116-123
6.2. N-HAZE database	123-130
6.3. D&N-HAZE Database	130-149

Chapter 7	151-168
Chapter 8	170-174
8.1. Literature Survey	170-
<i>8.1.1. Counting based automated poikilocytes detection methods</i>	170-171
<i>8.1.2. Handcrafted features based automated erythrocytes classification methods</i>	171-173
<i>8.1.3. CNN based automated erythrocytes classification methods</i>	173-174
Chapter 9	175-185
9.1. Data collection	175
9.2. Fine-tuning	175
9.3. Construction of feature spaces	175-176
9.4. Proposed two-stage feature selection method	176-180
<i>9.4.1. Fuzzy Entropy</i>	176
<i>9.4.2. Total Contribution Score</i>	176-178
<i>9.4.3. First stage of the proposed feature selection algorithm</i>	178-179
<i>9.4.4. Second stage of the proposed feature selection algorithm</i>	179-180
9.5. Experimental Results	180-185
Chapter 10	187-202
10.1. Existing despeckling filters	187-201
<i>10.1.1 Lee filter and its' proposed variants</i>	187-191
<i>10.1.2. Median filter and its' proposed variants</i>	191-194
<i>10.1.3. Hybrid Median Filter and its proposed variants</i>	194-195
<i>10.1.4. Frost filter and its' proposed variants</i>	196-201
10.2. Existing CNN based POLSAR image classification methods	200-201
Chapter 11	202-209
11.1. Filtering mechanism of the proposed filter	202-209
Chapter 12	210-242
12.1. Crop-Net architecture	210-211
12.2. Feature extraction	211
12.3. Feature selection	211-235
12.4. Experimental results	236-242
Chapter 13	243-244
References	245-253

List of figures

Chapter 1

<i>Fig.1.1.1</i> Pictorial representation of hazy image formation technique	5
<i>Fig.1.1.2.</i> (a) and (c): Daytime hazy image, (b) and (d): Histograms of (a) and (c)	5
<i>Fig.1.1.3.</i> (a) and (c): Nighttime hazy image, (b) and (d): Histograms of (a) and (c)	6
<i>Fig.1.1.1.1.</i> Daytime and Nighttime images ((a) & (b)), ((c) & ((d)) have almost similar scene contents)	6
<i>Fig.1.1.1.1.1</i> (a) Daytime haze-free image, ((b) & (c): Sky and Non-sky regions of (a)), ((d) & (e): Histograms of (b) & (c)), (f) Legend used in the histogram plot	7
<i>Fig.1.1.1.2.1</i> (a) Nighttime haze-free image, ((b) & (c): Sky and Non-sky regions of (a)), ((d) & (e): Histograms of (b) & (c)), (f) Legend used in the histogram plot	7
<i>Fig.1.1.1.2</i> (a)-(b)-Daytime hazy images, (c)-(e)-Nighttime hazy images	7
<i>Fig.1.2.2.1.</i> Sickle cell	11
<i>Fig.1.2.2.2.</i> (a) Echinocytes and (b) Acanthocytes	12
<i>Fig.1.2.2.3</i> Description of erythrocytes which are detected using the proposed method	13
<i>Fig.1.3.4.1</i> An example of SAR image	14
<i>Fig.1.3.4.2.</i> (a) POLSAR image of AIRSAR L Band Flevoland data, (b) Ground Truth image of (b) and (c) Class labels	15
<i>Fig.1.3.5.1.</i> Examples of thematic maps designed for Soil moisture	16
<i>Fig.1.4.3.1.</i> Fuzzy membership curves representing Fuzzy sets SHORT, MEDIUM and TALL	18
<i>Fig.1.4.3.2.1.</i> Graphical representation of Y function	18
<i>Fig. 1.4.3.2.2.</i> Graphical representation of s function	19
<i>Fig.1.4.3.2.3.</i> Graphical representation of L function	19
<i>Fig.1.4.3.2.4.</i> Graphical representation of Triangular membership function	19
<i>Fig.1.4.3.2.5.</i> Graphical representation of Π function	20
<i>Fig1.4.3.2.6.</i> Graphical representation of Gamma function	20
<i>Fig.1.4.3.4.1.</i> Block diagram of Fuzzy Controller System	23
<i>Fig.1.4.5.1.1.</i> A simple feedforward Neural Network	25
<i>Fig.1.4.5.1.2.</i> Some examples of activation function	26
<i>Fig.1.4.5.1.3.</i> Sample data (set 1) of each class	26
<i>Fig.1.4.5.1.4.</i> Sample data (set 2) of each class	27
<i>Fig.1.4.5.2.1.</i> Basic architecture of CNN	29
<i>Fig.1.4.5.2.1.1.</i> (a) $6 \times 6 \times 1$ image, (b) 3×3 Laplacian mask and (c) Generated feature map of $4 \times 4 \times 1$ dimension	29
<i>Fig.1.4.5.2.1.2.</i> Examples of pixel-wise feature value computation	29-30
<i>Fig.1.4.5.2.1.3.</i> Feature map computation of a color image	30-31
<i>Fig.1.4.5.2.1.4.</i> Feature map obtained after Max Pooling	31
<i>Fig.1.4.5.2.1.5.</i> Feature map obtained after Average Pooling	31
<i>Fig.1.4.5.2.1.6.</i> Feature map obtained after Sum Pooling	31

Chapter 2

<i>Fig. 2.2.4.2.1.1.</i> Pictorial representation of nighttime imaging environment	41
<i>Fig.2.2.4.5.1.</i> Model-wise grouping of nighttime image dehazing methods	42
<i>Fig.2.2.4.5.1.1.1.</i> (a) Source image, (b) Reference image (with different scene contents), (c) Output image	43
<i>Fig.2.2.4.5.1.1.2.</i> (a) Source image, (b) Reference image (with almost same scene contents), (c) Output image	43
<i>Fig. 2.2.4.5.1.1.3.</i> (a) Source image (S), (b) R1, (c) R2, (d) R3, (e) O (S, R1), (f) O (S, R2), (g) O (S, R3)	44
<i>Fig. 2.2.4.5.1.1.4.</i> (a) Source image (S), (b) R1, (c) R2, (d) R3, (e) O (S, R1), (f) O (S, R2), (g) O (S, R3)	44
<i>Fig. 2.2.4.5.1.1.5.</i> (a) Source image, (b) Reference image, (c) O1, (d) O2, (e) O3	47
<i>Fig. 2.2.4.5.2.2.1.</i> (a) Image 1, (b) Illumination map of Image 1, (c) Image 2, (d) Illumination map of Image 2	50
<i>Fig. 2.2.4.5.3.1.1.</i> (a) Image 1, (b) G of Image 1, (c) H of Image 1(d) Image 2, (e) G of Image 2, (f) H of Image 2	51
<i>Fig.2.2.4.5.4.1.1.</i> Architecture of DeGlow network	55
<i>Fig.2.2.4.5.4.1.2.</i> Pictorial representation of the workflow of the method proposed in [61]	56
<i>Fig.2.2.4.5.5.1.1.</i> Pseudo-code of the atmospheric light estimation method	57
<i>Fig.2.2.4.5.5.1.2.</i> Block diagram of the method	57
<i>Fig.2.2.4.5.6.1.1.</i> Intersection of two non-parallel patch planes with distinct color lines l_1 and l_2	58
<i>Fig. 2.2.4.5.7.1.1</i> HDP-Net architecture	60

Chapter 3

<i>Fig.3.1.1.3.1.</i> Graphical representations of fuzzy membership functions used to represent linguistic variables (a) HC, (b) LSF and (c) MP	65-66
<i>Fig 3.1.2.2.1.</i> Eight possible edge-directions within a 3x3 patch	68
<i>Fig.3.1.2.2.2.</i> (a) N, (b) S, (c) W, (d) E, (e) NW, (f) NE, (g) SW, (h) SE (Red arrows represent probable edge directions, blue arrows signify locations of pixels lying perpendicular to probable edge directions)	70
<i>Fig. 3.1.2.2.3.</i> (a) Possible edge-direction (marked with red arrow) and (b)-(e) Neighboring pixels which lie perpendicular to the possible edge-direction (marked with blue arrows)	71
<i>Fig. 3.1.2.2.4.</i> Graphical representations of fuzzy membership functions used for representing linguistic variables of FI rules designed for edge & noise detection (a) $ \nabla D_x $ and (b) D_0	73-74
<i>Fig. 3.1.2.2.5.</i> Graphical representations of fuzzy membership functions used for representing linguistic variables of FI rules designed for filtering of noisy pixels (a) ∇_{N_x} and (b) a'	76
<i>Fig.3.1.2.2.6.</i> Pixels considered for noise filtering if SW has highest membership in NOISE fuzzy set	77
<i>Fig.3.1.2.2.7.</i> 3×3 neighbourhood of the possible noisy pixel $I(x-1, y+1)$	77
<i>Fig.3.1.2.2.8.</i> 3×3 neighbourhood of the possible noisy pixel $I(x, y)$	78
<i>Fig.3.1.2.1.</i> Pseudo code of the BF and Fuzzy synergism based edge & noise detection method	79
<i>Fig.3.1.2.5.1.</i> 3×3 filter window	80
<i>Fig.3.1.2.6.1.</i> 3×3 image patch centered at pixel position, (x, y)	85
<i>Fig.3.1.2.6.2.</i> Some examples of edge maps estimated using the proposed method	88
<i>Fig. 3.1.3.1.</i> Graphical representations of fuzzy membership functions used for representing linguistic variables of FI rules designed for edge-sharpening (a) dD_0 , (b) c	90
<i>Fig.3.1.4.1.</i> Block diagram of the proposed method	90

Chapter 5

- Fig.5.1.** Attenuation of color channels (a) Hazy image, (b), (c) and (d): R , G and B channels of (a) respectively 106
- Fig.5.2.** Hazy (source) image, (b)-(c): Daytime hazy images, (d)-(e): Color transferred outputs obtained from [66] using (b) & (c) as reference images , (f) Color transferred output obtained from CCT 106
- Fig.5.3.** Color transferred outputs obtained using different values of 'k' 107
- Fig.5.1.1.** Graphical representation of fuzzy membership values of (a) sa, (b) in and (c) k 109-111

Chapter 8

- Fig.8.1.2.1.** Overlapping characteristics of handcrafted features (a) Compactness, (b) Eccentricity, (c) Solidity,(d)Form Factor, (e) Medical Axis Ratio, (f) 7 Invariant Moments 172-173

Chapter 9

- Fig.9.4.2.1.** Neural network designed to perform TCS evaluation 176
- Fig. 9.4.3.1.** Psuedo code of the 1st stage of the feature selection of each CNN model 178-179
- Fig.9.4.1.** Block diagram of the proposed two-stage feature selection method 180
- Fig. 9.5.1.** FE and TCS values of features (a) FE values, (b) TCS values (x-axis: Feature dimension, y-axis: Parameter values) 181
- Fig.9.5.2.** FE and TCS values of features selected at different stage (a) First stage, (b) Second stage(x-axis: Feature dimension, y-axis: Parameter values)(Please zoom to see the axis ranges) 181-182
- Fig.9.5.3.** Examples showing output results (bbox with maximum score are displayed only) (Please zoom to see class labels) 185

Chapter 10

- Fig.10.1.1.1.** Pseudo-code of the filtering strategy adapted by the Improved Lee Sigma Filter 189
- Fig.10.1.2.1.** Sequence following which sorting is performed by PSA algorithm 192
- Fig.10.1.2.2.** Sequence following which sorting is performed by RPSA algorithm 192
- Fig.10.1.2.3.** 3D sliding window 192
- Fig.10.1.2.4.** (a) and (b) Median value computation techniques proposed in [196] and [197] respectively 193
- Fig.10.1.3.1.** Pictorial representation of the filtering algorithm proposed in [198] 195
- Fig.10.1.4.1.** 2D sub-window comprising of 90° neighbors 196
- Fig. 10.1.4.2.** Eight possible edge directions in each 5×5 processing window (Shaded areas represent possible edge directions) 198

Chapter 11

- Fig.11.1.1.** 5×5 image patch 202
- Fig.11.1.2.** Graphical representation of fuzzy membership function (Triangular) used to represent (a) $diff_{scatt_diversity}$, (b) $diff_{degree_purity}$ and (c) w 204
- Fig.11.1.3.** Graphical representation of fuzzy membership function (Gaussian) used to represent (a) $diff_{scatt_diversity}$, (b) $diff_{degree_purity}$ and (c) w 205

Chapter 12

- Fig.12.1.1.** Crop-Net architecture 210

List of Tables

Chapter 1

<i>Table 1.1.1.1.</i> Characteristics of Daytime Vs Nighttime images	6-7
<i>Table 1.4.5.1.1.</i> Sample features (set 1)	27
<i>Table 1.4.5.1.2.</i> Sample features (set 2)	27

Chapter 2

<i>Table 2.2.1.1.</i> Assumptions considered in DCP	37
<i>Table. 2.2.4.5.1.1.1.</i> Basic differences between the color transfer techniques used in [66], [70] and [73]	47
<i>Table. 2.2.4.5.7.1.1.</i> Description of layers of HDP-Net	60-61

Chapter 3

<i>Table 3.1.1.3.1.</i> FI rules designed for performing contrast enhancement	65
<i>Table 3.1.2.2.1.</i> FI rules designed for performing simultaneous edge& noise detection	72-73
<i>Table. 3.1.2.2.2.</i> FI rules designed for filtering noisy pixels	75
<i>Table 3.2.1.5.1.</i> Comparative qualitative results obtained using CT images	82-83
<i>Table 3.2.1.5.2.</i> Comparative qualitative results obtained using CT images	83
<i>Table.3.1.2.6.1.</i> 3×3 kernels used for performing edge detection by Roberts, Prewitt and Sobel edge detectors	84
<i>Table. 3.1.2.6.2.</i> Effect of threshold (T) on estimated edge maps	87
<i>Table.3.1.2.6.3.</i> Effect of hysteresis thresholding on estimated edge maps	87
<i>Table 3.1.3.1.</i> FI rules designed to perform edge-sharpening	89

Chapter 4

<i>Table 4.3.1.</i> Comparative qualitative analyses results obtained using Waterloo IVC Dehazed Image database	93-94
<i>Table 4.3.2.</i> Comparative qualitative analyses results obtained using Middlebury Image database	94-95
<i>Table 4.3.3.</i> Comparative quantitative analyses results obtained using Middlebury Image database	96
<i>Table 4.3.4.</i> Comparative qualitative analyses results obtained using D-HAZY database	97-98
<i>Table 4.3.5.</i> Comparative quantitative analyses results obtained using D-HAZY database	98
<i>Table 4.3.6.</i> Comparative qualitative analyses results obtained using I-HAZE database	99-100
<i>Table 4.3.7.</i> Comparative quantitative analyses results obtained using I-HAZE database	100-101
<i>Table 4.3.8.</i> Comparative qualitative analyses results obtained using O-HAZE database	101-102
<i>Table 4.3.9.</i> Comparative quantitative analyses results obtained using O-HAZE database	103
<i>Table 4.3.10.</i> Meaning of symbols used in Table 4.3.1-Table 4.3.9	104

Chapter 5

<i>Table 5.1.1.</i> FI rules designed for reference image generation	109
<i>Table 5.2.1.</i> Comprehensive qualitative analysis results obtained using real world images	112-113
<i>Table 5.2.2.</i> Comprehensive quantitative analysis results obtained using real world images	113

Chapter 6

<i>Table 6.1.1.1.</i> Images having large sky regions	116-118
<i>Table 6.1.1.2.</i> Comparative quantitative analysis results (using full-reference quantitative parameters)	118-119

Table 6.1.1.3. Comparative quantitative analysis results (using no-reference quantitative parameters)	119
Table 6.1.1.4. Images having little sky regions	119-121
Table 6.1.1.5. Images having no sky regions	121-123

Chapter 7

Table 7.1. Selected parameters' values in a sky patch	152
Table 7.2. Selected parameters' values in a non-sky patch	153
Table 7.3. Novel sets of FI rules designed to perform sky segmentation	154-157
Table 7.4. Graphical representations of membership functions used to represent linguistic variables	157-166
Table 7.5. Results obtained using the proposed Fuzzy controller	167-168

Chapter 9

Table 9.5.1. Stage-wise reduction in dimension of each CNN's feature space	182
Table 9.5.2. Improvement in results obtained due to the introduction of the proposed two-stage feature selection algorithm	182
Table 9.5.3. Comparative analysis of proposed method with existing works using ALL –IDB database	183
Table 9.5.4. Comparative analysis of proposed method with existing CNN based works	183
Table 9.5.5. Comparative analysis of the proposed method with existing handcrafted features based methods	183-184

Chapter 11

Table 11.1.1. Novel FI rules designed to perform weight estimation	203
Table 11.1.2. Comparative qualitative analysis results	206-208
Table 11.1.3. Comparative quantitative analysis results	208

Chapter 12

Table 12.2.3.1. Estimated information content and TCS_{avgrank} of each feature	212-216
Table 12.2.3.2. Features included in the first set	216
Table 12.2.3.3. Features included in the second set	217
Table 12.2.3.4. Features included in the third set	218-219
Table 12.2.3.5. Features included in the fourth set	219-220
Table 12.2.3.6. Features included in the fifth set	221-223
Table 12.2.3.7. Features included in the sixth set	223-225
Table 12.2.3.8. Features included in the seventh set	225-228
Table 12.2.3.9. Features included in the eighth set	228-231
Table 12.2.3.10. Features included in the ninth set	232-235

Some commonly used symbols

Symbol	Meaning
I	Hazy Image
$t(x)$	Transmission value of pixel located at any pixel location say, x
A	Atmospheric light
x	Random pixel location
β	Scattering coefficient
d	Scene distance
$I_{\text{speckle-corrupted}}(i, j)$	Intensity of image corrupted with speckle noise at pixel position, (i, j)
$R_{\text{speckle-free}}(i, j)$	Intensity of image not corrupted with speckle noise at pixel position, (i, j)
$\eta(i, j)$	Speckle noise intensity at pixel position, (i, j)
U	Universal set
X, Y	Fuzzy sets
$\mu_X(el)$	Membership value of the element, el in Fuzzy set, X
N_b	Number of bacteria
$J(\theta^i), \theta^i \in R^{D_m}$	Cost function representing nutrient concentration in R^{D_m} (search space) of dimension D_m
$\theta^i(p, q, r)$	Position of the i^{th} bacterium in R^{D_m} (search space) after p^{th} chemotactic, q^{th} reproductive and r^{th} elimination-dispersal steps
$C(i)$	Step size considered for bacteria movement
N_s	Number of swims
$\Delta(i)$	Random vector used to determine the direction of movement after tumble
N_b	Total number of bacteria
N_{br}	Total number of healthy bacteria
J	Haze-free image/ Recovered scene radiance
J^{dark}	Dark channel of any daytime, haze free image say, J
$\Omega(x)$	Patch centred at pixel location x
$\hat{t}(x)$	Refined transmission value of pixel located at any pixel location say, x
ω	Haziness factor
$L(x)$	Intensity of light at any pixel location say, x
$\eta^c(x)$	Colour characteristics of any pixel located at any pixel location say, x in the colour channel c
$R^c(x)$	Reflectance of any pixel located at any pixel location say, x in the colour channel c
$A(x)$	Atmospheric light value at any pixel location say, x
$A_a(x)$	Glow intensity at any pixel location say, x
$APSF$	Atmospheric point spread function
$\hat{S}_k(x)$	Shape of ' k^{th} ' glow source influencing the pixel located at any pixel location say, x
St	Structure layer
Te	Texture layer
St_A	Glow part of Structure layer
St_B	Background layer (Glow free hazy part of Structure layer)
\hat{St}	Optimized Structure layer
μ	Enhancement parameter introduced to control texture level of output haze-free image say, J
\hat{Te}	Optimized Texture layer
I_h	Predicted haze density map of hazy image, I
M_s^l	Mean of l channel of Source image
M_s^α	Mean of α channel of Source image
M_s^β	Mean of β channel of Source image
M_r^l	Mean of l channel of Reference image
M_r^α	Mean of α channel of Reference image
M_r^β	Mean of β channel of Reference image
r	Reference image
s	Source image
$\frac{\sigma_r^l}{\sigma_s^l}$	Ratio between the standard deviations of l channel of Reference and Source images
$\frac{\sigma_r^\alpha}{\sigma_s^\alpha}$	Ratio between the standard deviations of α channel of Reference and Source images
$\frac{\sigma_r^\beta}{\sigma_s^\beta}$	Ratio between the standard deviations of β channel of Reference and Source images
l	l channel of hazy image, I in $l\alpha\beta$ colour space
α	α channel of hazy image, I in $l\alpha\beta$ colour space
β	β channel of hazy image, I in $l\alpha\beta$ colour space

I'	Output image obtained after performing image colour transfer
l'	l channel of colour transferred output image, I' in $l\alpha\beta$ colour space
α'	α channel of colour transferred output image, I' in $l\alpha\beta$ colour space
β'	β channel of colour transferred output image, I' in $l\alpha\beta$ colour space
$\frac{\sigma_r^l}{\sigma_{s,\Omega}^l(x)}$	Ratio between standard deviation of l channel of Reference image with patch-wise standard deviation of l channel of Source image
$\frac{\sigma_r^\alpha}{\sigma_{s,\Omega}^\alpha(x)}$	Ratio between standard deviation of α channel of Reference image with patch-wise standard deviation of α channel of Source image
$\frac{\sigma_r^\beta}{\sigma_{s,\Omega}^\beta(px)}$	Ratio between standard deviation of β channel of Reference image with patch-wise standard deviation of β channel of Source image
Z	Filtered image obtained by filtering hazy image, I with GIF
a_p, b_p	Self-adaptation parameters
$\Omega_{\zeta_1}(p')$	Patch of size $(2\zeta+1)\times(2\zeta+1)$ centered at pixel position p'
λ_{guided}	Regularization parameter used for penalizing large a_p
N	Total number of image pixels
$\sigma_{G,\zeta_1}(p')$	Standard deviation of a patch of size $(2\zeta+1)\times(2\zeta+1)$ centered at pixel p'
$\sigma_{G,1}(p')$	Standard deviation of a patch of size $(2\times 1+1)\times(2\times 1+1)$ centered at pixel p'
$\mu_{x,\infty}$	Mean of $\chi(p')$
ε	Infinitesimally small positive constant
$B^\circ p'$	Boundary information extracted by Prewitt operator
$Gray$	Grayscale image of I
$Detail$	Detail layer of I
$Saliency$	Saliency map of I
ψ	Patch having size twice as much as patch of size, Ω
$\widehat{R}^c(x)$	Surrogate reflectance value at any pixel position, x
ii	Lexographical representation of $\log(I^c(x))$
rr	Lexographical representation of $\log(\widehat{R}^c(x))$
ll	Lexographical representation of $\log(L(x))$
$\lambda_{ll}, \lambda_{rr}$	Smoothness parameters
L_p	Laplacian matrix
α_{weight}	Weight factor (whose value is equal to 0.5)
\widehat{I}^c	Illumination compensated image
η^c	Estimated colour map
$\underline{\eta}^c$	Lower bound of estimated colour map
\widetilde{I}^c	Colour corrected image
$\Delta\eta^c$	Residual colour term
$M_{\Omega}^c(x)$	Maximum reflectance map
G	Glow image
H	Glow free nighttime hazy image
λ	Smoothness parameter
$f_{1,2}$	First order derivative filter
f_3	Second order Laplacian filter
H_r	Red channel of Glow free nighttime hazy image, H
H_g	Green channel of Glow free nighttime hazy image, H
H_b	Blue channel of Glow free nighttime hazy image, H
A_{local}	Local atmospheric light
A_{global}	Global atmospheric light
k	Lightness values of LCh colour space
$p(k t)$	Probability of lightness k of calculated scene radiance when transmission is set to t
$J^L(x t)$	Lightness of calculated scene radiance at pixel location x when transmission is set to t
H_{mean}^L	Mean of H
$w(k t)$	Weight term when transmission is set to t
$\gamma(t)$	Information fidelity term when transmission is set to t
$h_j(t)$	Hue of scene radiance J when transmission is equal to t
$h_H(t)$	Hue of glow free nighttime hazy image H when transmission is equal to t
$\widetilde{S}t_B$	Enhanced background layer
\widehat{A}	Direction in which the colour line of a patch gets shifted from the origin of the plane
$m(x)$	Magnitude of airlight at pixel position, x

α_{shift}	Amount by which the colour line of a patch gets shifted from the origin of the plane
I^{cd}	Channel difference (Difference between bright channel and dark channel)
α	Brightness aware alpha map
t_{BCP}	Transmission estimated using BCP method
t_{DCP}	Transmission estimated using DCP method
$HC(i, j)$	Haze-concentration value at pixel position, (i, j)
$LSF(i, j)$	Log-sigmoid value at pixel position, (i, j)
MP^*	Defuzzified mapping constant value
a^*	Defuzzified correction term
$\theta^i(j, k, l)$	Current bacteria location
$\theta^i(j + 1, k, l)$	Bacteria location after tumble
$CNR_{dehazed\ output}$	Contrast-to-Noise ratio of dehazed output
CNR_{hazy}	Contrast-to-Noise ratio of hazy image
μ_A	Mean attenuation co-efficient of target region lying within region of interest
μ_B	Mean attenuation co-efficient of background region
σ_B	Background noise
n_r	Cardinal number representing the set of visible edges in dehazed outputs
n_o	Cardinal number representing the set of visible edges in original hazy image
dim_x	Height of the image
dim_y	Width of the image

ABSTRACT

Image restoration and image classification are the two most fundamental aspects of image processing.

Image restoration involves accurate preservation of colour, edge details, contrast, etc. of images which suffer from any sort of degradation either due to presence of noises like impulse noise, speckle noise, etc. or due to scattering and attenuation of scene light during inclement weather conditions. The scattering and attenuation of scene light increase during hazy weather condition due to the increase in the presence of dust, fog, mist and other aerosol particles in the atmosphere. Image restoration is necessary for various important applications like remote sensing, driver assistance systems, intelligent vehicles and for proper functioning of various computer vision algorithms.

Image classification deals with automated recognition of class labels of different objects present in images. In order to design accurate as well as automated image classification methods, it is very essential to enable the designed methods to learn the most distinctive as well as significant characteristics of the objects present in images. These features are fed as inputs to the designed methods during the training phase in order to enable those methods to distinguish between different objects using training algorithms which are designed to artificially replicate the learning process of human cognitive system, so that these methods can henceforth perform classification of similar objects based on the knowledge acquired by them about the characteristics of the objects during the learning (training) phase. Image classification has found its use in medical field, remote sensing applications, etc.

Image restoration deals with the preservation of lost details (color details/edge details/ contrast, etc. which are lost either due to noise corruption or due to atmospheric scattering and attenuation) in any degraded image and thus has to handle lot of uncertainties and randomness associated with the restoration process which arise due to the absence of true information about those lost details. Fuzzy Logic is one of the most important principles of Computational Intelligence which can effectively deal with such uncertainties and randomness.

Image classification deals with the identification of the class labels of different objects present in images by learning their characteristics. Classification is usually done using neural networks which is also an important principle of Computational Intelligence. These networks are designed to artificially replicate the learning behaviour of human cognitive system. These networks differ from one another based on the number of hidden layers (the layers which lie in between the input layer and output layer) present in their architecture.

Conventional neural networks usually comprise of lesser number of hidden layers while deep neural networks possess comparatively large number of hidden layers. The inherent feature extraction capability of deep neural networks have enabled them to improve their performance over conventional neural networks as it reduces the chances of erroneous classification which occur due to use of handcrafted features in conventional neural networks.

The above discussion clearly states the inter-relationship between Image processing and Computational Intelligence fields. In the present era, the principles of Computational Intelligence like Fuzzy Logic and Neural Networks have become the most favourable choices for the researchers to perform any image processing tasks as they can handle the fuzziness present in any data efficiently and can artificially replicate the learning behaviour of human cognitive system respectively. Novel methods for performing image restoration or image classification are designed in this thesis by exploiting the inter-connection between the Image processing and Computational Intelligence fields.

Three real life problems are studied in this thesis:

- a. Daytime and Nighttime image dehazing
- b. Detection of structurally variant erythrocytes.
- c. Land cover classification using full-polarimetric image data

These three problems have been chosen for study in this thesis because of their immense significance in real world. Detailed discussion on the practical significance of each of these topics is carried out in this thesis based on the statistical data available from various authorized sources.

A comprehensive literature survey on each of these topics is presented in this thesis and the shortcomings of the existing works have been listed down and new methods are proposed to overcome the shortcomings.

Some significant existing shortcomings of the chosen research areas which have served as the major inspiration behind choosing them are highlighted as follows:

1. Daytime and Nighttime image dehazing

Although image dehazing is a well-defined research area and many researchers have proposed several methods to perform effective dehazing of images (particularly daytime images) but in most of those methods, the authors have performed image dehazing mostly portraying it as a simple contrast enhancement problem. They have failed to consider edge-preservation and noise detection which are other equally important and crucial aspects of image dehazing.

Moreover, in most of the existing methods, image dehazing is performed assuming the nature of degradation to be similar across the entire image which in reality is not so as the nature of degradation varies region-wise across an image depending on many factors. So to perform effective image dehazing, there is a need of a method which performs dehazing considering the pixel-wise variation in the nature of degradation.

In addition to these shortcomings, another significant limitation in this research area lies in the fact that the existing research in this problem area is mostly focused on dehazing daytime images. Although nighttime image dehazing is a very significant research area but very few works are done in that research area. No proper systematic survey is conducted in that field. No proper benchmark databases comprising of real world nighttime hazy images as well as their Ground Truth images exist in this field.

2. Detection of structurally variant erythrocytes

The detection of structurally variant erythrocytes (poikilocytes) is an emerging research topic in medical image analysis field.

Inspired by the real-life significance of this field many researchers have proposed various poikilocytes classification or detection methods but those methods mostly perform classification or detection either by counting the number of poikilocytes present in blood smear images or using handcrafted textural, geometric and shape based features.

Methods designed in this field implementing the deep neural mechanism is very basic. Those methods simply use pre-trained networks to perform classification. This field currently lacks any deep neural architecture which can perform poikilocytes classification or detection using only significant and highly informative features from multiple networks.

3. Land cover classification using full-polarimetric image data

Deep neural network based land cover classification is an upcoming research area in remote sensing field. The number of existing deep neural networks which are designed especially to perform land cover classification is very few and their performances is mostly tested on Synthetic Aperture Radar (SAR) images.

As properties of SAR and Polarimetric Synthetic Aperture Radar (POLoSAR) images vastly differ from each other, so designing a deep neural network especially for performing land cover classification using POLoSAR images is required.

Some notable contributions of the works proposed in this thesis are: -

- a. Bacterial Foraging and Fuzzy Logic synergism based daytime image dehazing method.
- b. Fuzzy Logic and Refined Colour Channel Prior synergism based nighttime dehazing method.

- c. An extensive survey work is conducted in this thesis focusing on nighttime image dehazing field. Some topics which are highlighted in the survey work are listed as follows:
 - i. Dissimilarities between the characteristics of daytime and nighttime hazy images.
 - ii. Unexplored aspects and real-life significance of nighttime image dehazing.
 - iii. Inefficiency of the daytime image dehazing methods (methods which are designed based on the characteristics of daytime images) in performing nighttime image dehazing.
 - iv. Existing nighttime imaging models and the differences between the models as well as their variations from the atmospheric scattering model (daytime hazy image model)
 - v. Existing nighttime image dehazing methods and their limitations.

This survey needs special mention as it is the first review work which is published in the nighttime image dehazing field.

- d. S-HAZE, a novel database which is created to benchmark the sky scene restoration capabilities of image dehazing methods.
- e. N-HAZE, a novel database. (This is the first database which is designed exclusively for benchmarking the performances of nighttime image dehazing methods. It comprises of hazy as well as Ground Truth images.)
- f. Daytime and Nighttime Dehazing Database (D&N-HAZE Database), a novel database. (This is the first database among all currently existing databases in image dehazing field that comprises of both daytime and nighttime hazy images of similar scenes, captured in the presence of real atmospheric haze as well as synthetic haze along with their corresponding Ground Truth images).
- g. An automated intensity based sky segmentation method.
- h. Novel feature ensemble creation method. (This feature ensemble method is designed to select the best features among Fully Connected (FC), Rectified Linear Unit (ReLU) and InverseReLU features to nullify the information loss occurring due to the suppression of negative values in features by the ReLU activation layer present in Convolutional Neural Networks. The efficiency of the proposed feature ensemble creation method is validated by performing detection of nine different types of erythrocytes having varied morphology using the features ensemble created by the proposed method.)
- i. Total Contribution Score Parameter (a novel feature selection parameter).
- j. Ranking method (a novel feature selection method).
- k. A novel Degree of purity & Scattering diversity based Advanced Lee filter.
- l. Novel deep neural network namely, Crop-Net for performing POLSAR image classification is designed.

The utility of each of these proposed approaches in real world have been demonstrated and their efficiencies are validated by performing comparative analyses with the results obtained from existing methods. In most cases, the designed methods have achieved better performance efficiencies compared to existing methods.

Chapter 1

1. Introduction

Brief overview on each of the three problem statements studied in this thesis is presented in this Chapter which contains detailed discussion on the significance of each problem in real world based on statistical data available from authorized sources. Basic details of the principles of Computational Intelligence which are used for processing images in this thesis are given in this Chapter. This Chapter is concluded with a detailed description regarding the organization of the thesis.

1.1. Daytime and Nighttime image dehazing

Image dehazing is a type of image restoration problem which deals with preservation of the visibility of degraded images, captured during hazy weather conditions. The degradation of visibility of these images occur mainly due to the increase in scattering and attenuation of scene light during hazy weather conditions caused by the presence of fog, mist, dust and other aerosol particles in the atmosphere.

A degraded hazy image is usually represented using the following mathematical equation [1]:

$$I(x) = J(x)t(x) + A(1 - t(x)) \quad (1.1.1)$$

The above mathematical equation is also referred to as the atmospheric scattering model.

The two terms on the right hand side of (1.1.1) are termed as ‘direct attenuation’ and ‘airlight’ respectively. Among these terms, the first term i.e., the direct attenuation term is a multiplicative term. This term is used to estimate the exponential increase in scene decay in accordance to the increase in scene distance (as $t(x) = e^{-\beta d(x)}$) while the second term i.e. the airlight term is an additive term which gives direct estimation of region-wise haze-concentration of an image and also plays crucial role in performing image dehazing.

The terms β and A in equation (1.1.1) denote scattering co-efficient of scene light and atmospheric light respectively.

The term ‘ A ’ in the atmospheric scattering model (1.1.1) is globally uniform across the images (daytime images) and it is independent of x and $J(x)$. The atmospheric light in daytime images is mainly contributed by sunlight, whose influence over all the pixels in an image is considered to be uniform.

The airlight term, ‘ $A(1 - t(x))$ ’ in equation (1.1.1) represents the scattered/attenuated portion of scene light. As the distance between the camera/observer and the scene object increases, the scene light encounters more fog, mist, dust and aerosol particles on its’ way to the observer/camera and thus undergoes more attenuation/scattering. This increase in scattering/attenuation increases the value of the airlight which indicates more degradation of the visual quality of an image.

Mathematically, the above statement is explained as follows:

Let us assume, $A = 230$ and $\beta = 0.1$ (These parameters are considered to be uniform across an image.)

$$A(1 - t(x)) = A(1 - e^{-\beta d(x)})$$

When $d_1(x) = 10$

$$Airlight_1 = 230(1 - e^{-(0.1 \times 10)}) = 230(1 - 0.368) = 145.36$$

When $d_2(x) = 20$

$$Airlight_2 = 230(1 - e^{-(0.1 \times 20)}) = 230(1 - 0.135) = 198.95$$

For this reason, the ‘airlight’ consider the distance of between the object and the observer (camera), instead of the scatter and the the observer (camera).

The main aim of the dehazing problem is to recover the scene radiance say, ‘ J ’, from degraded hazy image say, ‘ I ’. In order to recover scene radiance (haze free image) from hazy image, one has to evaluate transmission and atmospheric light in addition to scene radiance. Hence, to perform dehazing of an image, three unknown variables (J, t and A) have to be evaluated using information derived from one known variable (I) which makes the image dehazing an ill-posed problem .

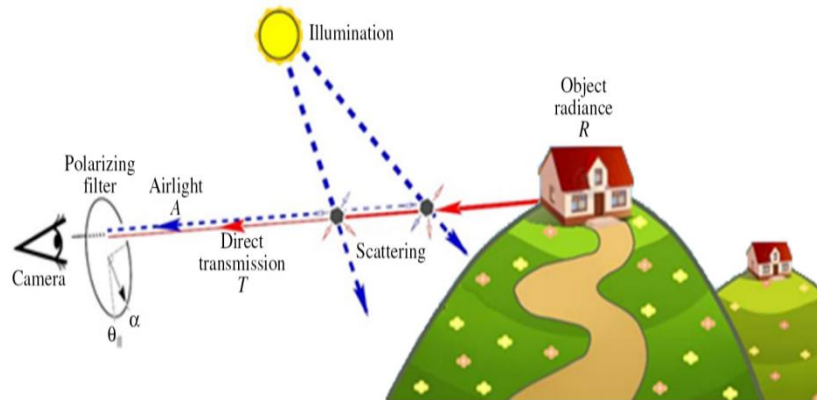


Fig.1.1.1 Pictorial representation of hazy image formation technique

The term ‘transmission’ represents the portion of the scene light that reaches to the camera sensor non-attenuated while airlight represents the scattered and attenuated portion of scene light. The visibility of images get degraded when the scattered and attenuated portion of scene light (airlight) combines with non-attenuated portion of scene light (transmission) at the camera sensor as shown in Fig.1.1.1.

It is quite evident from the images given in Fig.1.1.2 and Fig.1.1.3, that the presence of haze in the atmosphere leads to the degradation of contrast, colour and edge details of images which are captured during hazy weather conditions. Histograms of daytime and nighttime hazy images as well as their corresponding Ground Truth images in RGB colour space are also given to show how the change in the characteristics of images due to the presence of haze in the atmosphere.

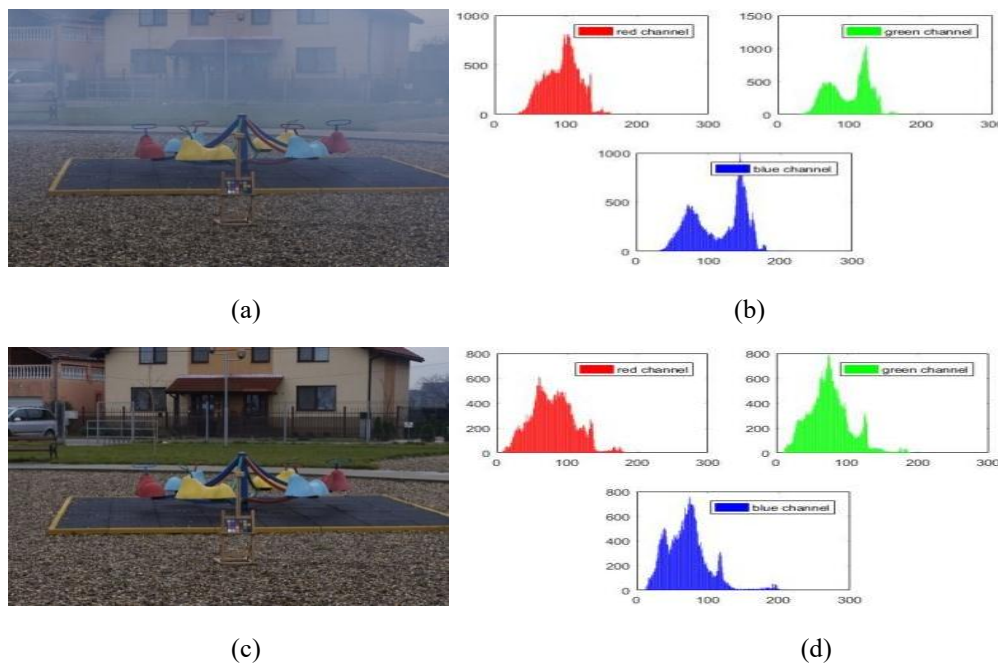


Fig.1.1.2. (a) and (c): Daytime hazy image, (b) and (d): Histograms of (a) and (c)

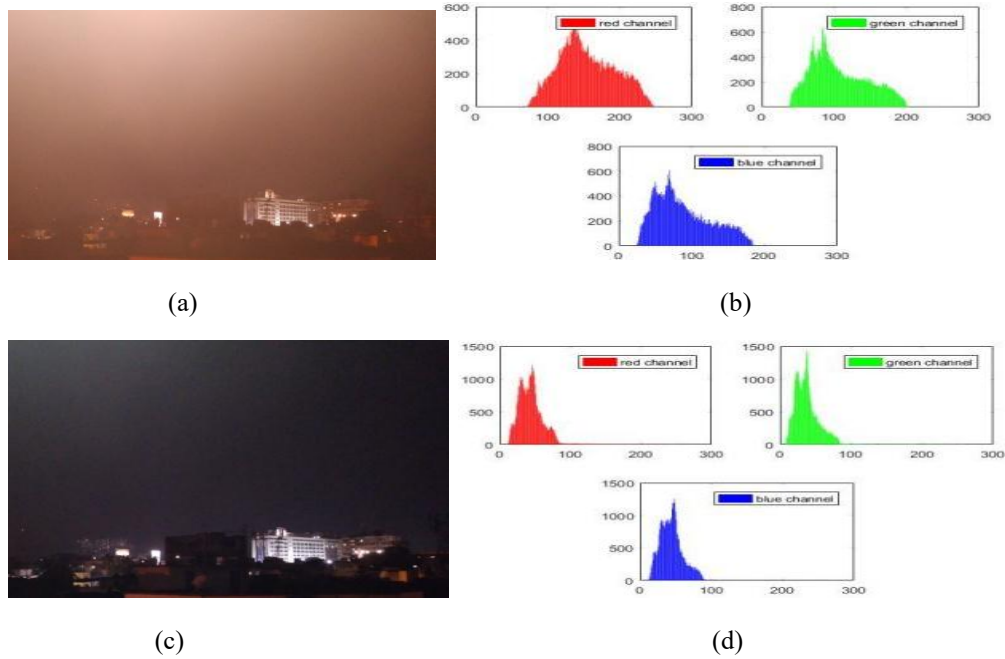


Fig.1.1.3. (a) and (c): Nighttime hazy image, (b) and (d): Histograms of (a) and (c)

Daytime hazy and haze-free images of the same scene and their corresponding histograms (given in Fig.1.1.2) suggest that the presence of haze in the atmosphere lead to the degradation of contrast, colour and edge details of images. Similar observation too holds for nighttime hazy and haze-free images of the same scene (given in Fig.1.1.3).

The differences in characteristics of daytime and nighttime images are discussed in details in the following section.

1.1.1. Daytime and Nighttime images' properties

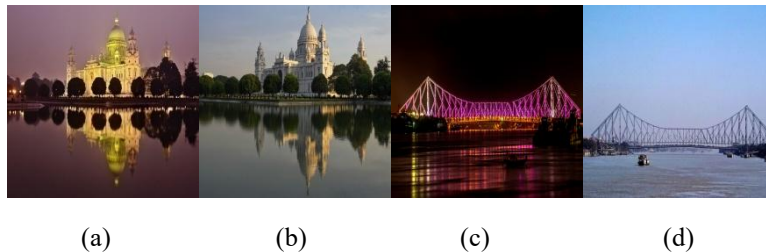


Fig.1.1.1.1. Daytime and Nighttime images ((a) & (b)), ((c) & (d)) have almost similar scene contents)

Table 1.1.1.1. Characteristics of Daytime Vs Nighttime images

Image characteristics	Daytime images	Nighttime images
Illumination	Globally uniform	Non-uniform (due to the presence of multi-colored artificial light sources)
Intensities of pixels belonging to sky regions of images	Usually High (due to the presence of bright sunlight during Daytime)	Usually Low (due to the absence of bright sunlight during Nighttime)
Source of scene light	Natural light source (sunlight)	Mostly artificial light sources (for e.g., street lights, automobile lights, etc.)

Color of atmospheric light	Usually white (as sunlight is a combination of various monochromatic lights)	Varies spatially in accordance to the variance of colors of lights emitted by multi-colored artificial light sources.
----------------------------	--	---

1.1.1.1. Sky Vs Non-sky regions of Daytime images

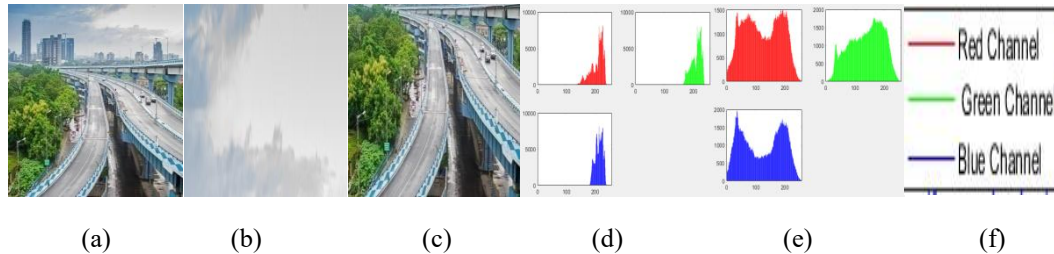


Fig.1.1.1.1.1 (a) Daytime haze-free image, ((b) & (c): Sky and Non-sky regions of (a)), ((d) & (e): Histograms of (b) & (c)), (f) Legend used in the histogram plot

1.1.1.2. Sky Vs Non-sky regions of Nighttime images

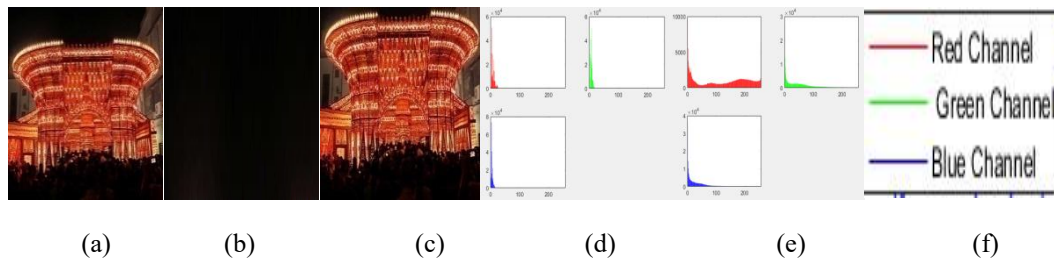


Fig.1.1.1.2.1 (a) Nighttime haze-free image, ((b) & (c): Sky and Non-sky regions of (a)), ((d) & (e): Histograms of (b) & (c)), (f) Legend used in the histogram plot

Daytime and nighttime images of almost similar scenes (given in Fig.1.1.1.1) and the images of sky and non-sky regions of daytime and nighttime images and their corresponding histogram plots (given in Fig.1.1.1.1.1 and Fig.1.1.1.2.1) depict that the properties of these images vary vastly. The differences in between the properties of daytime and nighttime images occur mainly due to the variations in their illumination characteristics. The illumination of daytime images is globally uniform as it is mostly contributed by sunlight. As sun is located at huge distance away from the earth (of the order of million kilometers) and the colour of sunlight is ideally white (as it is a combination of various monochromatic light sources), so the possibility of occurrence of non-uniform illumination in daytime images is considered to be negligible. On the other hand, the illumination characteristics of nighttime images vary spatially as it is mostly contributed by artificial light sources (for e.g., street light, automobile light, building light, etc.) which are mostly located in the non-sky regions of images and their influence is restricted within the nearby regions. The color and intensities of these multi-colored lights vary in accordance to the nature of the light sources.



Fig.1.1.1.2 (a)-(b)-Daytime hazy images, (c)-(e)-Nighttime hazy images

The variations in the illumination characteristics between daytime and nighttime hazy images are demonstrated in Fig.1.1.1.2. For better understanding, the artificial light sources in nighttime hazy images and the areas which are mostly influenced by those light sources are enclosed within boxes having red boundaries to show how light emitted by these sources mostly influence nearby regions. On the other hand, illumination of daytime hazy images appear to be globally uniform.

From the above discussion, it becomes clear that the differences in between the characteristics of daytime and nighttime images should be taken into account while designing image dehazing methods.

In the last decade, several researchers have proposed various daytime and nighttime image dehazing methods and designed various mathematical models to represent nighttime hazy images as atmospheric scattering model (1.1.1) cannot accurately represent the spatially varying illumination characteristics of nighttime hazy images as it contains constant atmospheric light term, 'A'. But this model can effectively represent globally uniform illumination characteristics of daytime hazy images. A comprehensive literature survey on existing dehazing methods is conducted and presented in Chapter 2. The details of nighttime imaging models designed to represent nighttime hazy images are given in Chapter 2.

Detailed description of the novel dehazing methods which are designed to perform haze removal from nighttime and daytime images focusing on the limitations of the existing works are given in Chapters 3, 4 and 5. The details of the databases which are created in this thesis to perform effective benchmarking of dehazing methods are given in Chapter 6.

1.1.2. Significance of image dehazing problem in real world

The data released by the Ministry of Road Transport and Highways [2], stated that the numbers of road fatalities occurring in India due to poor visibility caused by the presence of thick fog in the atmosphere (particularly during the months of December and January (winter)) have surged by almost 100% during 2014-2018. As per this data, the states lying in the Northern part of India like Punjab, Haryana, Uttar Pradesh witnessed maximum number of road fatalities occurring due to poor visibility as these states are engulfed in thick fog during winter season each year. Another report published in [3], has also stated that in Uttar Pradesh (a state in India) only, 12 persons were killed due to road accidents occurring in different districts on January 1, 2021 due to the presence of thick fog in the atmosphere.

These type of road fatalities is not a serious concern in India only but also worldwide. Data published in the official website of U.S. Department of Transportation, Federal Highway Administration [4] stated that over a time duration of 2007-2016, 3% of vehicle crashes, 3% of crash injuries and 9% of total weather-related crash fatalities were reported to have occurred due to fog-related visibility obstruction.

A comprehensive study was carried out in [5] based on the data of crashes which occurred in Florida during 2003-2007 due to poor visibility caused by the presence of fog and smoke in the atmosphere. After analyzing the crash data, the authors stated some interesting findings which are listed as follows:

- The morning hours of the months of December-February is the most prevalent time of a year for fog & smoke related crashes to occur.
- In comparison to the fatal crashes occurring during clear weather condition, the crashes occurring during fog related weather condition are more severe as such crashes involve large number of vehicles as well as persons which increase the chances of more people getting injured in such accidents.
- The severity of accidents occurring during fog & smoke related condition is also high as those accidents are either head on collisions or rear end collisions.
- The crashes occurring at night during fog & smoke related weather conditions are much more severe compared to the crashes occurring during daytime due to the absence of proper illumination at night in addition to poor visibility caused by the presence of fog in the atmosphere.

According to a recent report published in [6] on 15 February 2021, it is claimed that in a day at least 36 persons in Pakistan were killed by two separate fog-related accidents occurring on Siranwali-Kamoki road and Toba-Gojra road respectively.

Another report published in [7] on 24 November, 2020 claimed that a huge fire eruption took place on a Chinese Highway as more than 40 vehicles bumped into one another due to poor visibility caused by the presence of thick fog in the atmosphere. Three persons were killed in this accident and many suffered from severe injuries as they were trapped in fire.

The statistics and reports published in several authorized sources based on which the discussion on the significance of the image dehazing problem in real world is carried out, clearly demonstrates that visibility degradation caused by the presence of fog in the atmosphere is one of the major cause of occurrence of severe road fatalities. Similar data and statistics are provided in the following paragraph to show how situation worsens during nighttime compared to daytime due to the addition of proper illumination factor with the fog-related visibility degradation problem.

The authors in [8] conducted a study to predict the probability of occurrences of fatal accidents in comparison to the property damage only accidents using a Multiple Logistics Regression Model. This study, conducted using accident data provided by Expressway Authority of Thailand over the time span of 2007-2010 predicted that the severity of accidents increases at night. Similar study, carried out in [9] using the accident data available over the expressways of New Zealand also predicted that the chances of occurrences of severe accidents are highest during midnight to dawn period due to poor illumination as well as several other risk factors. The study carried out in [10] also stated that the chances of occurrences of severe accidents increase up to 2.31 times at nighttime compared to daytime.

The above discussion clearly states the necessity of installation of intelligent driver assistance systems in vehicles to restore the lost details of the captured degraded road scenes while driving during poor weather conditions and display the restored scenes to the drivers to make them aware of both moving and static objects present in those scenes, so that severe head on and rear end collisions can be avoided. In order to build an ideal driver assistant system which can perform restoration of lost details of scene images, it is very essential to design an effective dehazing method.

Apart from driver assistance systems and intelligent vehicles, image dehazing has also found its use in applications like remote sensing [11]-[13], surveillance system [14], etc.

1.2. Detection of structurally variant erythrocytes

1.2.1. Brief introduction on blood components

1.2.1.1. Erythrocytes/ red blood cells

According to the definition given in the official website of National Cancer Institute [15], red blood cells or erythrocytes are the types of cells which are formed in the bone marrow and are found in the blood stream. These cells contain a protein namely, haemoglobin which enables them to carry oxygen from lungs to other parts of the body.

1.2.1.2. Abnormality detection in blood components

The blood test which is conducted to count the total number of blood cells (red blood cells/ erythrocytes, white blood cells/leukocytes, plasma and platelets) is called Complete Blood Count (CBC) test. This test is conducted as a preliminary measure to detect any type of underlying disease in human body, as the CBC report of any infected person will show that at least one or more components (erythrocytes, leukocytes, plasma and platelets) of human blood have deviated from its normal range.

CBC test report includes:

- A. Leukocytes/White Blood Cells Count
- B. Erythrocytes/Red Blood Cells Count
- C. Platelet Count

D. Hematocrit red blood cell volume

E. Haemoglobin (oxygen carrying pigment) concentration in red blood cells.

F. Differential blood count.

According to the data published in the official websites of American Society of Haematology [16] and American Red Cross [17], the blood of a healthy human person comprises of 55% plasma, 1% of leukocytes (White Blood Cells) and about 40-45% of red blood cells and plasma.

1.2.1.3. Basic functions of blood components

Each component of human blood plays a vital role in maintaining the normal functionalities of each organ of human body. For e.g.

The main function of **plasma** is to transport blood cells along with antibodies, nutrients, clotting proteins, hormones, etc. all throughout the human body.

Red blood cells (erythrocytes) are used to carry oxygen from lungs to other parts of the body and carbon dioxide from other parts of the body to lungs to exhale it.

White blood cells (leukocytes) mostly deal with the production of antibodies, regulation of the performance of immune cells in order to enable the body to fight an infection.

Platelets are fragmented cells which mainly help in coagulation (clotting of blood) at the site of injury in order to prevent excessive loss of blood.

Deviation in the total count of any of these components from the normal range indicates the presence of some disease or infection in the human body.

Apart from count, other parameter which can be used to detect anomalies in blood cells is their morphological structures. In this thesis, a novel deep neural network based automated method is designed to detect nine different types of erythrocytes based on the variations in their morphology.

1.2.1.4. Poikilocytes

Normally human red blood cells have biconcave disc shaped like structures but in many cases when a person suffers from diseases like Sickle Cell Anaemia (SCA)/Sickle Cell Disease (SCD), elliptocytosis, thalassemia, etc. or any kind of nutrient deficiency like iron, vitamin B12, etc., the shapes of erythrocytes change from its normal biconcave nature.

Erythrocytes whose shapes differ from the biconcave disc shaped structure of normal erythrocytes are called poikilocytes. Details of some poikilocytes and the diseases associated with them are given in the following Section.

When the number of poikilocytes in blood stream of a person exceeds a certain permissible threshold, it is concluded that the person is suffering from some underlying medical conditions.

1.2.2. Significance of poikilocytes detection in real world

Poikilocytes in human blood stream indicates the presence of several life-threatening diseases. Hence, early detection of poikilocytes in human blood is absolutely essential for timely treatment of the underlying medical conditions in order to reduce their severity.

A detailed discussion on the severity of diseases which can be detected by the presence of poikilocytes in the human blood stream is carried out in this Section based on several data, statistics and reports available from various authorized sources.

Anaemia is one the most common as well as fatal disease which can be detected by the presence of poikilocytes in the human blood stream. Anaemia is mostly caused due to nutrient deficiency, mostly iron deficiency but in addition to that, the deficiency of Vitamin B12 and Vitamin A cause anaemia. Apart from nutrient deficiency, the presence of parasitic infections in human body like malaria, tuberculosis, HIV, etc. also cause anaemia. Although anaemia is prevalent mostly in low-income and

middle-income countries as malnutrition among children and pregnant women is common in those countries but for developed countries too it is a serious health concern. According to the data published in the official website of World Health Organization (WHO) [18], it is stated that in 2016 approximately 42% of children aged within 5 years and 40% of pregnant women suffered from this medical condition across the world.

The data published in the official website of National Health Portal of India [19], stated that the prevalence of anaemia is prominent among the pregnant women residing in South Asian Countries and India contributes about 80% of the total maternal deaths occurring due to anaemia in South Asia. The same report also stated that anaemia during pregnancy is also an important cause behind the occurrence of premature delivery, child and maternal death, low birth weight, etc.

Although the prevalence of anaemia among young children and women in India is a well-studied topic, but the number of studies conducted to estimate the prevalence of anaemia among Indian men is very less. According to an article published in the 'The Landcet Global Health' journal [20] it is stated that approximately 23.2% of Indian men aged between 15-54 years suffer from anaemia. Out of them, at least 5.4% suffer from moderate or severe anaemia and 0.5% suffer from severe anaemia. The authors in [20] have carried out their study considering the male population of all the States and Union Territories of India and have stated that the geographical, socio-demographic, economic and environmental factors have high correlation with the state-wise/district-wise number of anaemia cases.

SCA/SCD is a type of anaemia which leads to the production of an unusual type of haemoglobin namely, haemoglobin S in the body of a person who is suffering from SCD. Haemoglobin S changes the normal biconcave disc shape like structure of erythrocytes to sickle like structure. SCA/SCD is usually detected by the presence of sickle shaped red blood cells in human body as shown in Fig.1.2.2.1.



Fig.1.2.2.1. Sickle cell

In a recent report published in 2020 [21] based on the data available from American Society of Haematology, it is stated that the median life expectancy of people suffering from SCA/SCD is between 42-47 years.

According to an article published in the bulletin released by WHO [22], it is stated that although SCA/SCD is a serious health concern in all the countries across the world, yet it is mostly under recognized in sub-Saharan Africa which is home to more than 25 million persons suffering from SCA/SCD. It is also reported that approximately 75% of the total 400000 infants born every year with SCA/SCD reside in sub-Saharan Africa. SCA/SCD is considered as a substantial morbidity and has a mortality rate of about 5-16% among children whose age lies within 5 years.

According to the data published in the official website of WHO [23], it is reported that about 5% of world population is believed to be carrying some traits of haemoglobin disorders, out of which SCA/SCD and thalassemia top the list.

After going through these data and statistics which are obtained from authorized sources, it can be concluded that the prevalence of SCA/SCD is a global health concern. Apart from sub-Saharan Africa, other countries which are reported to have a large number of SCD/SCA cases are India, Dubai, Mediterranean countries like Italy, Turkey and Greece and Spanish speaking areas of Western Hemisphere like Caribbean, Central and South America, etc. According to the data published in [24], it is stated that approximately 100000 Americans suffer from SCA/SCD and 1 in13 African-American infants are born with SCA traits. SCA/SCD is also prevalent among Hispanic-American population, 1 out of every 16300 babies of this origin suffer from SCD/SCA.

Apart from SCD/SCA, Iron deficiency anaemia, other erythrocytes-related disorders are hereditary elliptocytes (HE), acanthocytosis, neuroacanthocytosis, hypothyroidism, etc.

HE is most prevalent among the persons of Mediterranean and African descent. It is reported in [25] that the global incidence of this disease lies somewhere between 1 in 2000 to 1 in 4000 individuals. Many persons suffering from HE is asymptomatic. Erythrocytes of persons suffering from this disease changes from its' normal biconcave disc shape like structure to an ellipse like structure.

Acanthocytes (spur cells) and Echinocytes (burr cells) are two popular poikilocytes which are almost structurally similar to each other. Both of these cells are spiculated but the major difference in their morphology lies in their pattern. Acanthocytes are irregular shaped spiculated red blood cells and Echinocytes are regular shaped spiculated red blood cells. Pictorial representation of both Acanthocytes and Echinocytes are given in Fig.1.2.2.2.

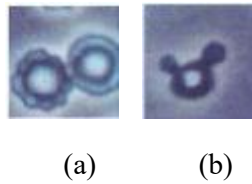


Fig.1.2.2.2. (a) Echinocytes and (b) Acanthocytes

Acanthocytes are associated with diseases like neuroacanthocytosis, hypothyroidism, splenectomy, malnutrition, etc. This type of poikilocyte is also present in the urine of diabetic patients who are suffering from glomerulonephritis (a type of kidney disease). It is also believed that the presence of acanthocytes in human blood is a significant cause of haemolytic anaemia as because of their irregular shapes, they are often trapped in spleen and get destroyed.

According to a report published in the website of National Organization for Rare Disorders [26], it is stated that a type of rare progressive disorders namely neuroacanthocytosis can change the normal biconcave disc shape like structure of erythrocytes to an irregular spiculated structure (acanthocytes). This disease is mainly associated with neurological abnormalities particularly movement disorders, muscle weakness, cognitive failure, etc. The severity of disabilities caused by this disease often increases with time if left untreated and may lead to loss of lives.

Echinocytes are also associated with uremia, stomach cancers and haemorrhagic ulcers, etc.

Dacrocytes which is also known as tear drop cells is a type of poikilocyte which is associated with granulomatous inflammation, infiltration of bone marrow, vitamin B12 deficiency, etc.

The above discussion clearly states the practical significance of this problem statement. A novel feature ensemble method is proposed in this thesis and implemented to generate a feature ensemble comprising of most relevant features derived from multiple deep neural networks to perform automated detection of nine different types of erythrocytes whose morphological details are given in Fig.1.2.2.3.










No	Cell type	Cell description	Cell structure
1	Normal	Biconcave shaped cell	
2	Elliptocytes	Elongated or oval shaped cell	
3	Sickle Cells	Crescent shaped cell	
4	Acanthocytes	Spiculated cells with irregularly spaced projections of variable length	
5	Echinocytes	Spiculated cells with evenly spaced projections of similar length	
6	Dacrocytes	Tear drop shaped cell	
7	Schistocytes	Fragmented RBC	
8	Spherocytes	Sphere shaped cell	
9	Stomatocytes	Erythrocytes with a slit-like central pallor	

Fig.1.2.2.3 Description of erythrocytes which are detected using the proposed method

1.3. Land cover classification using full-polarimetric image data

1.3.1. Land cover

Land cover refers to the physical material present on Earth's surface. It may be vegetation, urban area, water, etc.

1.3.2. Synthetic Aperture Radar

Synthetic Aperture Radar (SAR) is an imaging radar which generates 2D/3D/multi-dimensional images of land cover by mapping back-scattered electromagnetic waves into a finite number of resolution cells of fixed dimension.

The imaging technique of SAR is explained as follows:

SAR initially transmits a series of electromagnetic waves of certain polarization (horizontally polarized/vertically polarized) towards the target regions (the regions whose satellite images are to be generated). These waves hit the surface of the target area at certain incident angle and a part of the incident waves are reflected back to the radar depending upon the orientation, roughness and moisture content of the land cover present in the target area. These reflected waves are also referred to as back-scattered waves. The polarization of the reflected waves can be similar to that of the incident wave or can vary in accordance to the characteristics of the land cover present in the target area.

From the above discussion, it becomes quite evident that the characteristics of land covers can be interpreted from intensities of images generated by mapping the back-scattered waves into a finite number of resolution cells of fixed dimension in generated SAR images.

1.3.3. Advantages of SAR over conventional radars with physical antenna

In radar imaging field, there is a general rule that irrespective of the type of radar (i.e., whether the radar possess synthetic aperture or it is a conventional radar possessing physical antenna), the resolution of the images generated by them are directly proportional to the size of the aperture of the radar. As SAR is mounted on moving platforms like aircrafts and spacecrafts, it covers large distance

over target regions within specified time required for the transmitted waves to get reflected back to the radar and thus creates a large synthetic aperture with much less resources compared to conventional radar. This makes SAR a cost-effective way of generating high-resolution satellite images. In addition to this, the other advantages which SAR has over conventional radars is weather independence i.e., the images produced by SAR are independent of weather condition prevailing in the target area during the time when they are captured. The images provided by SAR are also day and night independent.

1.3.4. Characteristics of images generated by SAR

Initially, SAR is used to generate grayscale images which are referred to as SAR images. An example of SAR image is given in Fig.1.3.4.1 below:



Fig.1.3.4.1 An example of SAR image

Radar images are quite different from optical images mainly because of the differences in their imaging techniques. This makes radar images much more complex for human interpretation. However, to ease the interpretation process, some thumb rules [27] are proposed like:

- **The higher the back-scattered intensity, the rougher is the surface being imaged.**

This rule is created based on the observation that the transmitted waves are reflected more by the surfaces having higher roughness compared to smoother surfaces and based on the characteristics of back-scattered waves reflected from those surfaces, pixels corresponding to the areas (areas having comparatively higher soil roughness) are marked as higher intensity pixels in comparison to pixels used for marking smoother surfaces.

The intensities of SAR images also give an estimation of soil moisture as target areas possessing wet soil results in higher back-scattering compared to target areas possessing dry soil. Flooded areas on the other hand results in lower back-scattering and thus appears as dark areas in images.

In the era of automation, manual interpretation of any object, image, etc. has become quite obsolete not only because that process is tedious as well as time-consuming but it is also subjected to human error which largely effects the accuracy of the process.

For this reason, in the present era neural networks are largely used for performing image classification. As these neural networks are designed to replicate the cognitive functioning of human brain, the features which are most attractive to human cognitive system are also believed to be ideal features to be given as inputs to the neural networks.

Among all the features like colour, shape, size, orientation, edge, etc., the human cognitive system is more sensitive to color feature compared to other features. Since, SAR images do not provide colour information, in this thesis Polarimetric Synthetic Aperture Radar (POLoSAR) images and the features generated from them are chosen as inputs to the neural networks designed for performing image

classification over SAR images as POLSAR images provide colour information in addition to the information provided by SAR images. These images are produced by exploiting the polarimetry property in SAR. An example of AIRSAR L Band Flevoland POLSAR image and its corresponding Ground Truth (GT) image are given in Fig.1.3.4.2 below.

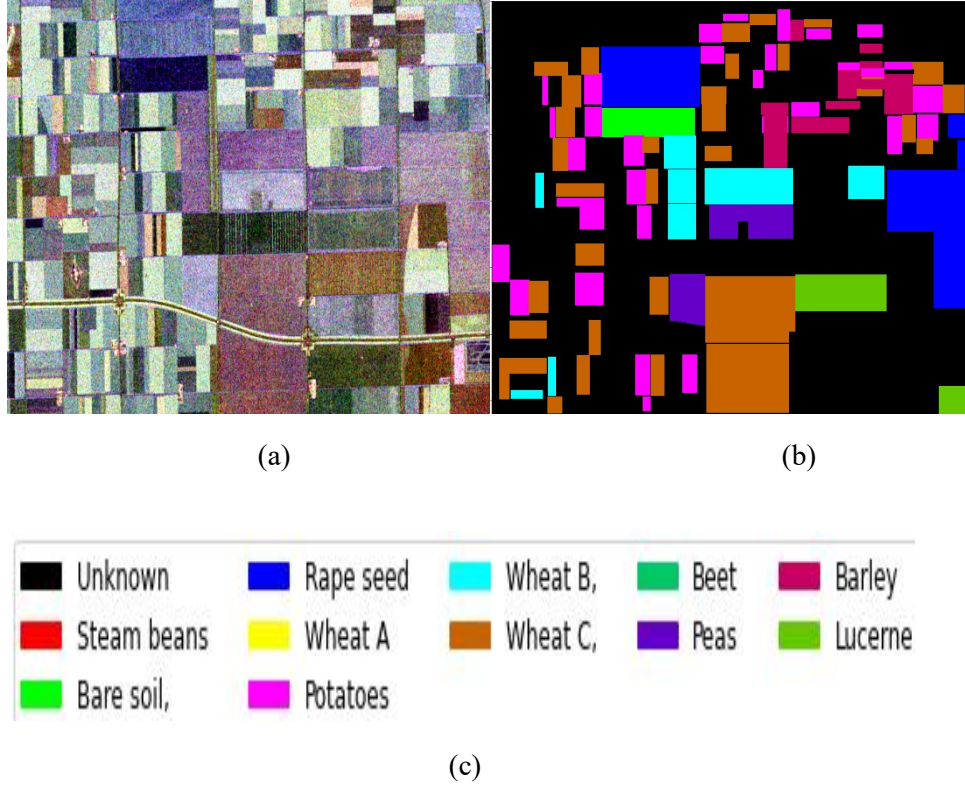


Fig.1.3.4.2. (a) POLSAR image of AIRSAR L Band Flevoland data, (b) Ground Truth image of (b) and (c) Class labels

Images generated by SAR undoubtedly play crucial role in various remote sensing applications like land cover monitoring, crop monitoring, oil spill detection, etc. but these images are inherently corrupted with multiplicative speckle noise which appears as black and white dots on images. The presence of speckle noise not only degrades the quality of images but it often leads to erroneous interpretation. Speckle noise is caused due to coherent nature of SAR.

As discussed earlier, SAR generates images by mapping the back-scattered waves into a finite number of resolution cells of fixed dimension, Hence, a large number of back-scattered waves having different relative phases are mapped into one resolution cell. These back-scattered waves add among themselves either constructively or destructively depending upon their relative phases which lead to the presence of black and white dots on images. Mathematically, a speckle noise corrupted image is represented as follows:

$$I_{speckle-corrupted}(i, j) = R_{speckle-free}(i, j)\eta(i, j) \quad (1.3.4.1)$$

In (1.3.4.1), the terms ' $I_{speckle-corrupted}(i, j)$ ' and ' $R_{speckle-free}(i, j)$ ' represent intensities of noisy (speckle noise corrupted) and noise-free (speckle-free) images at pixel position, (i, j) respectively while the term ' $\eta(i, j)$ ' denotes the intensity of speckle noise present at that pixel position.

The nature of speckle noise is quite different from other types of noise like impulse noise, Gaussian noise as it is not introduced in images due to unwanted modification of input signals caused by external disturbances like camera distortion, channel inference, etc. Instead, it is a signal itself which varies in accordance to the fluctuation of relative phases of back-scattered waves.

Despeckling (speckle noise removal) of images generated by SAR is absolutely necessary prior to using them as inputs in any kind of computer vision tasks like image classification as images corrupted with speckle noise largely enhances the chances of obtaining erroneous results.

At present there are several existing despeckling filters which are designed by several authors to perform speckle noise removal from images. After going through the filtering strategies proposed in the existing works, it is noted that the existing filters have several limitations. A comprehensive survey on the filtering strategies proposed in the existing works is carried out in Chapter 10, where the strengths and limitations of each filter are also highlighted. Taking into account the limitations of the existing filters, a novel adaptive despeckling filter is designed in this thesis to overcome the shortcomings of existing filters and to perform effective speckle noise filtering. The performance efficiency of the proposed filter is validated using several images.

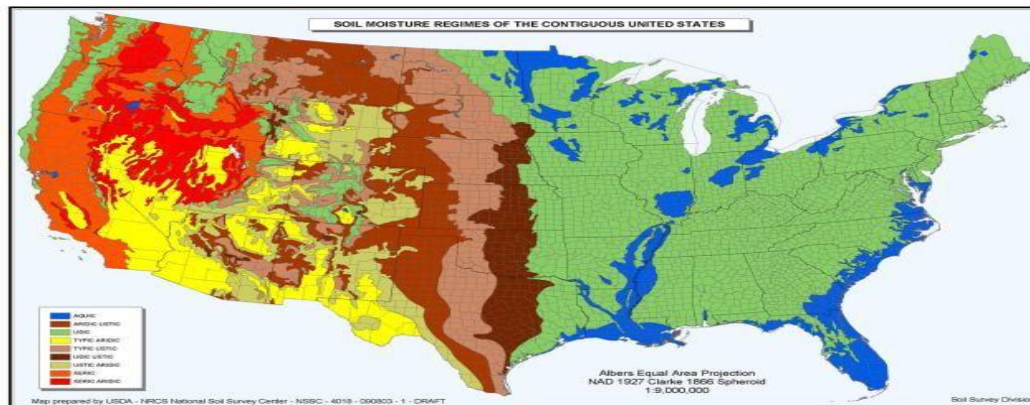
The detailed description of the designed despeckling filter and the comparative qualitative and quantitative results obtained by applying the proposed filter as well as several well-known existing filters on similar set of data are given in Chapter 11.

A novel Convolutional Neural Network (CNN) namely, Crop-Net is designed in this thesis to perform land cover classification. The CNN is termed as Crop-Net as it is mainly designed to classify the different types of crops present in farmlands. A detailed discussion on the architecture of CNN and its performance efficiency is carried out in Chapter 12.

1.3.5. Significance of land cover classification in real world

Land cover classification is very essential to detect the topology of the target area which in turn is needed for various important applications like crop monitoring, oil spill detection, generation of thematic maps, etc.

Thematic map is a special-purpose statistical map which is designed to visualize a specific theme (which may be soil moisture, climate, etc.) over a certain geographic area. These type of maps are created for analyzing the present condition of a particular theme (crop productivity, climate, etc.) over a geographic area so that the social, environmental, health, agricultural and economic development projects over that area can be planned accordingly. Examples of thematic maps are given in Fig.1.3.5.1.



(a)

Fig.1.3.5.1. Examples of thematic maps designed for Soil moisture

1.4. Introduction to the principles of Computational Intelligence

1.4.1. Computational Intelligence

Although there is no standard definition of Computational Intelligence, it can be termed as the ability of any computing system to learn about certain tasks based on the knowledge derived from the data and experimental observations.

1.4.2. Principles of Computational Intelligence

The five basic principles of Computational Intelligence are:

- Fuzzy Logic
- Neural networks
- Evolutionary Computation
- Learning Theory
- Probabilistic method

Among all the principles of Computational Intelligence in this thesis, Fuzzy Logic and an Evolutionary algorithm namely, Bacterial Foraging optimization algorithm are used for performing image dehazing and image filtering. Another principle namely, Neural Networks particularly Deep Neural Networks are used for performing land cover classification and detection of nine different types of erythrocytes having varying morphology.

1.4.3. Basics of Conventional Logic and Fuzzy Logic

The idea of Fuzzy Logic [28] was first conceived by Zadeh in the year 1965, has proved itself to be a useful tool in solving several engineering problems having immense practical significance. Its' capability of representing human logic in the form of several rules namely, Fuzzy Inference (FI) rules comprising of antecedent and consequent parts make Fuzzy Logic an ideal tool for solving various real-life problems.

Prior to starting discussion on Fuzzy sets, brief details of Conventional sets and its' disadvantages are given in Section 1.4.3.

Sets are defined to be collection of objects which are termed as members or elements [29].

For e.g., if a set contain the exam marks of students, it can be referred to as **EXAM MARKS** set.

Let Y be a set and y is a member/element of that set, then it is mathematically expressed as:

Suppose there is another set say, P which contain an element/member say p . Let all the elements/members of the set P are also members of the set Y , then in such case the set P is termed as a subset of the set Y .

If in such case where all the elements/members present in P are members of the set Y , but all the elements/ members of Y are not the elements/members of P , then P is called a proper subset of Y .

In other case, where all the elements/members present in P are members of the set Y and all the elements/ members of Y are the elements/members of P , then P and Y are termed as equal.

In other case, if P and Y are two different sets but there is a common element say, x which simultaneously exists in both of these sets.

In Conventional sets, the demarcations between set boundaries are very rigid which often lead to improper allocation of the elements. In real world, it is very difficult to allocate objects in sets having such rigid boundaries. This disadvantage of Conventional sets is explained with an example as follows:

Suppose there is an Universal set say, HEIGHT and SHORT, MEDIUM and TALL are the subsets of the Universal set HEIGHT. The boundaries of subsets are defined as:

SHORT= $\{height \in \text{HEIGHT}: 0 \text{ ft} \leq height < 5 \text{ ft}\}$,

MEDIUM= $\{height \in \text{HEIGHT}: 5 \text{ ft} \leq height < 6 \text{ ft}\}$,

TALL= $\{height \in \text{HEIGHT}: 6 \text{ ft} \leq height < 10 \text{ ft}\}$,

In this example, $height$ is variable which can take on any value within the closed interval of [0ft 10ft]. Referring to the above demarcations between the sets SHORT, MEDIUM and TALL, it can be stated that a $height$ of 4.99 ft will be a member of the set SHORT while a height of value 5 ft is a member of

the set MEDIUM. According to Conventional Logic, the member of any set has a membership of 1.0 only in that set and membership of 0.0 in other sets.

$$\mu_{SHORT}(height = 4.99 ft) = 1 \quad (1.4.3.1.7)$$

$$\mu_{MEDIUM}(height = 4.99 ft) = 0 \quad (1.4.3.1.8)$$

Hence, these sharp demarcations between the sets are often imprecise and it leads to wrong allocation of the elements in real world.

In 1965, Fuzzy Logic is designed by Zadeh to solve this disadvantage of Conventional Logic as well as to represent the reasoning ability of human cognitive system in form of several FI rules and thus helps to solve several real-life problems .

Fuzzy Logic extends the binary membership of Conventional Logic $\{0,1\}$ into a spectrum of $[0 1]$. Unlike Conventional Logic where the members of Universal set are members of one of its' subsets, in Fuzzy Logic the members of Universal set are the members of all of its' subsets.

$$0 \leq \mu_{SHORT}(height) \leq 1 \quad (1.4.3.1.9)$$

In this example, all the members of Universal set, HEIGHT are members of Fuzzy set, SHORT.

Unlike Conventional set, Fuzzy Logic allows any *height* value lying within the interval $[0ft 10ft]$ to be a member of all the Fuzzy sets SMALL, MEDIUM and TALL with varying membership values lying between $[0 1]$. This is explained in Fig.1.4.3.1.

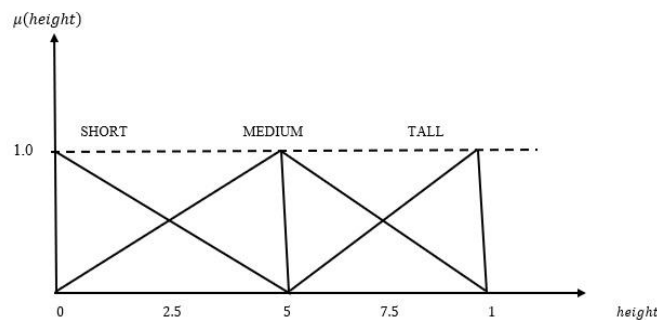


Fig.1.4.3.1. Fuzzy membership curves representing Fuzzy sets SHORT, MEDIUM and TALL

From Fig.1.4.3.1, it can be noticed that any value of height is a member of all three Fuzzy sets namely SHORT, MEDIUM and TALL but have varying membership values. For e.g., *height* value of 5ft has a membership of 1.0 in Fuzzy set, MEDIUM but has membership of less than 1.0 in Fuzzy sets, SHORT and TALL.

In the above figure, Triangular membership Function is used to represent the membership of Fuzzy sets but other than Triangular membership Function, there are also other membership functions which can be used to represent Fuzzy sets. Membership functions are represented in the following Section both mathematically as well as graphically .

1.4.3.1. Membership functions used for representing Fuzzy sets

a. Y Function

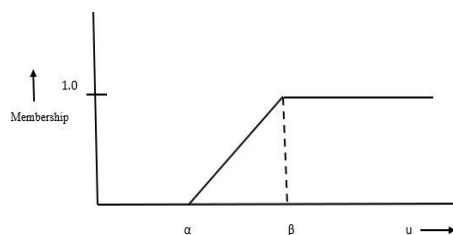


Fig.1.4.3.2.1. Graphical representation of Y function [29]

$$Y(u; \alpha, \beta) = \begin{cases} 0, & u \leq \alpha \\ \frac{u-\alpha}{\beta-\alpha}, & \alpha < u \leq \beta \\ 1, & u > \beta \end{cases} \quad (1.4.3.1.1)$$

b. s Function

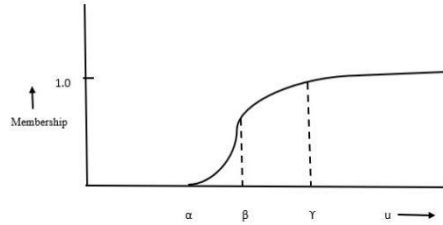


Fig. 1.4.3.2.2. Graphical representation of s function [29]

$$s(u; \alpha, \beta, \gamma) = \begin{cases} 0, & u \leq \alpha \\ 2[(u-\alpha)/(\beta-\alpha)]^2, & \alpha < u \leq \beta \\ 1 - 2[(u-\beta)/(\gamma-\beta)]^2, & \beta < u \leq \gamma \\ 1, & u > \gamma \end{cases} \quad (1.4.3.1.2)$$

c. L Function

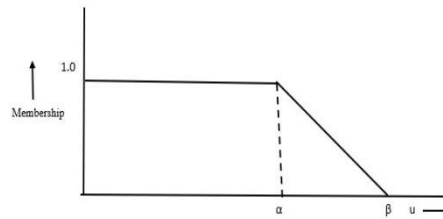


Fig.1.4.3.2.3. Graphical representation of L function [29]

$$L(u; \alpha, \beta) = \begin{cases} 1, & u \leq \alpha \\ \frac{\alpha-u}{\beta-\alpha}, & \alpha < u \leq \beta \\ 0, & u > \beta \end{cases} \quad (1.4.3.1.3)$$

d. Triangular membership Function

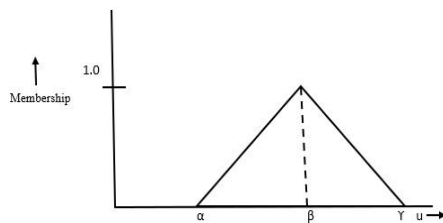


Fig.1.4.3.2.4. Graphical representation of Triangular membership function [29]

$$\Lambda(u; \alpha, \beta, \gamma) = \begin{cases} 0, & u \leq \alpha \\ (u - \alpha)/(\beta - \alpha), & \alpha < u \leq \beta \\ (\alpha - u)/(\beta - \alpha), & \beta < u \leq \gamma \\ 1, & u > \gamma \end{cases} \quad (1.4.3.1.4)$$

e. Π Function

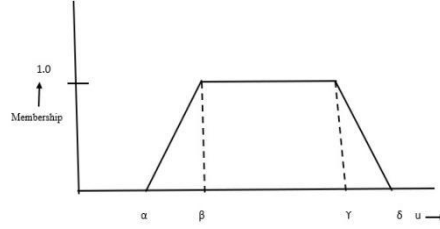


Fig.1.4.3.2.5. Graphical representation of Π function [29]

$$\Pi(u; \alpha, \beta, \gamma, \delta) = \begin{cases} 0, & u \leq \alpha \\ (u - \alpha)/(\beta - \alpha), & \alpha < u \leq \beta \\ 1, & \beta < u \leq \gamma \\ \frac{\gamma - u}{\delta - \gamma}, & \gamma < u \leq \delta \\ 0, & u > \delta \end{cases} \quad (1.4.3.1.5)$$

f. Gamma Function

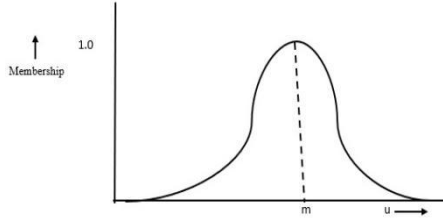


Fig1.4.3.2.6. Graphical representation of Gamma function [29]

$$G(u; m, \sigma) = \exp \left[- \left\{ \frac{u - m}{\sqrt{2\sigma}} \right\}^2 \right] \quad (1.4.3.1.6)$$

1.4.3.2. Fuzzy sets

Fuzzy set is mathematically defined as follows [2]:

$$X = \{(x, \mu_X(x)) : x \in U\} \quad (1.4.3.2.1)$$

In (1.4.3.3.1), U represents the Universal set while X denotes a fuzzy set. The term $\mu_X(el)$ is used to represent the membership value of the element, el in fuzzy set, X .

1.4.3.2.1. Representation of Fuzzy linguistic variable

Fuzzy linguistic variable is usually represented using four tuples $\langle el, LV, DR, \mu \rangle$

Where, el : name of fuzzy linguistic variable (For e.g., *height* in the above example)

LV : the linguistic values which the fuzzy linguistic variable say el , can take. (For e.g., {SHORT, MEDIUM TALL} in the above example)

DR : Dynamic range of fuzzy linguistic variables (For e.g., [0ft 10ft] in the above example)

μ : membership value of fuzzy linguistic variables in Fuzzy sets included within the LV tuple.

1.4.3.3. Intersection and Union operations on Fuzzy sets

1.4.3.3.1. Intersection (Fuzzy T-Norm) operation

The intersection of two fuzzy sets X and Y under a Universal set, U is characterized by T-norm operator. Mathematically, it is defined as follows:

$$\mu_{X \cap Y}(x) = T(\mu_X(x), \mu_Y(x)) \quad (1.4.3.3.1.1)$$

Considering p, q, s and t as membership values, the following T-norm operations can be mathematically defined as follows:

$$\begin{aligned} T(0,0) = 0, T(p, 1) = T(1, p) = p & \quad (\text{boundary}) \\ T(p, q) \leq T(s, t) \text{ if } p \leq s \text{ and } q \leq t & \quad (\text{monotonicity}) \\ T(p, q) = T(q, p) & \quad (\text{commutativity}) \\ T(p, T(q, s)) = T(T(p, q), s) & \quad (\text{associativity}) \end{aligned}$$

1.4.3.3.2. Union (Fuzzy S-Norm) operation

The union of two fuzzy sets X and Y under a Universal set, U is characterized by S-norm operator. Mathematically, it is defined as follows:

$$\mu_{X \cup Y}(x) = S(\mu_X(x), \mu_Y(x)) \quad (1.4.3.3.2.1)$$

Considering p, q, s and t as membership values, the following S-norm operations can be mathematically defined as follows:

$$\begin{aligned} S(1,1) = 1, S(a, 0) = S(0, a) = a & \quad (\text{boundary}) \\ S(p, q) \leq S(s, t) \text{ if } p \leq s \text{ and } q \leq t & \quad (\text{monotonicity}) \\ S(p, q) = S(q, p) & \quad (\text{commutativity}) \\ S(p, S(q, s)) = S(S(p, q), s) & \quad (\text{associativity}) \end{aligned}$$

1.4.3.4. Fuzzy Control Systems

In general, Fuzzy Control Systems are of two types:

- a. Mamdani Fuzzy Control System
- b. Takagi-Sugeno Fuzzy Control System

1.4.3.4.1. Mamdani Type Fuzzy Control System

PR_j : If x_1 is A_1^j and x_2 is A_2^j and x_m is A_m^j Then y is B_j .

For $j=1$ to n

x_i : i – th input linguistic variable with $i = [1 \ m]$

A_j^i : i – th input Fuzzy set with $i = [1 \ m]$

y : output linguistic variable

B_j : output Fuzzy set for $j = [1 \ n]$

n : number of production rules in the knowledge base

In this thesis, Mamdani Type novel Fuzzy Controller Systems are designed to perform image dehazing, image filtering, edge and noise detection, intensity based sky detection, etc.

PR_j is the j^{th} production rule included in the Knowledge Base of the Fuzzy Controller System whose block diagram is given in Fig. 1.4.3.4.1.

In a Fuzzy Inference rule / Production rule, ' x_1 is A_1^j and x_2 is A_2^j and x_m is A_m^j ' portion is termed as Antecedent part (AM) and ' y is B_j ' part is termed as Consequent part (CM).

1.4.3.4.1.1. Steps of Mamdani Type Fuzzy Controller System

PR_j: IF x_1 is A_1^j and x_2 is A_2^j and and x_m is A_m^j THEN y is B_j .

x_i : i^{th} input fuzzy linguistic variable with $i = [1 \ m]$

A_i^j : i^{th} input fuzzy linguistic value with $i = [1 \ m]$

y : output fuzzy linguistic variable

B_j : output fuzzy linguistic variable for $j = [1 \ n]$

n : Number of FI rules present in the knowledge base.

MAMDANI TYPE FUZZY CONTROL SYSTEM:

Step 1: $AM_j = t(\mu_{A_1^j}^j(x_1), \mu_{A_2^j}^j(x_2), \dots, \mu_{A_m^j}^j(x_m)) = \min(\mu_{A_1^j}^j(x_1), \mu_{A_2^j}^j(x_2), \dots, \mu_{A_m^j}^j(x_m))$

t : T norm operator, AM_j : Antecedent Membership of the j^{th} rule.

Step 2: $CM_j = \mu_{B_j}(y)$ CM_j : Consequent Membership of the j^{th} rule.

Step 3. $R_j(AM, CM) = \mu_{R_j}(AM_j, CM_j) = \min(AM_j, CM_j)$

μ_R denotes the membership value to belong to the fuzzy relation R .

Here, $\mu_{R_j}(AM_j, CM_j) = \min(AM_j, CM_j)$ as Mamdani implication relation function is considered.

Step 4. $x_1^*, x_2^*, \dots, x_m^*$ {Crisp values of fuzzy input linguistic variables}

Following fuzzification corresponding membership values of these fuzzy input linguistic variables

$\mu_{A_1^j}^j(x_1^*), \mu_{A_2^j}^j(x_2^*), \dots, \mu_{A_m^j}^j(x_m^*)$

Step 5. $AM^j = t(\mu_{A_1^j}^j(x_1^*), \mu_{A_2^j}^j(x_2^*), \dots, \mu_{A_m^j}^j(x_m^*)) = \min(\mu_{A_1^j}^j(x_1^*), \mu_{A_2^j}^j(x_2^*), \dots, \mu_{A_m^j}^j(x_m^*))$

Step 6. $\mu_{B_j}^j(y) = (AM^j \circ R_j)$ for $j = [1 \ n]$

Symbol ' \circ ' is used here to represent Max-Min Composition operator

Step 7. $\mu_B(y) = \max_{j=1}^n (\mu_{B_j}^j(y))$

Step 8. $y^* = \frac{\sum_{y \in Y} (y \times \mu_B(y))}{\sum_{y \in Y} \mu_B(y)}$ {Defuzzification is done using **Centre of Area method**}

' \forall ' symbol is used to denote Universal operator.

1.4.3.4.1.2. Centre of Area defuzzification method

$$y^* = \frac{\sum_{y \in Y} y \times \mu_B(y)}{\sum_{y \in Y} \mu_B(y)} \quad (1.4.3.4.1.2.1)$$

In (1.4.3.5.2.1), y^* denotes defuzzified output value.

1.4.3.4.2. Takagi-Sugeno Type Fuzzy Control System

PR_j: If x_1 is A_1^j and x_2 is A_2^j and x_m is A_m^j Then $y_j = f_j(x_1, x_2, \dots, x_m)$

For $j=1$ to n

In most cases, $y_j = \sum_{i=1}^m p_{ji} x_i$

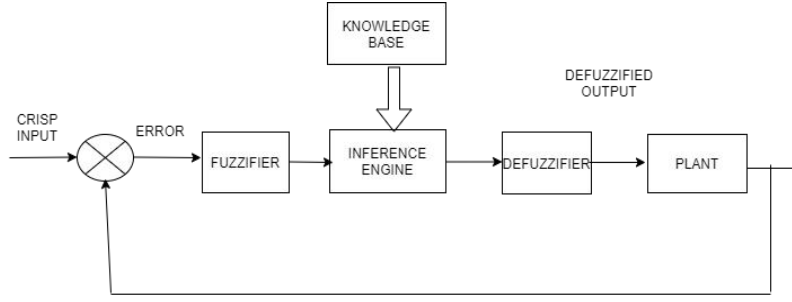


Fig.1.4.3.4.1. Block diagram of Fuzzy Controller System [29]

1.4.4. Evolutionary Computation

This principle of Computational Intelligence refers to algorithms whose designs are inspired by the biological evolution theory to find the optimized solutions of engineering problems.

These algorithms are designed taking into account the behaviour of species like ants (Ant Colony optimization algorithm), cuckoos (Cuckoo search algorithm), etc.

In this thesis an Evolutionary algorithm namely, Bacterial Foraging optimization (BFO) algorithm [30] is used to perform the selection of most suitable edge-direction within each image patch in combination with a novel set of FI rules. The detailed description of the proposed method is given in Chapter 3.

BFO algorithm is designed based on the foraging characteristics of E.Coli bacteria. Foraging behaviour of any species is termed as an optimization process through which they reach the region having the highest food concentration. Social foraging or group foraging is always considered to be better compared to individual foraging as it involves collective intelligence and thus helps in movement of bacteria towards the region having high food concentration.

E.Coli bacteria make only two kinds of movements namely, a. Tumble and b. Swim. These movements of E.Coli bacteria is controlled by the concentration of the food in the current location. Whenever bacterium reach a neutral region (i.e., a region having neither high food concentration nor is the region noxious), they tumble. Bacteria mostly swim in regions having positive nutrient gradient and tumble in regions having negative nutrient gradient.

The entire life cycle of E.Coli bacteria can be divided into four stages namely,

- a. Chemotaxis
- b. Swarming
- c. Reproduction
- d. Elimination and dispersal

Some terms which are used to denote the movement of bacteria in BFO algorithm are:

N_b : Total number of bacteria

$J(\theta^i), \theta^i \in R^{D_m}$: The cost function used for representing nutrient concentration in R^{D_m} (search space) of dimension D_m .

$\theta^i(p, q, r)$: The position of the i^{th} bacterium in R^{D_m} (search space) after p^{th} chemotactic, q^{th} reproductive and r^{th} elimination-dispersal steps.

The movement of bacteria is determined by the solution of the cost function $J(\theta^i)$ as the bacteria always move towards the region having high food concentration. A brief description of each stage of the life cycle of E.Coli bacteria is given as follows:

- a. **Chemotaxis:** This movement of bacteria is usually determined by the nutrient concentration in R^{D_m} , which is estimated using cost function, $J(\theta^i)$. Bacteria usually move in random direction with a step

size, $C(i)$. Whenever bacteria reach a region having constant high nutrient concentration, they continue to swim in that region by taking a step of size, $C(i)$, until they reach a region having comparatively less nutrient concentration or exceeds the maximum number of swims (N_s). If bacteria reach a less nutrient rich region or if N_s along same direction exceed, then the bacteria tumble and reach a new location. This movement is mathematically defined as follows:

$$\theta^i(p + 1, r, q) = \theta^i(p, q, r) + C(i) \frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}} \quad (1.4.4.1)$$

In (1.4.4.1), $\Delta(i)$ represents the random vector which is used to determine the direction of movement after tumble.

b. Swarming: E.Coli bacteria communicate with each other using chemical signalling, instead of performing cell-to-cell communication. This characteristics of E.Coli bacteria is replicated using certain repellent and attractant parameters.

c. Reproduction: In this stage of life-cycle, each of a pre-determined number of healthy bacteria say, N_{br} split and generate two bacteria. The newly generated daughter bacteria are also placed in the same position as that of the mother bacteria. Normally, N_{br} is considered to be half of the total population of bacteria (N_b). In order to make the population of bacteria constant, the number of healthy bacteria is always considered to be equal to the number of unhealthy bacteria which die in elimination step.

d. Elimination and dispersal: Elimination step regulates bacteria population by eliminating the bacteria which reach areas with less nutrient content or noxious areas. Dispersal step deals with the movement of few bacteria to less explored regions in order to find the true global optimum.

The utilization of BFO algorithm in performing edge&noise detection within image patches is described elaborately in Chapter 2.

1.4.5. Neural Networks

These types of networks are designed to artificially replicate the learning behaviour of human cognitive system. These networks are composed of several layers (input layer, hidden layer and output layer) which in turn consist of several nodes which are artificial replica of biological neurons (learning units of human brain).

These types of networks have the ability of establishing the interconnection between input data and outputs by leaning their pattern without the need of any explicit programming like Fuzzy Logic.

In practical applications for e.g..electronic appliances where Fuzzy Logic is used say washing machine, the time required for washing, soaking and drying clothes is determined by a Fuzzy Controller whose knowledge base consists of a set of *If-Then* rules. These rules are created by experts considering all the possible cases which may arise while washing clothes. Relevant inputs like weight of clothes, amount of water needed, etc. which determine the total time required for washing, soaking and drying cloths are considered as input fuzzy linguistic variables of the *If-Then* rules present in the knowledge base of the Fuzzy Controller system designed for this purpose.

In this case, there is a need of explicit programming by experts as Fuzzy Logic does not support inherent learning but in case of neural networks there is no need of explicit programming as these networks have inherent leaning capability.

Learning can be of two types:

A. When labels of input data are known. In this case, the neural networks are trained with labeled data so that those networks can establish the inter-connection between input data and outputs by studying the relation between input patterns and their corresponding outputs. This type of learning is called supervised learning which involves transfer of human knowledge in form of labeled data to the networks. Classification is done using labeled data by using the supervised learning strategy.

B. When labels of input data are not known. In this case, clustering of data are done based on the similarities between the data. This type of learning is called unsupervised learning which is quite

popular in real world as the data available in real world are mostly unlabeled data. Clustering is done using unlabeled data by using the unsupervised learning strategy.

1.4.5.1. Basic structure of Neural Network

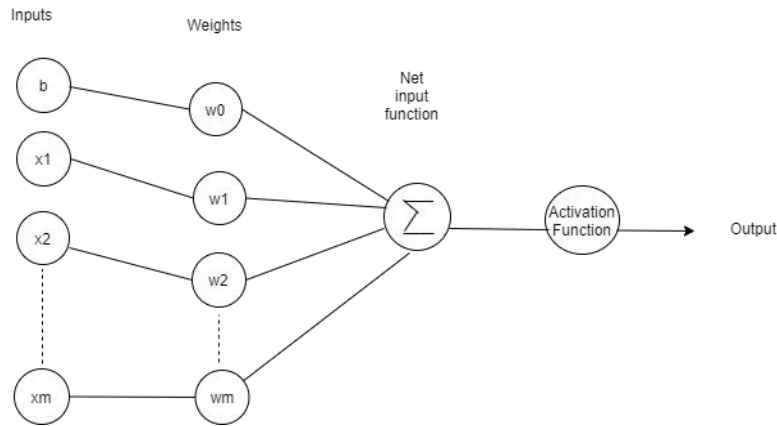


Fig.1.4.5.1.1. A simple feedforward Neural Network

Feedforward Neural Network is a type of network which contains no feedback path. The neural network shown in Fig 1.4.5.1.1 has no hidden layers. It comprises of a simple input layer and output layer. The bias and the inputs $\{b, x_1, x_2, \dots, x_m\}$ are connected to the output node using separate weighted connections. These weights represent the strengths of the connections and help in estimating the influence of neurons on outputs.

The output is calculated using the following formula:

$$y(\text{output}) = f(b.w_0 + x_1.w_1 + x_2.w_2 + \dots + x_m.w_m) \quad (1.4.5.1.1)$$

$$y(\text{output}) = f(bw_0 + \sum_{i=1}^m x_i.w_i) \quad (1.4.5.1.2)$$

The term 'f' in (1.4.5.1.1)-(1.4.5.1.2) represents activation function. The firing of neurons are dependant on thresholds of activation functions.

Activation functions can be linear as well as non-linear. Some examples of both linear as well as non-linear activation functions are given in Fig. 1.4.5.1.2.


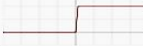







Name	Plot	Equation	Derivative
Identity		$f(x) = x$	$f'(x) = 1$
Binary step		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x \neq 0 \\ ? & \text{for } x = 0 \end{cases}$
Logistic (a.k.a Soft step)		$f(x) = \frac{1}{1 + e^{-x}}$	$f'(x) = f(x)(1 - f(x))$
TanH		$f(x) = \tanh(x) = \frac{2}{1 + e^{-2x}} - 1$	$f'(x) = 1 - f(x)^2$
ArcTan		$f(x) = \tan^{-1}(x)$	$f'(x) = \frac{1}{x^2 + 1}$
Rectified Linear Unit (ReLU)		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Parameteric Rectified Linear Unit (PReLU) [2]		$f(x) = \begin{cases} \alpha x & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} \alpha & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Exponential Linear Unit (ELU) [3]		$f(x) = \begin{cases} \alpha(e^x - 1) & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} f(x) + \alpha & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
SoftPlus		$f(x) = \log_e(1 + e^x)$	$f'(x) = \frac{1}{1 + e^{-x}}$

Fig.1.4.5.1.2. Some examples of activation function [222]

As methods are designed in this thesis mainly to perform image classification, so the discussions here are focused mostly on the utilization of Neural Networks in performing image classification.

Image classification using a simple machine learning algorithm for e.g. Support Vector Machine can be broadly divided into four steps as discussed here:

- a. Data collection
- b. Feature extraction
- c. Feature selection
- d. Classification

Here all these four steps are discussed using an example which involves classification of cars and bicycles for better understanding.

a. Data collection

In this step, images of cars and bicycles having varied characteristics are collected as data samples to train, validate and test the network. Higher the number of balanced samples (the number of samples collected for each class are more or less equal) collected in this step, more is the chance of getting accurate results as it guarantees better training.

An example of each type of data present in the collected database are given in Fig.1.4.5.1.3.



(a)



(b)

Fig.1.4.5.1.3. Sample data (set 1) of each class

b. Feature extraction

Although Neural Networks are designed to artificially replicate the behaviour of human cognitive system but in reality the human cognitive system is far more smarter compared to Neural Networks as it enables humans to perform complex tasks with much more ease.

For e.g., in face recognition task, humans can easily differentiate the variations in between the faces of persons irrespective of all of them having two eyes, one nose, two ears and one lip.

But how does this happen?

This complex task of recognition of human faces having wide variety of features appears to be very simple to human cognitive system as it involves the computation of the differences of facial features of any person from a standard reference instead of identifying the features themselves as these differences can capture the inter-variation between the facial features of different persons effectively which makes the recognition task much more easier.

The features which are chosen to classify bicycles and cars are given in Table 1.4.5.1.1.

Table1.4.5.1.1. Sample features (set 1)

Features	Car	Bicycle
Colour	Blue (for e.g. [0.55 0.64])	Red (for e.g. [0.21 0.43])
Number of Wheels	4	2
Accommodation	4	1

c. Feature selection

This step involves the selection of most significant as well as relevant features among the extracted features to improve the accuracy of the classification tasks.

Like in this example, the features namely, ‘Colour’, ‘Number of wheels’ and ‘Accommodation’ given in Table.1.4.5.1.1 can efficiently perform classification of sample data (given in Fig.1.4.5.1.3) as the numerical values of those features sharply vary for different classes.

But if the same features to classify the sample data are used (given in Fig.1.4.5.1.4), then it often lead to erroneous classification as in this case, the numerical value of the first feature (i.e. Colour feature) overlaps (as shown in Table 1.4.5.1.2) for objects belonging to different classes.



(a)



(b)

Fig.1.4.5.1.4. Sample data (set 2) of each class

Table 1.4.5.1.2. Sample features (set 2)

Features	Car	Bicycle
Colour	Blue (for e.g. [0.55 0.64])	Blue (for e.g. [0.55 0.64])
Number of Wheels	4	2
Accommodation	4	1

In this case, the second and third features namely, ‘Number of Wheels’ and ‘Accommodation’ are much more significant and relevant compared to the first feature (‘Colour’) as their numerical values sharply differ in case of distinct objects.

Hence, if in this case instead of using three features, the second and third features are used for performing classification, it reduces the chances of erroneous classification.

The procedure of selecting the most significant and relevant features from a set of extracted features is called feature selection.

In this thesis, a novel feature selection method namely, Ranking method is proposed which performs feature selection using two parameters namely, a. Fuzzy Entropy [31] and b. a newly designed Total Contribution Score parameter.

The details of the proposed feature selection method is given in Chapter 12 and description of the selection parameters are given in Chapter 9.

d.Classification

In this stage, the classification between distinct objects are done using machine learning algorithms. Some popular machine learning algorithms are Support Vector Machine [32], Extreme Machine Learning algorithm [33], etc.

In conventional machine learning algorithms, there is a need of handcrafted feature extraction. The handcrafted features are fed as inputs to the networks along with their corresponding labels to train the networks for performing classification. As the handcrafted features are computed by human experts hence, it is often subjected to human error which increases chances of erroneous classification.

The discrepancies which arise while performing classification using handcrafted features are explained in details with an example in Chapter 8.

This shortcoming of conventional machine learning algorithms is solved by deep learning algorithms which has inherent feature extraction capability.

The inherent feature extraction capability of deep neural networks increases the accuracy of the classification tasks as it does not require any handcrafted features to be fed as inputs. This capability of deep neural networks have make deep neural networks the most suitable choice of the researchers for performing classification tasks because of their overwhelming performances.

Although there are various types of deep neural networks but the most popular among these networks are Convolutional Neural Network (CNN). This type of neural networks are widely used for performing image classification tasks. A brief introduction to CNN architecture is given in the following Section.

1.4.5.2. Brief introduction to the architecture of CNN

The organization of layers (which comprise of several nodes) in CNN architecture is analogous to the organization of neurons in human visual cortex.

As discussed earlier, CNN does not need any handcrafted features to perform image classification, instead it effectively analyses the temporal and spatial dependencies of pixels in images by applying suitable filters at each convolutional layer which help in better image understanding.

Basic architecture of a CNN (given in Fig.1.4.5.2.1) comprises of the following layers:

- a. Convolutional layer
- b. Pooling layer
- c. Fully Connected layer
- d. Classification layer

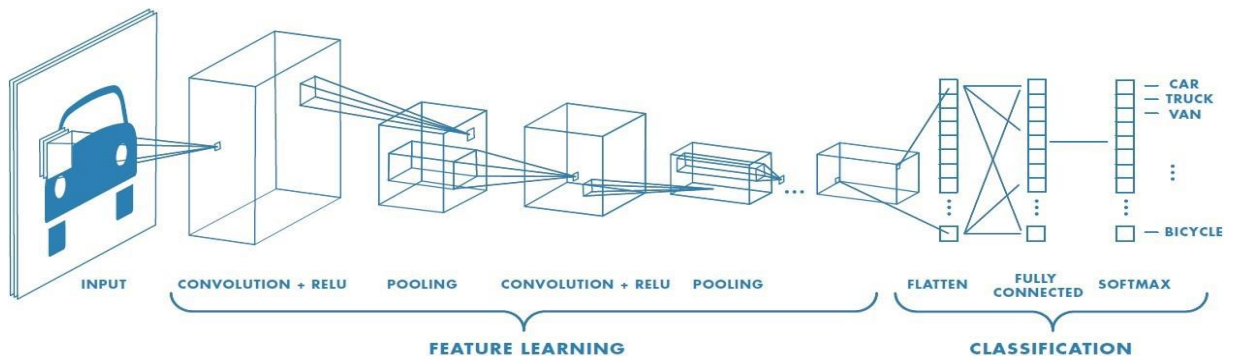


Fig.1.4.5.2.1. Basic architecture of CNN [223]

In Fig.1.4.5.2.1, there are some blocks which are labelled as CONVOLUTION+RELU. These blocks comprise of convolutional layer followed by Rectified Linear Unit (ReLU) layer. The convolutional layer comprises of various filters and extract features from images given as inputs to those layers by applying those filters on patches generated from the input images.

1.4.5.2.1. Description of CNN layers

A. Convolutional layer

Unlike human beings, computing systems visualize any image as a matrix of 2D or 3D dimension depending upon whether the image is a grayscale image or a color image. When images are given as inputs to CNN, it extracts features from the images using appropriate filters. The filters are chosen according to the tasks to be performed like for edge detection one can use the Laplacian filter, etc.

In a CNN architecture, there are several convolutional layers which extract both low-level features and high-level features. The initial convolutional layers extract low-level features while the convolutional layers located at higher depth extract high-level features.

The feature extraction procedure carried out in convolutional layers of CNN architecture is explained here with examples.

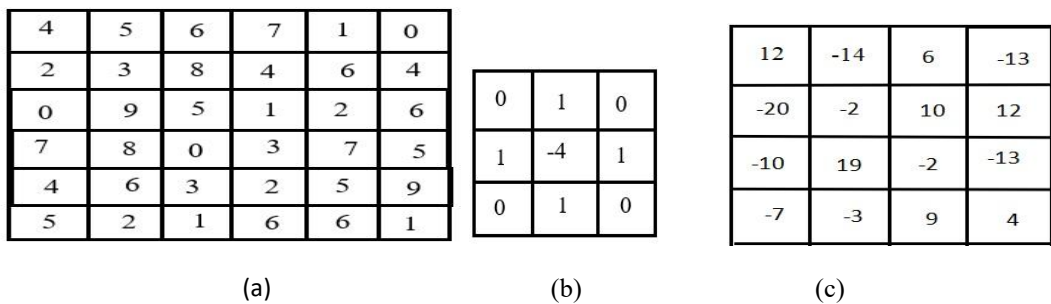
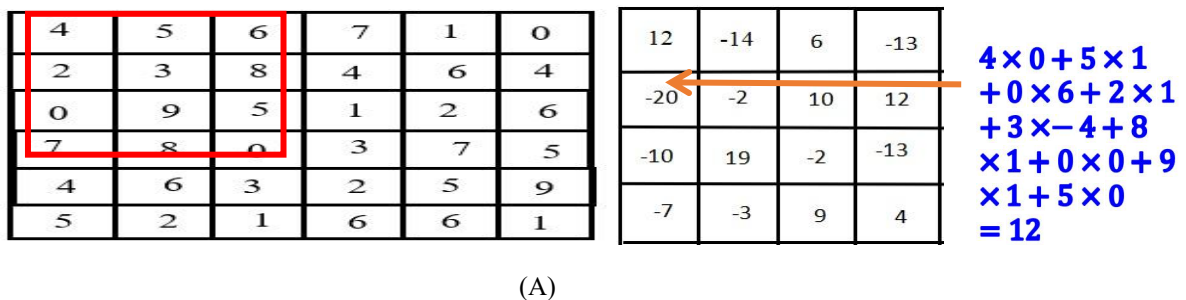
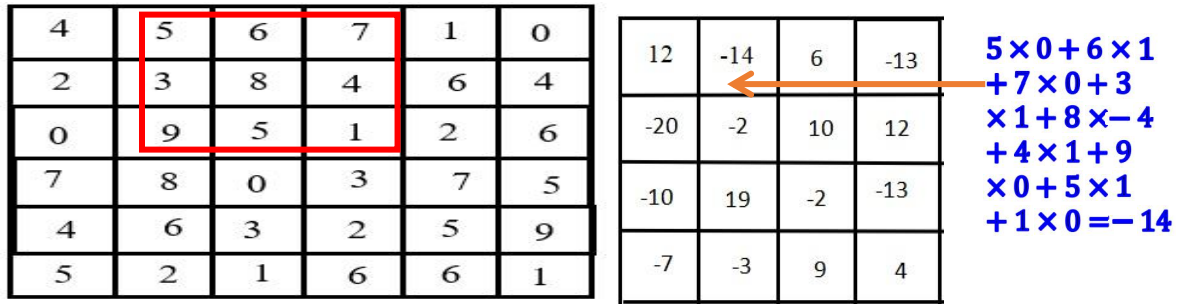


Fig.1.4.5.2.1.1. (a) 6 × 6 × 1 image, (b) 3 × 3 Laplacian mask and (c) Generated feature map of 4 × 4 × 1 dimension





(B)

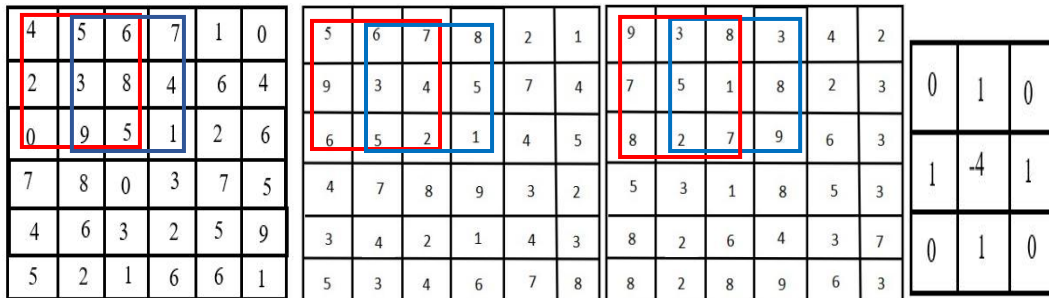
The other elements in the feature map are also computed in similar fashion.

Fig.1.4.5.2.1.2. Examples of pixel-wise feature value computation

In the above example, the dimension of the input image is $6 \times 6 \times 1$, the dimension of the filter kernel is 3×3 , so the dimension of feature map is $(\text{height of the input image} - \text{height of the filter kernel} + 1) \times (\text{width of the input image} - \text{width of the filter kernel} + 1) \times 1 = (6 - 3 + 1) \times (6 - 3 + 1) \times 1 = 4 \times 4 \times 1$.

In this example, the stride value is considered to be 1 as the filter window slides by one pixel position.

The example considered in Fig.1.4.5.2.1.1. and Fig. 1.4.5.2.1.2. is a 2D image, but if the image is a color image, then in such case feature map is computed as discussed below:



(a) Red Channel

(b) Green Channel

(c) Blue Channel

(d) Filter kernel

$$\begin{aligned}
 \text{Red Channel} &= (0 \times 4) + (1 \times 5) + (0 \times 6) + (1 \times 2) + (3 \times -4) + (1 \times 8) + (0 \times 0) + (1 \times 9) + (5 \times 0) = 5 + 2 - 12 + 8 + 9 = 12 \\
 \text{Green Channel} &= (0 \times 5) + (1 \times 6) + (7 \times 0) + (9 \times 1) + (-4 \times 3) + (1 \times 4) + (6 \times 0) + (1 \times 5) + (2 \times 0) = 6 + 9 - 12 + 4 + 5 = 12 \\
 \text{Blue Channel} &= (0 \times 9) + (3 \times 1) + (0 \times 8) + (1 \times 7) + (5 \times -4) + (1 \times 1) + (0 \times 8) + (2 \times 1) + (0 \times 7) = 3 + 7 - 20 + 1 + 2 = -7 \\
 \text{Sum1} &= \text{Red Channel} + \text{Green Channel} + \text{Blue Channel} + \text{Bias}(= 1) = 12 + 12 - 7 + 1 = 18
 \end{aligned}$$

$$\begin{aligned}
 \text{Red Channel} &= (5 \times 0) + (1 \times 6) + (7 \times 0) + (3 \times 1) + (8 \times -4) + (1 \times 4) + (9 \times 0) + (1 \times 5) + (1 \times 0) = 6 + 3 - 32 + 4 + 5 = -14 \\
 \text{Green Channel} &= (6 \times 0) + (7 \times 1) + (8 \times 0) + (1 \times 3) + (4 \times -4) + (5 \times 1) + (5 \times 0) + (1 \times 2) + (1 \times 0) = 7 + 3 - 16 + 5 + 2 = 1 \\
 \text{Blue Channel} &= (3 \times 0) + (8 \times 1) + (3 \times 0) + (5 \times 1) + (1 \times -4) + (8 \times 1) + (2 \times 0) + (7 \times 1) + (9 \times 0) = 8 + 5 - 4 + 8 + 7 = 24 \\
 \text{Sum2} &= \text{Red Channel} + \text{Green Channel} + \text{Blue Channel} + \text{Bias}(= 1) = -14 + 1 + 24 + 1 = 12
 \end{aligned}$$

Sum1	Sum2		

Fig..1.4.5.2.1.3. Feature map computation of a color image

B. Pooling layer

This layer is used to diminish the dimension of the computed feature map by keeping only the relevant features to reduce the computational burden.

Pooling can be of three types namely:

a. Max Pooling:

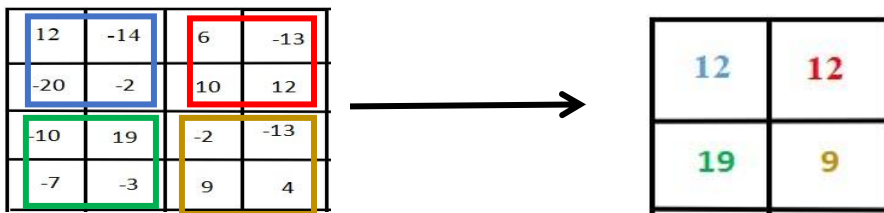


Fig.1.4.5.2.1.4. Feature map obtained after Max Pooling

b. Average Pooling:

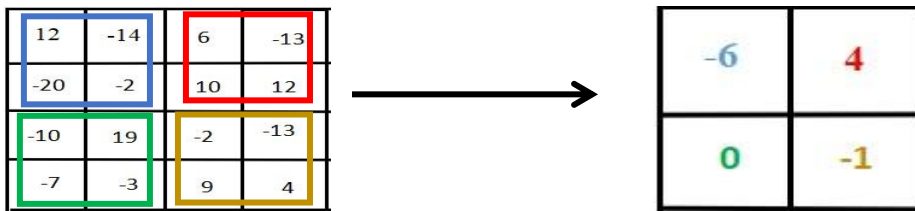


Fig.1.4.5.2.1.5. Feature map obtained after Average Pooling

c. Sum Pooling

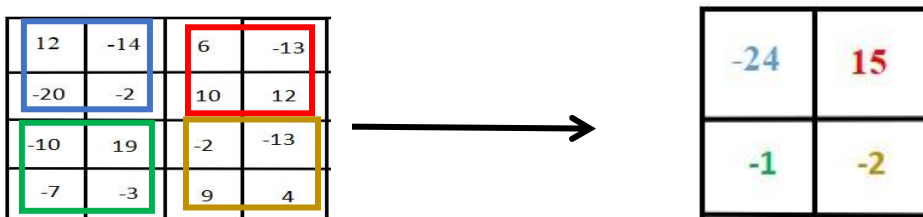


Fig.1.4.5.2.1.6. Feature map obtained after Sum Pooling

Max Pooling is most popular pooling operation compared to Average Pooling and Max Pooling as besides reducing the dimension of feature maps by choosing most informative features, it also performs effective noise suppression.

But Average Pooling as well as Sum Pooling performs better than Max Pooling in some applications. Hence, the choice of the type of Pooling layer to be included in the architecture of any CNN entirely depends upon the application for which the CNN is designed.

C. Fully Connected layer

The feature map which is obtained as the output of the Pooling layer is flattened into a column vector and fed as an input to the Fully Connected layer which learns the non-linear combination of features.

D. Classification layer

In general, Softmax classifier is generally used as a Classification layer in CNN architecture. The number of nodes in Classification layer is equal to the number of output classes. Softmax classifier is also known as the Maximum Entropy Classifier. Mathematically, Softmax classifier is defined as:

$$P(y = j | \theta^{(i)}) = \frac{e^{\theta_j^{(i)}}}{\sum_{k=0}^K e^{\theta_k^{(i)}}} \quad (1.4.5.2.1.1)$$

In (1.4.5.2.1.1), $\theta = w_0.x_0 + w_1.x_1 + w_2.x_2 + \dots + w_k.x_k = \sum_{i=0}^k w_i.x_i = w^T x$

1.5. Thesis Organization

This thesis focuses on three real-life problems :

- Daytime and Nighttime image dehazing
- Detection of structurally variant erythrocytes
- Land cover classification using full-polarimetric image data

For simplicity in understanding the presentation of this thesis, it is broadly divided into three divisions where each division is dedicated to each problem statement.

The first part of this thesis is dedicated to “Daytime and Nighttime image dehazing” problem. This part comprises of **Chapter 2 to Chapter 7**.

In **Chapter 2**, comprehensive literature survey on existing daytime and nighttime image dehazing methods is conducted and the limitations of each existing method is highlighted.

Detailed description of novel Bacterial Foraging (BF)-Fuzzy synergism based daytime image dehazing method which is proposed in this thesis to overcome the drawbacks of existing daytime dehazing methods is given in **Chapter 3** and comparative qualitative and quantitative analyses of outputs obtained from the proposed method as well as several state-of-the-art methods using several benchmark databases are carried out in **Chapter 4** of this thesis to validate the efficiency of the proposed method.

Refined Color Channel Transfer prior is proposed in this thesis to enable the daytime image dehazing methods to perform haze removal from nighttime hazy images efficiently alongside daytime hazy images. This prior is generally introduced as a pre-processing step of daytime image dehazing methods. Detailed description of the proposed prior is given in **Chapter 5** and its efficiency over existing priors is proved by performing subjective and objective analyses of outputs obtained from the methods in the same Chapter.

Several benchmark databases comprising of hazy images possessing varied characteristics as well as their corresponding GT images are created in this thesis focusing on various aspects of image dehazing problem. Details of the databases, their applications and significance are discussed in **Chapter 6** of this thesis. Some examples of images included in these databases are also given in that Chapter.

A novel Fuzzy Logic based sky segmentation method is designed in this thesis by exploiting the analogy between the sky segmentation problem and the image dehazing problem. Detailed description of the proposed sky segmentation method is given in **Chapter 7** of this thesis.

Second part of the thesis is dedicated to “Detection of structurally variant erythrocytes” which comprises of **Chapter 8** and **Chapter 9**. Literature survey of this problem statement is given in **Chapter 8** and detailed description of the novel feature ensemble creation method proposed in this

thesis to solve the information loss problem inherent in traditional CNN networks is given in **Chapter 9** of this thesis. The efficiency of the proposed feature ensemble method is validated in the same Chapter by performing detection of nine different types of erythrocytes using feature ensemble formed by the proposed method.

Third part of the thesis is dedicated to “Land cover classification using full-polarimetric image data” problem. This part of the thesis comprises of **Chapter 10** to **Chapter 12**.

Comprehensive survey on filtering strategies and limitations of existing despeckling filters and architectures of existing deep neural networks designed to perform POLSAR image classification is conducted in **Chapter 10** of the thesis.

A novel despeckling filter namely, Degree of purity & Scattering diversity based Advanced Lee filter is designed in this thesis to solve the drawbacks of existing filters. The despeckling strategy of the designed filter is discussed elaborately in **Chapter 11** of this thesis. The efficiency of this filter is validated by performing qualitative and quantitative analyses of outputs obtained from the designed filter as well as from various well-known filters in the same Chapter.

The architecture of Crop-Net, the deep neural network designed to perform POLSAR image classification in this thesis is described in **Chapter 12**. A novel feature selection technique namely, Ranking method which is proposed in this thesis to perform selection of features having high information content and significant contribution in classification to perform POLSAR image classification is also described in this Chapter. The excellence of Crop-Net over existing networks is validated in terms of Test Accuracy, Precision, Recall and F1 score in **Chapter 12**.

The thesis is concluded in **Chapter 13** where the future scope of the work is also highlighted.

**DAYTIME & NIGHTTIME
IMAGE
DEHAZING**

Chapter 2

Literature survey

Image dehazing has emerged as a popular field of research over the years because of its practical significance (as discussed in details in **Section 1.1.2 of Chapter 1**). These real-life significance of the image dehazing problem have inspired numerous researchers to design various methods to perform effective dehazing of degraded images which are captured during hazy weather condition.

In this Chapter, brief details of each of these existing image dehazing methods are given and their respective shortcomings are highlighted.

This Chapter begins with a comprehensive survey on image dehazing methods which are designed based on characteristics of daytime images followed by brief discussion on the challenges faced by these methods in performing effective dehazing of nighttime hazy images. The discussion also focuses on the inefficiency of atmospheric scattering model (1.1.1) (described in **Section 1.1 of Chapter 1**) in accurately representing nighttime hazy images.

This discussion is followed by a detailed description of mathematical models which are designed by various researchers for accurately representing nighttime hazy images and the methods which are designed for effectively dehazing nighttime hazy images.

2.1. Three crucial aspects of image dehazing

Three important aspects of image-dehazing are:

- Contrast Enhancement
- Edge-preservation (edge-detection & edge-sharpening)
- Noise removal

2.2. Brief description of daytime image-dehazing methods

The daytime image dehazing methods whose methodologies are discussed in this Section are designed based on the atmospheric scattering model (1.1.1). The discussion begins with detailed description of methods which perform image dehazing considering information derived from multiple images of the same image scene.

The authors in [34]-[35] have performed dehazing using information derived from multiple images of the same scene captured under different weather conditions whereas in [36]-[37], the authors have performed dehazing using information derived from the differences between images of the same scene captured using different polarization angles. These methods fail to satisfactorily remove haze from images corrupted with dense haze as polarized light is not the major degradation factor in these images. Moreover, obtaining multiple images of the same scene is not always practically possible which often make the real-life implementation of the methods proposed in [34]-[37] not feasible.

These **limitations** of multi-image based image dehazing methods lead to the need of designing new methods which can perform dehazing using information derived from single image only, thereby making the ill-posed nature of the image dehazing problem much more challenging.

The discussion on such image dehazing methods begins with a detailed description of dark channel prior (DCP) method [38] as it is one of the most popular existing image dehazing methods proposed till date and several researchers have designed numerous image dehazing methods based on this prior.

2.2.1. Dark Channel Prior

The authors in [38] have proposed this prior based on their observation that in non-sky patches of haze-free, daytime images there is at the minimum one color channel which contains low intensity pixels ('dark pixels') while analyzing the histograms of non-sky regions of 5000 daytime, haze-free images of landscapes and cityscape which they have collected from Flickr.com. The authors have cited the presence of colorful objects, trees, leaves, etc. in those images as the primary reason behind the occurrence of these dark pixels. They have termed that color channel as the 'dark channel' from where the name of the prior is derived.

Here the discussion on DCP begins with detailed description of the properties of daytime haze-free images based on which this prior is designed followed by the details of each step of DCP and the challenges which arise while performing nighttime image dehazing using DCP. The techniques adapted by the authors who have designed their nighttime image dehazing methods based on DCP concept to deal with these challenges are also explained in Section 2.2.4.5 .

DCP comprises of the following steps:

- Dark channel evaluation
- Atmospheric light estimation
- Transmission estimation
- Transmission refinement
- Scene radiance recovery

Step 1: Dark channel evaluation

Ideally, the dark channel of any haze-free, daytime image comprises of very low intensity pixels (75% of the pixels have intensities approximately equal to zero and about 90% of the pixels have intensities below 25) but the dark channel of corresponding hazy image usually possess pixels with comparatively higher intensities due to the influence of airlight. Higher the haze-concentration of image-patches, more will be the influence of airlight on those patches and higher will be the intensities of pixels belonging to those patches and vice versa. So, the dark channel of any hazy image gives a rough approximation of its patch-wise haze-concentration.

Mathematically, the dark channel of any haze-free image, say J , is evaluated as follows:

$$J^{dark}(x) = \min_{y \in \Omega(x)} \left(\min_{c \in \{r, g, b\}} J^c(y) \right) \quad (2.2.1.1)$$

The authors in [38] have stated that ideally the values of dark channel of J should be approximately equal to zero as stated in (2.2.1.1) because of the presence of these dark pixels.

$$J^{dark}(x) = \min_{y \in \Omega(x)} \left(\min_{c \in \{r, g, b\}} J^c(y) \right) \approx 0 \quad (2.2.1.2)$$

Step 2: Atmospheric light estimation

The atmospheric light of any daytime hazy image is globally uniform as it is mainly contributed by sunlight whose intensity is spatially invariant and color is white (as it is a combination of several monochromatic lights). So, the authors in [38] estimated the atmospheric light of an image by averaging the intensities of 0.1% of the brightest pixels belonging to its dark channel assuming that these pixels will always lie in the sky regions of daytime images as the pixels belonging to the sky regions of daytime images usually possess higher intensities compared to the pixels lying in the non-sky regions due to the presence of sunlight. But this assumption fails in the presence of white surfaces/objects in images leading to erroneous atmospheric light estimation.

Step 3: Transmission estimation

Following atmospheric light estimation, the authors estimated transmission as stated in (2.2.1.4) which is obtained by substituting the term ' $\min_{y \in \Omega(x)} \left(\min_{c \in \{r, g, b\}} \frac{I^c(y)}{A^c} \right)$ ' in (2.2.1.3) by '0' (reason stated in (2.2.1.2)) and introducing the haziness factor, ' ω ' in (2.2.1.4) as stated below:

$$\min_{y \in \Omega(x)} \left(\min_{c \in \{r, g, b\}} \frac{I^c(y)}{A^c} \right) = \widetilde{t}(y) \min_{y \in \Omega(x)} \left(\min_{c \in \{r, g, b\}} \frac{J^c(y)}{A^c} \right) + 1 - \widetilde{t}(y) \quad (2.2.1.3)$$

$$\widetilde{t}(y) = 1 - \omega \min_{y \in \Omega(x)} \left(\min_{c \in \{r, g, b\}} \frac{I^c(y)}{A^c} \right) \quad (2.2.1.4)$$

The haziness factor ' ω ' is introduced in (2.2.1.4) to control the amount of haze to be removed from images. Its value lies between [0 1] where the values closer to '1' signify better haze removal. The authors in [38] have fixed it at 0.9 to keep little amount of haze in output images as it is necessary to preserve the depth information as well as naturalness of images which helps observers to distinguish between far and nearby objects.

Step 4: Transmission refinement

The output haze-free images obtained using transmission maps estimated in Step 3, often contains halos and blocking artifacts in some image-patches where intra-patch transmission variations occur. To overcome this limitation, the authors have introduced this additional transmission refinement step in DCP to refine the initially estimated transmission maps using the soft matting algorithm [39].

Step 5: Scene radiance recovery

The authors in [38] generated final output haze-free images in this step using estimated atmospheric light and refined transmission values using the following equation:

$$J(x) = \frac{I(x) - A}{\max(\widetilde{t}(x), t_0)} + A \quad (2.2.5)$$

The constant ' t_0 ' is introduced in (2.2.5) to maintain the denominator at a value greater than zero to avoid the production of noisy outputs.

The authors have also stated that DCP can handle both sky and non-sky regions of images gracefully as colors and intensities of sky regions of daytime images and atmospheric light are almost similar.

Table 2.2.1.1. Assumptions considered in DCP

Assumptions	Daytime images	Nighttime images
Atmospheric light is globally uniform	✓	✗
Colors of sky regions of images are almost similar to that of atmospheric light	✓	✗
Pixels possessing high intensities mostly lie in sky-regions of images	✓	✗

2.2.1.1. Challenges which arise while performing nighttime image dehazing using DCP

- DCP performs atmospheric light estimation by averaging the intensities of 0.1% of the brightest pixels in images assuming that these pixels have high intensities in all color channels. This assumption is explicitly valid in case of daytime images as atmospheric light of images captured during daytime are mostly contributed by sunlight whose color is usually white and it thus possess high intensities in all color channels. But this assumption fails in case of nighttime images as atmospheric light of those images are mostly contributed by multicolored artificial light sources which emit blue light, yellow light etc. These lights do not possess high intensities in all color channels. For e.g. blue light possess high intensities in blue channel, yellow light possesses high intensities in green and red channels etc. This often leads to erroneous atmospheric light estimation.

- As DCP is designed following the atmospheric scattering model, it can neither handle the spatially varying illumination characteristics nor the glow characteristics of nighttime images which often lead to the production of color distorted output images containing halo-artifacts.
- The authors in [38] have specifically mentioned that DCP can handle both sky and non-sky regions of images gracefully as the colors of sky regions of daytime images are almost similar to that of their atmospheric light but this condition does not hold for nighttime images as colors of atmospheric light of such images vary spatially in accordance to the colors of light emitted by multicolored artificial light sources. Moreover, sky regions of nighttime images usually possess relatively low intensity pixels compared to their atmospheric light due to the absence of bright sunlight at night. These discrepancies often lead to the introduction of undesired distortions in output images.
- DCP fails to produce desirable results for images containing white objects (even in daytime images).

2.2.2. Daytime image dehazing methods

In this Section, brief details of the existing daytime image dehazing methods other than DCP are given and their respective shortcomings are highlighted.

In [40]-[42], the authors have proposed fusion based dehazing techniques. In [40], the authors have performed dehazing of any hazy image by fusing the image obtained after subtracting the haze layer from each channel of the hazy image with the image obtained after performing gamma correction and detail enhancement of the gray version of hazy image using local linear fusion.

Limitation of the method proposed in [40]: This method although accurately enhances regions of images possessing dense haze but degrades the contrast of the rest of the images as it performs contrast enhancement of all the pixels belonging to a hazy image identically using a constant gamma value irrespective of their haze content. Moreover, the detail enhancement method used in [40] also boosts the presence of noise in output images.

The authors in [41] have dealt with the drawbacks of the dehazing method proposed in [40] by fusing the input generated after white balancing the hazy image with the input generated by subtracting the average luminance of the hazy image from the hazy image itself using three weight maps (luminance, chromatic and saliency). These weight maps help in enhancing the contrast of each region of hazy image according to their nature of degradation. To eliminate the artifacts arising in output images due to sharp transitions in weight maps, the authors have fused the generated inputs and the weight maps level-wise after decomposing them into Laplacian and Gaussian pyramids respectively instead of combining them directly.

Limitation: This method is designed focusing only on the contrast enhancement aspect of image dehazing. It does not focus on the other important aspects of dehazing.

In [42], the authors have performed dehazing by fusing the high-frequency (HF) components of Red Green Blue (RGB) and Near Infrared (NIR) images of the same scene. This method is designed based on the observation that HF components of local patches of haze free RGB and NIR images of same scene are similar but HF components of local patches of hazy RGB and NIR images of same scene are different.

Limitation: The major hindrance behind the practical implementation of this method is acquiring RGB and NIR images of the same scene. Although the authors have tried to deal with this issue using the recovered R channel which is obtained by processing RGB hazy image as proposed in [38] as a substitute of NIR image, the output images thus obtained is texture-less.

Tarel et al. [43] have proposed a fast modified median filter based dehazing method.

Limitation: This method can preserve edges but only along obtuse angles.

In [44], the authors have introduced a weighted L1-norm and contextual regularization based dehazing method to decrease the halo-artifacts occurring in output images obtained from DCP [38].

Limitation: This method cannot restore true color in sky regions of images.

Xiao et al. [45] proposed a guided joint bilateral filter based scene depth recovery method for preserving the edge details of images.

Limitation: This method provides distorted results in the presence of sunlight.

The authors in [46] have performed dehazing by portraying it as a simple contrast enhancement problem. They performed contrast enhancement of hazy images by minimizing an objective function which consists of a constant term and an information loss term to prevent the production of over-saturated output images.

Salazar-Colores et al. [47] have performed dehazing by portraying it as a simple contrast enhancement problem. They have proposed a multi-layer perceptron based transmission estimation method to estimate transmission directly from the dark channel of image unlike DCP [38], which requires an additional transmission refinement step to perform final transmission estimation. This method produces output images with lesser halo-artifacts compared to DCP.

Limitations of the methods proposed in [46]-[47]: The authors have designed these methods by simply considering dehazing as a contrast enhancement problem. They have not taken into account the edge-preservation and noise removal aspects of dehazing.

Fattal [48] proposed a Markov Random Field based transmission estimation method by exploiting the property of color distribution within patches.

Limitation: This method cannot dehaze monochromatic images.

Following similar color line concept, in [49] the authors have performed transmission estimation based on their observation that colors in haze free images can be approximated by limited set of tight clusters in RGB color space while in hazy images each cluster becomes a line (haze line) and the position of each pixel on the line reflects its transmission level.

Limitation: This method fails to dehaze portions of images where airlight exceeds scene radiance.

Zhu et al. [50] have performed scene depth recovery of images using their proposed Color Attenuation Prior (CAP) in combination with the gradient descent algorithm.

Limitation: Although this method can accurately dehaze images possessing homogeneous haze but it fails to provide desired results for images having non-homogeneous haze distribution. Moreover, this method uses Sobel edge detector which often gives erroneous results. The shortcomings of this detector are discussed in Chapter 3.

In [51], the authors have estimated transmission required for performing image dehazing by training a Random Forest based regression model with haze-relevant features (Multi-scale dark channel, Multi-scale local max contrast, Hue disparity and Multi-scale local max saturation).

Limitation: This method boosts the presence of noise and artifacts in outputs of images corrupted with dense haze.

Cai et al. [52] proposed an end-to-end transmission estimation network, DehazeNet which estimates transmission using haze-relevant features which are extracted using the concepts of DCP and CAP proposed in [38] and [50]. Similarly, the authors in [53] introduced a Multi-scale Convolution Neural Network which also estimates transmission using haze-relevant features.

Limitation: The output results obtained from these methods contain halo-artifacts.

This shortcoming is somewhat overcome by the end-to-end dehazing network, AOD-Net [54].

Limitation of the methods proposed in [52-54]: These methods have performed dehazing by portraying it as a simple contrast enhancement problem, they have not taken into account the other crucial aspects of the dehazing problem.

The authors of the methods proposed in [55-57] have performed dehazing using the HSV color space.

Limitation of the methods proposed in [56-57]: The authors have performed dehazing considering it as a simple contrast enhancement problem.

Limitation of the method proposed in [57]: The authors have focused on the edge-sharpening aspect of dehazing but they have sharpened both the edges as well as noise present in images identically which gives rise to undesirable patterns in output images obtained from this method.

From the above discussion, it becomes quite evident that none of these existing daytime image dehazing methods focus on all the crucial aspects of dehazing (contrast enhancement, edge preservation and noise removal). So, there is a need of a method which gives equal importance to all these aspects besides solving the above-mentioned drawbacks.

2.2.3. Inefficiency of daytime image dehazing methods in performing nighttime image dehazing

The image dehazing priors like DCP and CAP which have inspired many researchers to design their methods based on the properties of daytime images. In DCP [38], the dark channel concept is derived from the observation that there must be one color channel which possess very low intensity pixels (after analyzing the histograms of clear daytime images). This prior also considers the fact that the sky regions of images possess higher intensity pixels compared to non-sky regions.

Although in nighttime images, there are color channels which contain very low intensity pixels but the fact that the sky-regions of images possess higher intensity pixels compared to non-sky regions is not valid in case of nighttime images mainly due to the absence of sunlight during night (as discussed in **Section 1.1.2 of Chapter 1**).

The CAP [50] is also designed based on the observation that haze-concentration of pixels can be estimated from the magnitudes of differences between their saturation and value channels in HSV color space. The images based on whose properties CAP concept is established in [50] are all clear daytime images.

All the fusion based methods, deep neural networks, optimization based methods, etc. discussed in **Section 2.2.2** are designed based on the properties of daytime images.

Moreover, as discussed in **Section 1.1.1 of Chapter 1**, the main difference between daytime and nighttime images lies in their illumination characteristics as illumination of daytime images are usually globally uniform mainly due to uniform distribution of sunlight whose color and intensity is invariant over the entire image scene but in case of nighttime images, the illumination is mainly contributed by artificial light sources whose colour and intensities vary spatially according to the nature of light sources.

The variation in the illumination properties of daytime and nighttime images makes the atmospheric scattering model (1.1.1) unsuitable for proper representation of nighttime images as it contains a constant atmospheric light term, 'A' which can represent only globally uniform characteristics of daytime images but cannot efficiently represent the non-uniform illumination characteristics of nighttime images.

To overcome this shortcoming of atmospheric scattering model, many mathematical models are designed to accurately represent nighttime images. These mathematical models are discussed in details in the next Section.

2.2.4. Nighttime image models

2.2.4.1. New imaging model [58]

Mathematically, this model is stated as follows:

$$I^c(x) = L(x)\eta^c(x)R^c(x)t(x) + L(x)\eta^c(x)(1 - t(x)) \quad (2.2.4.1.1)$$

The authors in [58] have termed this model as New imaging model as it is the first model proposed taking into account the spatially varying illumination characteristics of nighttime images. They have derived this model by replacing the constant atmospheric light term 'A' in (1.1.1) with ' $L(x)\eta^c(x)$ ' which enables it to perform point-wise environmental illumination estimation and thus

allows it to overcome the limitations of atmospheric scattering model to some extent. But like atmospheric scattering model, this model too does not contain any glow term and hence, it cannot represent the glow characteristics of nighttime images.

2.2.4.2. Nighttime imaging models (including the glow characteristics)

All three nighttime hazy image models discussed in this Section were designed taking into account both the spatially varying illumination characteristics as well as the glow characteristics of nighttime images. Here these models are discussed in details to show how the same characteristics of nighttime images were modeled using different perspective by various authors in their respective works.

2.2.4.2.1. Multiple light colors and glow model [59]

Mathematically, this model is represented as follows:

$$I(x) = J(x)t(x) + A(x)(1 - t(x)) + A_a(x) * APSPF \quad (2.2.4.2.1.1)$$

In contrary, to any of the previous models proposed in [1] and [58], this model contains an additional glow term to represent the glow characteristics of nighttime images. Similar to the work proposed by Narasimhan and Nayar [60], the authors have portrayed glow present in nighttime images as an atmospheric point spread function in their work as the appearance of glow becomes smoother as it decays with increasing scene distance. The reason behind the formation of glow in nighttime images is pictorially explained in Fig. 2.2.4.2.1.1. Apart from representing glow characteristics of images efficiently, this model also facilitates pixel-wise atmospheric light estimation which enables it to take into account the spatially varying illumination characteristics of nighttime images.

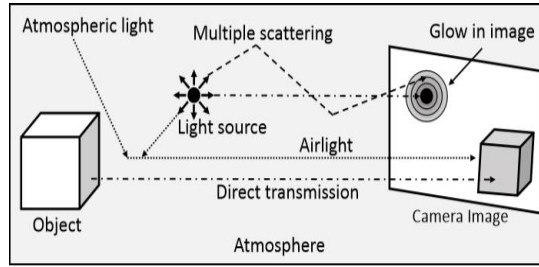


Fig. 2.2.4.2.1.1. Pictorial representation of nighttime imaging environment [59]

2.2.4.2.2. Nighttime haze model model [28]

This model is mathematically represented as follows:

$$I(x) = J(x)t(x) + A(x)(1 - t(x)) + \sum_{k=1}^n \widehat{S}_k(x) \widehat{G}(1) \quad (2.2.4.2.21)$$

Where

$$\widehat{G} = 1, \text{ for glow regions}$$

$$\widehat{G} = 0, \text{ for non - glow regions}$$

Unlike [59], the authors in [61] have represented glow as the summation of the products obtained by multiplying the shapes and illumination directions of glow sources with their corresponding glow regions (regions of images highly influenced by the glow sources). This type of glow representation enables the researchers to remove undesirable glow effects from glow regions of images only and thus facilitates the production of output images with good visual effects as it prevents over enhancement of non-glow regions.

2.2.4.2.3. Improved atmospheric scattering model [62]

The authors in [62] have designed this model by combining the total variation model [63] with the glow model. The total variation model states that any input hazy image, say I , is a combination of the structure layer and the texture layer where the structure layer comprises of the main scenes of I including its brightness and haze while the texture layer contains its detail and noise. Mathematically, the total variation model is represented as follows:

$$I(x) = St(x) + Te(x) \quad (2.2.4.2.3.1)$$

The authors in [62] have derived Improved atmospheric scattering model by further decomposing the structure layer of the total variation model into glow layer and background layer as shown in (2.2.4.2.3.2) to incorporate the glow characteristics of nighttime images within the model.

$$St(x) = St_A(x) + St_B(x) \quad (2.2.4.2.3.2)$$

Where

$$St_B(x) = \widehat{St}(x)t(x) + A(1 - t(x)) \quad (2.2.4.2.3.3)$$

$$St_A(x) = A_a(x) * APSF \quad (2.2.4.2.3.4)$$

$$St(x) = \widehat{St}(x)t(x) + A(1 - t(x)) + A_a(x) * APSF \quad (2.2.4.2.3.5)$$

This model defines output haze-free image as a combination of optimized structure layer and enhanced version of optimized texture layer as stated below:

$$J(x) = \widehat{St}(x) + \mu \widehat{Te}(x) \quad (2.2.4.2.3.6)$$

2.2.4.3. Relaxed atmospheric scattering model [64]

Mathematically, this model is represented as follows:

$$I(x) = J(x)t(x) + A(x)(1 - t(x)) \quad (2.2.4.3.1)$$

In contrary to the atmospheric scattering model [1] which is designed assuming the atmospheric light of an image to be always globally uniform, the authors in [64] have replaced the constant atmospheric light term of that model with a spatially varying atmospheric light term to efficiently handle non-uniform illumination characteristics of nighttime images. Hence, this model is termed as the relaxed atmospheric scattering model.

2.2.4.4. Nighttime hazy image model with HDP function [65]

This model simply represents output haze-free image, 'J' as the difference between hazy image 'I' and haze density map, 'I_h' as stated in (2.2.4.4.1). Unlike any of the above models, this model does not contain any transmission term and atmospheric light term. Hence, the methods designed following this model do not involve estimations of these parameters to perform image dehazing thereby reducing their computational burden and chances of producing distorted outputs which mainly occur due to erroneous estimations of these parameters.

$$J(x) = I(x) - I_h(x) \quad (2.2.4.4.1)$$

2.2.4.5. Survey on existing nighttime image dehazing methods

The above discussion clearly depicts the differences between nighttime hazy image models. So, the methods designed following different models also vary vastly from each other. Here comprehensive review of existing nighttime image dehazing methods are carried out after grouping them according to different image models following which they were designed as shown in Fig.2.2.4.5.1.

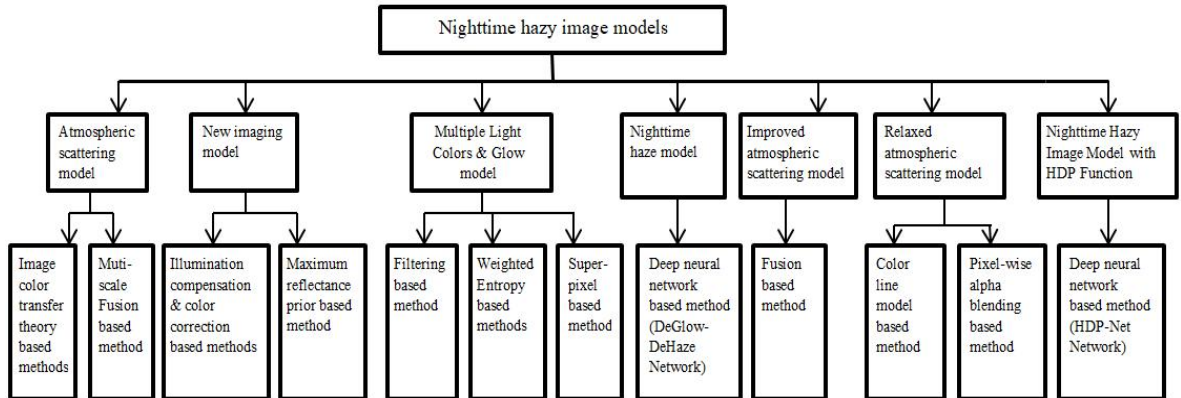


Fig.2.2.4.5.1. Model-wise grouping of nighttime image dehazing methods

2.2.4.5.1. Methods designed following Atmospheric scattering model

2.2.4.5.1.1. Image color transfer theory based methods

The nighttime image dehazing method proposed by Pei et.al.[66] is probably the first method designed exclusively for dehazing nighttime hazy images. In this work, the authors have performed image dehazing using a refined DCP method after transferring the “blue shift” airlight color of nighttime images to “grayish” colour. They have carried out this color transfer assuming the “grayish” color to be the usual airlight color of daytime images using the global color transfer method proposed in [67] considering nighttime hazy image as Source image and any daytime hazy image as Reference image in $l\alpha\beta$ color space.

The authors have selected $l\alpha\beta$ color space for performing color transfer because of its inherent capability of automatically compensating the chromatic loss occurring due to attenuation and scattering of scene light without estimating the chromatic loss directions. The l , α and β channels of $l\alpha\beta$ color space represent achromatic channel, chromatic yellow-blue and red-green opponent channels respectively. After converting the input image from RGB to $l\alpha\beta$ color space, they performed color transfer using the following set of equations:

$$\begin{aligned} l' &= \left(\frac{\sigma_r^l}{\sigma_s^l}\right) \times (l - M_s^l) + M_r^l \\ \alpha' &= \left(\frac{\sigma_r^\alpha}{\sigma_s^\alpha}\right) \times (\alpha - M_s^\alpha) + M_r^\alpha \\ \beta' &= \left(\frac{\sigma_r^\beta}{\sigma_s^\beta}\right) \times (\beta - M_s^\beta) + M_r^\beta \end{aligned} \quad (2.2.4.5.1.1.1)$$

The standard deviation ratios in (2.2.4.5.1.1.1) serve as key conversion coefficients in the color transfer process. The output image, I' thus obtained is converted back to RGB color space. The main aim of the authors in [66] behind performing this color transfer is to obtain an output color transferred image, I' possessing the scene characteristics of input nighttime image (Source image) and color characteristics of daytime image (Reference image) so that after performing color transfer the illumination characteristics of nighttime hazy images, I becomes globally uniform.



Fig.2.2.4.5.1.1.1. (a) Source image, (b) Reference image (with different scene contents), (c) Output image



Fig.2.2.4.5.1.1.2. (a) Source image, (b) Reference image (with almost same scene contents), (c) Output image

The output images (c) of both the figures (given above) prove that this global color transfer method cannot preserve the naturalness of images accurately.

Next to color transfer, the authors performed dehazing of I' using refined DCP method which is almost similar to traditional DCP method [38] but performs dark channel evaluation using pixel-wise operation instead of patch-wise operation stated in (2.2.1.1) to reduce halos and blocking artifacts occurring due to intra-patch transmission variations in [38] as follows:

$$J^{dark}(x) = \min_{y \in pixel(x)} \left(\min_{c \in \{r, g, b\}} J^c(y) \right) \quad (2.2.4.5.1.1.2)$$

It also performs transmission refinement using Guided Image Filter (GIF) [68] instead of soft-matting algorithm like DCP in order to achieve better edge-preservation.

The haze-free images thus obtained often suffer from poor contrast and color over-saturation. To overcome these drawbacks, the authors performed post-processing of output images thus obtained using the Bilateral filter in local contrast correction method [69].

Pros:

- This method produces output images with reduced halo-artifacts compared to the DCP method proposed in [38] as it performs pixel-wise dark channel evaluation instead of evaluating it patch-wise as done in [38] and thus overcomes the shortcomings arising due to intra-patch transmission variations to some extent.
- The use of GIF instead of soft matting algorithm for performing transmission refinement in this method also enhances its edge-preservation capability compared to the DCP method.

Cons:

- The global color transfer process used in this method cannot preserve the color characteristics of images accurately as shown in Fig.2.2.4.5.1.1.1 and Fig.2.2.4.5.1.1.2 which often leads to the introduction of undesired color distortions in output images.
- The color transfer process used in this method performs color transfer between images globally without taking into account the local variations in scene characteristics of Source images which often leads to the loss of important edge details.
- The differences between the scene contents and imaging conditions of Source and Reference images acts as the major reason behind obtaining distorted outputs.

Jiang *et al* [70]: In contrary to the method proposed in [66], the authors in [70] carried out color transfer between Source and Reference images using their proposed modified color transfer method which performs scene color transfer between images taking into account the local variations in scene characteristics of Source image and thus overcomes the information loss problem occurring in [66]. They made this modified color transfer method sensitive to the local variations in scene characteristics of images by replacing the global standard deviation ratios of Source image in (2.2.4.5.1.1.1)) with patch-wise standard deviation ratios as shown in (2.2.4.5.1.1.3).

$$l' = \left(\frac{\sigma_r^l}{\sigma_{s,\Omega(px)}^l} \right) \times (l - M_s^l) + M_r^l$$

$$\alpha' = \left(\frac{\sigma_r^\alpha}{\sigma_{s,\Omega(px)}^\alpha} \right) \times (\alpha - M_s^\alpha) + M_r^\alpha \quad (2.2.4.5.1.1.3)$$

$$\beta' = \left(\frac{\sigma_r^\beta}{\sigma_{s,\Omega(px)}^\beta} \right) \times (\beta - M_s^\beta) + M_r^\beta$$

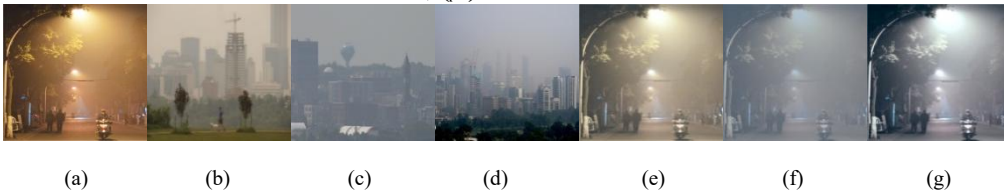


Fig. 2.2.4.5.1.1.3. (a) Source image (S), (b) R1, (c) R2, (d) R3, (e) O (S, R1), (f) O (S, R2), (g) O (S, R3)

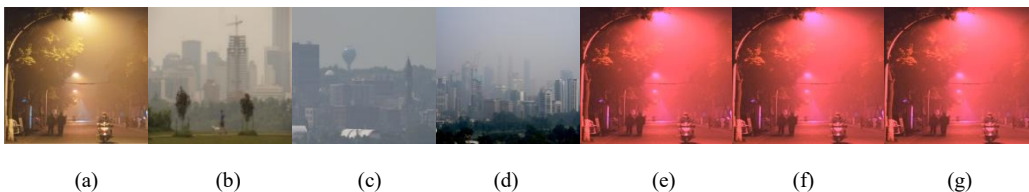


Fig. 2.2.4.5.1.1.4. (a) Source image (S), (b) R1, (c) R2, (d) R3, (e) O (S, R1), (f) O (S, R2), (g) O (S, R3)

In Fig.2.2.4.5.1.1.3 and Fig.2.2.4.5.1.1.4, R1, R2 and R3 denote three different Reference images and O (S, Rx) denotes the output obtained using the Source image, S and Reference image, Rx.

The outputs obtained by performing both global and modified color transfer methods respectively using same Reference and Source images given in the above figures prove that the modified color transfer method produces more or less similar outputs irrespective of the variance of scene characteristics and imaging conditions of Reference images compared to the global color transfer method.

Following color transfer, the authors performed dehazing of I' using a modified GIF based dehazing method which generates output haze-free image by adding amplified boundary information of I' with the filtered version of I' , Z .

Mathematically, the filtering operation of GIF is represented as follows:

$$Z(x) = a_p \cdot I'(x) + b_p, \forall x \in \Omega_{\zeta_1}(p') \quad (2.2.4.5.1.1.4)$$

In [68], the authors evaluated the self- adaptation parameters, a_p and b_p by minimizing the following energy function:

$$E(a_p, b_p) = \sum_{x \in \Omega_{\zeta_1}(p')} ((a_p \cdot I'(x) + b_p - I'(x))^2 + \lambda_{guided} a_p^2) \quad (2.2.4.5.1.1.5)$$

The values of a_p, b_p thus obtained make GIF acts as a boundary preservation filter only in the boundary regions of images while in other regions it simply acts as a mean filter. So to make it act as boundary preservation filter along entire image for achieving better edge-preservation, the authors in [71] designed the energy function stated in (2.2.4.5.1.1.6) for evaluating the self-adaptation parameters which allows ' λ_{guided} ' to vary in accordance to the local variations of image characteristics in each image region.

$$E(a_p, b_p) = \sum_{x \in \Omega_{\zeta_1}(p')} ((a_p \cdot I'(x) + b_p - I'(x))^2 + \frac{\lambda_{guided}}{\Gamma_G(p)} (a_p - \gamma_p)^2) \quad (2.2.4.5.1.1.6)$$

Where

$$\Gamma_G(p') = \frac{1}{N} \sum_{p=1}^N \frac{\chi(p') + \varepsilon}{\chi(p) + \varepsilon} \quad (2.2.4.5.1.1.7)$$

$$\gamma_p = 1 - \frac{1}{1 + e^{\eta(\chi(p') - \mu_{x,\infty})}} \quad (2.2.4.5.1.1.8)$$

$$\chi(p') = \sigma_{G,\zeta_1}(p') \sigma_{G,1}(p') \quad (2.2.4.5.1.1.9)$$

$$\eta = \frac{4}{(\mu_{x,\infty} - \min(\chi(p')))} \quad (2.2.4.5.1.1.10)$$

Filtering images using the self-adaptation parameters evaluated by optimizing the energy functions stated in (2.2.4.5.1.1.5) and (2.2.4.5.1.1.6) often lead to the production of erroneous results in the presence of complex structures and textures in images. The chances of erroneous evaluations further increase in case of nighttime images due to their spatially varying illumination characteristics.

To overcome this limitation, the authors in [70] designed a new energy function which is stated in (2.2.4.5.1.1.11) for computing a_p and b_p using boundary information of I' instead of I' itself.

$$E(a_p, b_p) = \sum_{x \in \Omega_{\zeta_1}(p')} ((a_p \cdot I'(x) + b_p - I'(x))^2 + \frac{\lambda_{guided}}{\Gamma_G(B^{\circ}p)} (a_p - \gamma_{B^{\circ}p})^2) \quad (2.2.4.5.1.1.11)$$

They evaluated the self-adaptation parameters using boundary information of I' extracted using the Prewitt operator [72] due to its significant edge detection and noise suppression capabilities.

After evaluating a_p and b_p , the authors computed filtered image, Z as stated in (2.2.4.5.1.1.4) and generated final haze-free image by adding Z with the amplified boundary information obtained by multiplying the difference of Z and I' with a constant amplification factor.

Some terms are highlighted in (2.2.4.5.1.1.6) and (2.2.4.5.1.1.11) using blue and red boxes respectively to show how the energy function designed by the authors in [70] using the boundary information of I' differs from the energy function stated in (2.2.4.5.1.1.6).

Pros:

- The modified color transfer method proposed in [70] produces more or less similar I' irrespective of the variance of color characteristics and imaging conditions of Reference images compared to the global transfer method used in [66]. Hence, the visibility of final haze-free outputs obtained from this method remain more or less same irrespective of differences in Reference images.
- The modified GIF based image dehazing method proposed in [70] does not involve the computation of atmospheric light map and transmission map for performing dehazing, which reduces its overall computational burden.
- Evaluation of self-adaptation parameters using boundary information of I' instead of I' itself in this method for performing image dehazing often reduces the chances of producing color distorted and excessively enhanced output images.

Cons:

- The modified color transfer technique proposed in this method improves visual effect of output images but it often introduces color distortions in output images.
- This method fails to recover visibility of distant scenes accurately due to non-availability of true boundary information of those scenes.

Ancuti et al. [73]: The authors in [73] proposed a novel Color Channel Transfer (CCT) prior based on their observation that in each hazy image there is at least one color channel which gets severely attenuated due to scattering of scene light by fog, dust, mist and other aerosol particles present in the atmosphere during hazy weather conditions. This phenomenon often leads to significant information loss and degrades visibility of images.

CCT compensates this information loss by transferring information from significant color channels to attenuated ones. CCT highly resembles the global color transfer procedure used in [66] and also performs information transfer using the same set of mathematical equations stated in (2.2.4.5.1.1.1) but unlike [66] and [70] where the authors used any daytime hazy image as Reference image irrespective of its scene contents and imaging conditions to perform color transfer, CCT performs color channel transfer using Reference image derived from Source image itself using the following mathematical equation:

$$r(x) = Gray(x) + Detail(x) + Saliency(x)I(x) \tag{2.2.4.5.1.1.12}$$

The saliency map is acquired similarly as proposed in [74] and the detail layer is obtained by subtracting the Gaussian blurred version of hazy image, I from I itself.

The authors in [73] have stated that the inclusion of CCT as a pre-processing step in any daytime dehazing method enables it to dehaze nighttime hazy images effectively in addition to daytime hazy images.

Pros:

- CCT performs color transfer using Reference image derived from Source image itself which enables it to overcome the shortcomings arising in [66] and [70] due to the differences between scene characteristics and imaging conditions of Source and Reference images.
- The introduction of CCT as a pre-processing step in any daytime dehazing method enables it to dehaze nighttime hazy images effectively in addition to daytime hazy images.

Cons:

- As CCT performs dehazing only in combination with other dehazing methods, its performance efficiency is largely dependent on the dehazing capabilities of these methods.



Fig. 2.2.4.5.1.1.5. (a) Source image, (b) Reference image, (c) O1, (d) O2, (e) O3

O1, O2 and O3 are the outputs obtained using the global color transfer method [66], modified color transfer method [70] and CCT [73] respectively. Both (c) and (d) are obtained using Reference image (b).

The outputs obtained from different color transfer methods prove that CCT outperforms both the global and modified color transfer methods.

Table . 2.2.4.5.1.1.1. Basic differences between the color transfer techniques used in [66], [70] and [73]

Method	Color space used	Source image	Reference image	Type of transfer
Pei <i>et al.</i> [66]	$l\alpha\beta$	Nighttime hazy image	Any daytime hazy image	Global
Jiang <i>et al.</i> [70]	$l\alpha\beta$	Nighttime hazy image	Any daytime hazy image	Local
Ancuti <i>et al.</i> [73]	$L\alpha\beta/CIE L^*a^*b^*$	Nighttime hazy image	Reference image generated from Source image itself	Channel Information

2.2.4.5.1.2. Multi-scale Fusion based method

Inspired by the DCP method proposed in [38], the authors in [75] proposed a multi-scale fusion based method to perform nighttime image dehazing, but unlike DCP where the authors have estimated atmospheric light by averaging the intensities of 0.1% of the brightest pixels belonging to the dark channel of an image or in other words it can be said by maximizing the minimum over the set of color channels. The authors in [75] performed patch-wise atmospheric light estimation for each color channel independently to handle the spatially varying illumination characteristics of nighttime images.

They initially highlighted the drawbacks which arise while estimating atmospheric light of nighttime images using the global atmospheric light estimation method proposed in [38] which are stated as follows:

- Atmospheric light map estimated using the global estimation method cannot handle the spatially varying non-uniform illumination characteristics of nighttime images accurately.
- Estimation of atmospheric light is carried out in [38] by maximizing the minimum over the set of color channels assuming the brightest pixels have high intensities in all color channels. This assumption is valid in case of daytime images as their atmospheric light is mainly contributed by sunlight whose color is white which implies that it has high intensities in all color channels but the assumption fails in case of nighttime images as the colors of their atmospheric light vary spatially in accordance to the colors of lights emitted by multiple artificial light sources. For e.g. blue light has highest intensity in blue channel, yellow light has highest intensities in green and red channels, etc.

The authors in [75] have handled these challenges as stated below:

- Instead of performing global atmospheric light estimation, the authors here performed patch-wise atmospheric light estimation to handle the spatially varying non-uniform illumination characteristics of nighttime images.

- They estimated atmospheric light for each color channel independently taking into account the varicolored lights emitted by multiple artificial light sources which enables them to overcome the second drawback.

Again, selection of optimum size patches for performing atmospheric light estimation is a challenging problem as incorrect selection often leads to erroneous estimations. Small patches provide fine spatial resolution but it reduces chances of capturing haze corrupted pixels and thus leads to the production of color distorted output images with poor global contrast. On the other hand, large patches improve global contrast and eliminate distortions introduced by airlight efficiently but fail to provide accurate results when multiple light sources are present in a single patch. In such cases, it takes the value of the winner.

To avoid any discrepancy, the authors have estimated atmospheric light using two patches, Ω and Ψ of different sizes as stated in (2.2.4.5.1.2.1) where they have fixed the size of Ψ to twice the size of Ω in order to take into account the influence of any light sources on nearby patches which may spread beyond the patch, Ω .

$$A^c = \max_{y \in \psi(z)} \left[\min_{x \in \Omega(y)} (I^c(x)) \right] = \max_{y \in \psi(z)} [I_{MIN}^c(x)] \quad (2.2.4.5.1.2.1)$$

For performing image dehazing, the authors have generated three inputs where the first and second inputs are atmospheric light maps generated using ‘ Ω ’ of two different sizes respectively. The first input is generated considering the size of ‘ Ω ’ to be small (for e.g. if the image size is 800x600, the size of ‘ Ω ’ is considered to be 20x20) while the second input is generated considering the size of ‘ Ω ’ to be large (for e.g. if the image size is 800x600, the size of ‘ Ω ’ is considered to be 80x80). The third input is the discrete Laplacian of hazy image, I which is derived mainly to preserve the fine edge details of images.

After deriving all three inputs, the authors computed three different weight maps (contrast, saliency and saturation) to assign higher weights to regions of inputs possessing high contrast (to emphasize local variations), high saliency (to enhance visibility) and small saturation (to reduce noise). Finally, they acquired output haze-free images by fusing the inputs and weight maps at multiple levels after decomposing them into Laplacian pyramid and Gaussian pyramid respectively instead of combining them directly to avoid halo-artifacts.

Pros:

- Patch-wise atmospheric light estimation method introduced in [75] effectively overcomes the shortcomings of the global atmospheric light estimation method proposed in the DCP method [38].
- Estimation of atmospheric light maps using both small and large size patches reduce the chances of producing distorted outputs which may occur while performing dehazing considering atmospheric light maps estimated using patches of incorrect sizes.

Cons:

- The output images obtained from this method contain halo-artifacts around the glow sources as it cannot effectively handle the glow effects.

2.2.4.5.2. Methods designed following New imaging model

2.2.4.5.2.1. Illumination compensation & color correction based methods

Zhang et al. [76]: The authors performed illumination compensation and color correction of nighttime hazy images prior to dehazing in [76] to obtain color corrected output images possessing balanced illumination using the following assumptions. This also enables them to effectively handle the spatially varying illumination characteristics of nighttime images.

- L , η and t are piece-wise smooth.
- R is piece-wise continuous.

They derived these assumptions from the works published in [77-78] and [38], [79] respectively. Before performing illumination compensation, the authors have replaced ‘ $\eta^c(x)R^c(x)t(x) +$

$\eta^c(x)(1 - t(x))$ term in (2.2.4.1) with ' $\widehat{R}^c(x)$ ' for simplifying the representation while performing illumination compensation as follows :

$$I^c(x) = L(x)\widehat{R}^c(x) \quad (2.2.4.5.2.1.1)$$

Taking log of the terms on both sides of (2.2.4.5.2.1.1), it becomes:

$$\log(I^c(x)) = \log(L(x)) + \log(\widehat{R}^c(x)) \quad (2.2.4.5.2.1.2)$$

Lexographical representation of (2.2.4.5.1.3.2) is:

$$ii = ll + rr \quad (2.2.4.5.2.1.3)$$

The authors have represented some terms in their work using lexographical representation to reduce the complexity as this form of representation is independent of color channels.

They obtained initial estimates of ll and rr by optimizing the objective function stated in (2.2.4.5.2.1.4) similarly as proposed in [78] but the only difference is instead of performing optimization using Bilateral filter as done in [78], the authors in [77] have performed optimization using GIF because of its improved edge-preservation and halo-artifact removal capabilities.

$$\{ll, rr\} = \underset{ll, rr: ll \geq ii}{\operatorname{arg\,min}} \{ \lambda_{ll} \|ll - ii\|_2^2 + ll^T L_p ll \} \\ + \alpha_{weight} \{ \lambda_{rr} \|rr - ii + ll\|_2^2 + rr^T L_p rr \} \quad (2.2.4.5.2.1.4)$$

Next to performing optimization, the authors refined the obtained optimized outputs, ll and rr with GIF and then computed L and \widehat{R} by applying exponential operators on those refined values. Following these computations, they balanced the overall illumination of image by performing gamma correction as follows:

$$\widehat{I}^c(x) = (L(x)^\gamma)\widehat{R}^c(x) = L^\gamma(x)\eta^c(x)R^c(x)t(x) + L^\gamma(x)\eta^c(x)(1 - t(x)) \quad (2.2.4.5.2.1.5)$$

Subsequently, they estimated the lower bound of the colour map by optimizing the objective function stated in (2.2.4.5.2.1.6) to perform color correction of the illumination balanced image, \widehat{I}^c .

$$\eta^c = \underset{\eta^c}{\operatorname{arg\,min}} \left\| \eta^c - \underline{\eta}^c \right\|^2 + \lambda_\eta (\eta^c)^T L_p \eta^c \quad (2.2.4.5.2.1.6)$$

Where

$$\underline{\eta}^c(x) = \frac{\max_{j \in \Omega_j} \widehat{I}^c(x)}{\max_{j \in \Omega_j} L^\gamma(x)} \quad (2.2.4.5.2.1.7)$$

The authors then obtained the final estimate of the colour map, η^c by multiplying the optimized output obtained from (2.2.4.5.2.1.6) with the amplification factor $((\frac{1}{3} \sum_c \eta^c)^{\gamma_0} / \frac{1}{3} \sum_c \eta^c)$ and performed colour correction of the illumination balanced image, \widehat{I}^c as follows:

$$\widetilde{I}^c(x) = \frac{\widehat{I}^c(x)}{\eta^c(x)} = L^\gamma(x)R^c(x)t(x) + L^\gamma(x)(1 - t(x)) \quad (2.2.4.5.2.1.8)$$

They stated that the final estimation of colour map, η^c may not be accurate sometimes as it is acquired simply by multiplying the initially estimated lower bound of the colour map with the amplification factor without taking into account the local variations of image colour characteristics. To overcome this limitation, they introduced an additional colour residual term, $\Delta\eta^c$ in (2.2.4.5.2.1.8) to take into account the residual colour effect as follows:

$$\widetilde{I}^c(x) = L^\gamma(x)R^c(x)t(x) + L^\gamma(x)\Delta\eta^c(x)(1 - t(x)) \quad (2.2.4.5.2.1.9)$$

$$\widetilde{I}^c(x) = L^\gamma(x)R^c(x)t(x) + L^\gamma(x) \left(\frac{\sigma^c(x)}{\eta^c(x)} \right) (1 - t(x)) \quad (2.2.4.5.2.1.10)$$

Following illumination compensation and colour correction, the authors performed dehazing of \tilde{I}^c similarly as proposed in [38].

Pros:

- As this method performs point-wise environmental illumination estimation, it can handle the spatially varying illumination characteristics of nighttime images effectively.

Cons:

- The output images produced by this method often contain halo-artifacts and colour-distortions around the glow sources.

Similar work is presented by same authors in compact form in [58].

2.2.4.5.2.2. Maximum Reflectance Prior based method

Zhang *et al* [79]: The authors in [79] performed dehazing using a novel prior namely Maximum Reflectance Prior (MRP) proposed by them based on their observation that each colour channel of every daytime, haze-free, bright image contains some pixels with very high intensities. In case of daytime images, the intensities of these pixels are mainly contributed by sunlight but for nighttime images the intensities of these pixels are mainly contributed by various multicoloured artificial light sources which enable the authors to perform estimations of atmospheric light maps of nighttime images using MRP. Examples of some nighttime hazy images and their corresponding ambient illumination maps obtained using MRP are given in Fig.2.2.4.5.2.2.1.

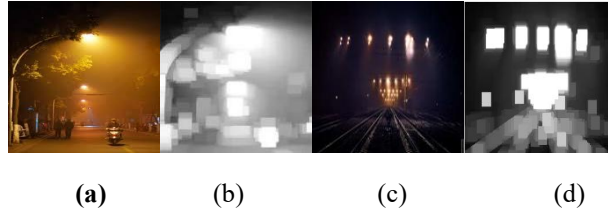


Fig. 2.2.4.5.2.2.1. (a) Image 1, (b) Illumination map of Image 1, (c) Image 2, (d) Illumination map of Image 2

The maximum reflectance map of any daytime image is evaluated as follows:

$$M_{\Omega(x)}^c = \max_{x \in \Omega(y)} I^c(x) = \max_{x \in \Omega(y)} L_{\Omega(y)} R^c(x) \quad (2.2.4.5.2.2.1)$$

The incident light intensity, $L_{\Omega(y)}$ of any daytime haze-free image is globally uniform and its value is considered to be approximately equal to 1 as atmospheric light of daytime image is mainly contributed by sunlight which is usually white in colour and possess high intensities. Moreover, the objects and surfaces which possess the pixels having maximum local intensities at each colour channel also have high reflectance values. Hence, maximum reflectance maps of daytime, haze-free images are ideally $M_{\Omega(x)}^c \approx 1$. But the situation differs in case of nighttime images due to the presence of multi-coloured artificial light sources. Hence, the maximum reflectance maps of images are computed as follows:

$$M_{\Omega(x)}^c = \max_{x \in \Omega(y)} I^c(x) = \max_{x \in \Omega(y)} R^c(x) \left(L_{\Omega(y)} \eta_{\Omega(y)}^c t_{\Omega(y)} \right) + L_{\Omega(y)} \eta_{\Omega(y)}^c (1 - t_{\Omega(y)}) \quad (2.2.4.5.2.2.2)$$

The authors substituted ' $\max_{x \in \Omega(y)} R^c(x)$ ' in (2.2.4.5.2.2.2) by '1' assuming the reflectance properties of nighttime and daytime images to be similar and estimated the colour maps as follows:

$$M_{\Omega(x)}^c = L_{\Omega(y)} \eta_{\Omega(y)}^c t_{\Omega(y)} + L_{\Omega(y)} \eta_{\Omega(y)}^c - L_{\Omega(y)} \eta_{\Omega(y)}^c t_{\Omega(y)} \quad (2.2.4.5.2.2.3)$$

$$M_{\Omega(x)}^c = L_{\Omega(y)} \eta_{\Omega(y)}^c \quad (2.2.4.5.2.2.4)$$

$$\eta^c_{\Omega(y)} = \frac{M^c_{\Omega(x)}}{L_{\Omega(y)}} \quad (2.2.4.5.2.2.5)$$

They assumed incident light intensity, $L_{\Omega(y)}$ to be independent of colour channels and fixed it's value equal to the maximum of $M^c_{\Omega(x)}$ obtained for all colour channels as stated in (2.2.4.5.2.2.6). The authors of this method too performed colour correction of images similarly as proposed in [76].

$$L_{\Omega(y)} = \max_{c \in \{r, g, b\}} \left(\max_{x \in \Omega(y)} \tilde{I}^c(x) \right) \quad (2.2.4.5.2.2.6)$$

Finally, using the estimated values of $\eta^c_{\Omega(y)}$ and $L_{\Omega(y)}$, they performed dehazing of images similarly as proposed in [38].

Pros:

- Unlike DCP, MRP can effectively handle the spatially varying illumination characteristics of nighttime images as it performs patch-wise atmospheric light estimation instead of estimating it globally.
- The pixels belonging to the glow regions of images usually possess high intensity values. As MRP computes atmospheric light maps of images taking into account the pixels possessing high intensity values, it can handle the undesirable glow effects present in nighttime images to some extent.

Cons:

- MRP is designed based on the observation derived from the characteristics of daytime, haze-free images which are very different from those of nighttime images. This often leads to the production of distorted output images.

2.2.4.5.3. Methods designed following Multiple Light Colors & Glow model

2.2.4.5.3.1. Filtering based method

Li et al. [59]: In contrary to the methods discussed above, the authors in [59] have designed their method taking into account both the glow characteristics as well as the spatially varying illumination characteristics of nighttime images. Prior to performing image dehazing, they removed glow from images to avoid the presence of any undesirable colour distortions in output images which are introduced by these glow sources using the target layer separation method proposed in [80] portraying nighttime hazy image as a summation of glow image and glow free hazy image as follows:

$$I(x) = G(x) + H(x) \quad (2.2.4.5.3.1.1)$$

Some examples of nighttime hazy images along with their glow images and glow free hazy images are given in Fig.2.2.4.5.3.1.1.

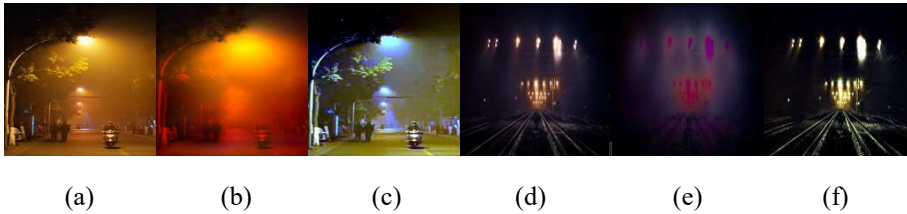


Fig. 2.2.4.5.3.1.1. (a) Image 1, (b) G of Image 1, (c) H of Image 1 (d) Image 2, (e) G of Image 2, (f) H of Image 2

The authors performed glow removal by optimizing the objective function stated in (2.2.4.5.3.1.2) using the half-quadratic splitting technique. They have designed this objective function considering $G(x) = I(x) - H(x)$ as a smooth layer as the brightness of glow decreases gradually with increasing scene distance.

$$E(H) = \sum_x (\rho(H(x) * f_{1,2}) + \lambda((I(x) - H(x)) * f_3)^2) \quad (2.2.4.5.3.1.2)$$

$$s. t. 0 \leq H(x) \leq I(x)$$

$$\sum_x H_r(x) = \sum_x H_g(x) = \sum_x H_b(x)$$

The first term on the right hand side of (2.2.4.5.3.1.2), $\rho(s) = \min(s^2, \tau)$ is a robust function which is used to preserve all the large gradients of hazy image in H while the second term uses L_2 norm regularization to force the gradients of G to produce a smooth glow layer which often leads to loss of color information of images. To overcome this drawback, the authors in [59] introduced two inequality constraints stated below (2.2.4.5.3.1.2) where the first inequality constraint restricts the range of the solution obtained by optimizing (2.2.4.5.3.1.2) and the second inequality constraint forces the intensities of different color channels to be balanced.

Following glow decomposition, the authors estimated spatially varying atmospheric light map of H using a GIF based estimation method which performs selection of intensity of brightest pixel belonging to each of 15×15 patch of H as that patch's atmospheric light value. Finally, they generated output haze-free images after estimating transmission map similarly as proposed in DCP [38].

Pros:

- This method can handle both the glow characteristics as well as the spatially variant illumination characteristics of nighttime images effectively.

Cons:

- Ideally, the smoothness parameter ' λ ' in the objective function (2.2.4.5.3.1.2) should be automatically evaluated in accordance to several factors like haziness of scenes, scene depth and types of light sources for obtaining accurate results. But as it becomes very challenging to derive the values of all these parameters using information derived from single image only, so the authors in [59] have set it to a constant value (500) manually which often leads to the production of distorted outputs.

2.2.4.5.3.2. Weighted Entropy based methods

Park et al [81]: The authors of this method too performed glow removal from images using the target layer separation method like the authors in [59] but instead of estimating atmospheric light map using GIF based method, they estimated it using their proposed atmospheric light selection rule as they pointed out that simply selecting the intensity of the brightest pixel of each image patch as its atmospheric light value often leads to erroneous atmospheric light estimations especially in low light regions as well as in the regions having uniform illumination. Hence, dehazing images using the atmospheric light map thus estimated often leads to the production of distorted outputs. To overcome this shortcoming, they performed atmospheric light estimation using their proposed atmospheric light selection rule which is mathematically represented in (2.2.4.5.3.2.1) as follows:

$$A(x) = s_{local_map}(x)A_{local}(x) + (1 - s_{local_map}(x))A_{global} \quad (2.2.4.5.3.2.1)$$

Where

$$s_{local_map}(x) = \begin{cases} 1, & \|A_{local}(x)\| \geq \|A_{global}\| \\ 0, & otherwise \end{cases}$$

The authors estimated A_{global} and A_{local} using the quad tree subdivision method [82] and the GIF based atmospheric light estimation method proposed in [59] respectively.

Following the estimation of atmospheric light map, they estimated transmission by optimizing a weighted entropy based objective function as stated in (2.2.4.5.3.2.2). Initially, the authors considered "entropy" as an ideal parameter for designing the objective function for estimating transmission as entropy of haze-free images are greater compared to hazy images. But later they found that estimating transmission by simply maximizing image entropy often leads to the production of visually distorted results as it gives rise to truncation errors and forces most of the pixels to get concentrated at very low intensity values. To overcome these limitations, they designed the objective function based on the "weighted entropy" of image which besides providing good contrast also minimizes truncation errors.

$$f_{objective}(t) = - \sum_k w(k|t) p(k|t) \log p(k|t) \quad (2.2.4.5.3.2.2)$$

Where

$$w(k|t) = \gamma(t) \exp \left[\frac{(k - (H)_{mean}^L)^2}{\sigma^2} \right]$$

$$\gamma(t) = \frac{1}{N} \sum_x \delta(x|t) \quad \delta(x|t) = \begin{cases} 1, & 0 \leq H^L(x|t) \leq 100 \\ 0, & otherwise \end{cases}$$

The weight term, $w(k|t)$ is introduced in the above objective function to minimize truncation errors and to maintain the image brightness.

The authors estimated final transmission map by optimizing the objective function for each of m sub-block of H as follows:

$$t_m^{block} = \underset{t \in \{0.01 \leq t \leq 1\}}{\text{arg max}} f_{objective}(t) \quad (2.2.4.5.3.2.3)$$

Finally, they obtained haze-free images using the values of estimated transmission and atmospheric light maps as stated in (2.2.5).

Pros:

- The atmospheric selection rule proposed in this method allows it to perform accurate estimation of atmospheric light map for all types of image regions possessing diverse illumination characteristics and thus enables this method to effectively overcome the shortcomings of GIF based atmospheric light estimation method proposed in [59].
- Weighted entropy based transmission estimation technique proposed in this method, enables it to produce output images with good contrast.

Cons:

- Ideally the smoothness parameter ‘ λ ’ and the parameter ‘ σ ’ which plays a crucial role in evaluating the weight term as shown in (2.2.4.5.3.1.2) and (2.2.4.5.3.2.2) respectively should be automatically calculated according to several image characteristics to obtain accurate results but here the authors manually have set their values to some constants which often leads to the introduction of artifacts and noise in output images.

Similar work is published in compact form in [83].

Jin et al. [84]: The authors of this method too performed glow removal from images using the target layer separation method similarly as done in methods discussed above. They estimated atmospheric light map using the multi-scale depth fusion method proposed in [75] by fusing two inputs and two weight maps (contrast weight map and saturation weight map). They generated first and second inputs from H using Ω of two different sizes respectively as stated in (2.2.4.5.1.2.1). (For e.g. If size of H is 500x300, then first input is generated considering the size of Ω to be 10x10 and second input is generated considering the size of Ω to be 30x30).

The authors in [84] also estimated transmission by optimizing weight entropy based objective function stated in (2.2.4.5.3.2.2) but only after inserting an additional $f_{hue}(t)$ term to it. They stated that output images obtained using transmission map estimated by optimizing the objective function stated in (2.2.4.5.3.2.2) often produces colour distorted outputs as it was designed without considering the hue characteristics of images. So, to solve this limitation, the authors modified (2.2.4.5.3.2.2) by introducing the new term, $f_{hue}(t)$ in the re-defined version of (2.2.4.5.3.2.2) which is stated in (2.2.4.5.3.2.4) as follows:

$$f_{objective}(t) = f_{entropy}(t) \cdot f_{fidelity}(t) \quad (2.2.4.5.3.2.4)$$

$$f_{objective}(t) = \frac{f_{entropy}(t) \cdot f_{fidelity}(t)}{\max(f_{hue}(t), 0.1)} \quad (2.2.4.5.3.2.5)$$

The term $f_{hue}(t)$ measures the hue of an image before and after haze removal and is mathematically represented as:

$$f_{hue}(t) = \frac{h_J(t) - h_H(t)}{N} \quad (2.2.4.5.3.2.6)$$

Following the estimations of atmospheric light and transmission, the authors of this method generated output haze-free images using the estimated values similarly as proposed in (2.2.5).

Pros:

- The introduction of the $f_{hue}(t)$ term in the objective function designed for performing transmission estimation in this method enables it to take into account the hue characteristics of images in addition to their contrast and fidelity and thus ensures the production of output images with minimum color distortions.
- This method provides contrast enhanced images with good visual effect.

Cons:

- The shortcomings arising due to manual evaluation of ‘ λ ’ and ‘ σ ’ parameters in [59] and [81] also exist in this method.

2.2.4.5.3.3. Super-pixel based method

Yang *et al* [85]: The authors in [85] too performed glow removal from images using the same target layer separation method but in contrary to performing patch-based atmospheric light and transmission estimations as done in the methods discussed above, the authors of this method estimated atmospheric light and transmission maps using their proposed super-pixel based estimation techniques which can efficiently overcome the shortcomings of patch-based estimation techniques.

A super-pixel is a collection of adjacent pixels possessing similar color, texture, brightness, etc. So, all the pixels belonging to same super-pixel usually have similar transmission values and hence, there is no possibility of intra-patch transmission variation which reduces the chances of presence of halo-artifacts in output images which mostly occur due to intra-patch transmission variation in patch based estimation methods.

Prior to performing atmospheric light estimation, the authors of this method initially divided a glow free nighttime hazy image, H into several super-pixels using Simple Linear Iterative Clustering (SLIC) method [86] and then performed estimation of atmospheric light map by selecting the intensity of the brightest pixel of each super-pixel as the value of that super-pixel’s atmospheric light. Mathematically, this super-pixel based atmospheric light estimation technique is represented as follows:

$$J^c(x) = R^c(x)A^c(x) \quad (2.2.4.5.3.3.1)$$

$$H^c(x) = R^c(x)A^c(x) t(x) + A^c(x)(1 - t(x)) \quad (2.2.4.5.3.3.2)$$

$$\max_{x \in S_i(y)} H^c(x) = \max_{x \in S_i(y)} \{R^c(x)\}A^c(x) t(x) + A^c(x)(1 - t(x)) \quad (2.2.4.5.3.3.3)$$

Substituting, $\max_{x \in S_i(y)} \{R^c(x)\} = 1$ in (2.2.4.5.3.3.3) using the same logic proposed in MRP [79], the following equation is derived:

$$\max_{x \in S_i(y)} H^c(x) = A^c(x) \quad (2.2.4.5.3.3.4)$$

After acquiring the initially estimated atmospheric light map, the authors refined it using weighted GIF [71] instead of refining it with GIF [68] as weighted GIF is more sensitive towards the local variations in image characteristics.

Next to performing atmospheric light estimation, the authors estimated transmission and performed dehazing as stated below.

$$\min_{x \in S_i} \left(\min_{c \in \{r, g, b\}} \frac{H^c(x)}{A^c(x)} \right) = \widetilde{t}(y) \min_{y \in S_i} \left(\min_{c \in \{r, g, b\}} \frac{J^c(x)}{A^c(x)} \right) + 1 - \widetilde{t}(x) \quad (2.2.4.5.3.3.5)$$

$$J(x) = \frac{H(x) - A(x)}{\max(t(x), t_m)} + A(x) \quad (2.2.4.5.3.3.6)$$

Where

$$t_m = \begin{cases} 0.375, & \text{for dense haze} \\ 0, & \text{otherwise} \end{cases}$$

Pros:

- The use of weighted GIF by the authors of this method to refine the initially estimated atmospheric light map enhances its edge-preservation capability as weighted GIF is more sensitive to the local variations in image characteristics compared to GIF.
- As this method performs super-pixel based transmission estimation instead of estimating it using patch based techniques like other methods, it produces outputs containing almost negligible halo-artifacts and distortions as there is no chance of intra-patch transmission variation.
- The adaptive threshold, t_m introduced in (2.2.4.5.3.3.6) reduced the distortions occurring in output images especially in the sky regions. Accurate selection of t_m also enhances the chances of producing visually improved outputs.

Cons:

- The use of super pixel concept estimation techniques in this method enhances its time complexity.

2.2.4.5.4. Method designed following Nighttime haze model

2.2.4.5.4.1. Deep neural network based method (DeGlow-DeHaze Network)

Kuanar et al. [61]: The authors of this method performed detection and removal of glow present in nighttime hazy images using their proposed DeGlow network as shown in Fig.2.2.4.5.4.1.1 below through recursive learning.

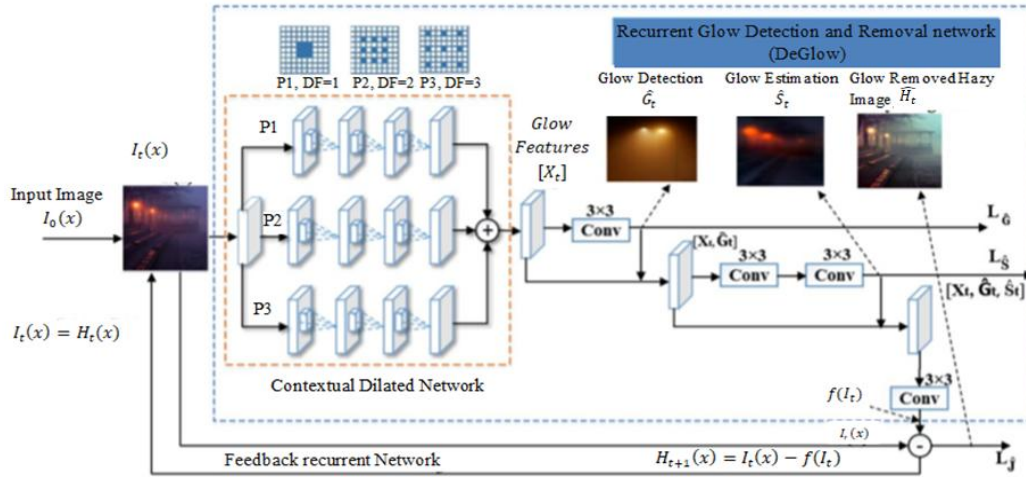


Fig.2.2.4.5.4.1.1. Architecture of DeGlow network [61]

DeGlow network performs glow detection and removal using a set of glow features $[X_t]$ extracted using a contextual dilated network which consists of three dilated paths, P1, P2 and P3 possessing three convolution layers with kernel size 3×3 but with different dilation factors (DF) (P1: DF=1, P2:DF=2, P3:DF=3). The use of different DF in these paths facilitates the aggregation of more contextual information and thus prevents the loss of local image details and resolutions.

Next to performing glow removal, the authors estimated transmission from output glow-free, hazy image, H obtained from DeGlow network using DeHaze network whose architecture is similar to

contextual dilated network but with one recurrence. DeHaze network performs transmission estimation following the same logic as proposed in [52].

Following transmission estimation, the authors computed atmospheric light by averaging the intensities of 0.1% of the brightest pixels belonging to estimated transmission map considering the atmospheric light of H to be globally uniform.

The pictorial representation of the entire workflow of the method proposed in [61] is given in Fig.2.2.4.5.4.1.2 .

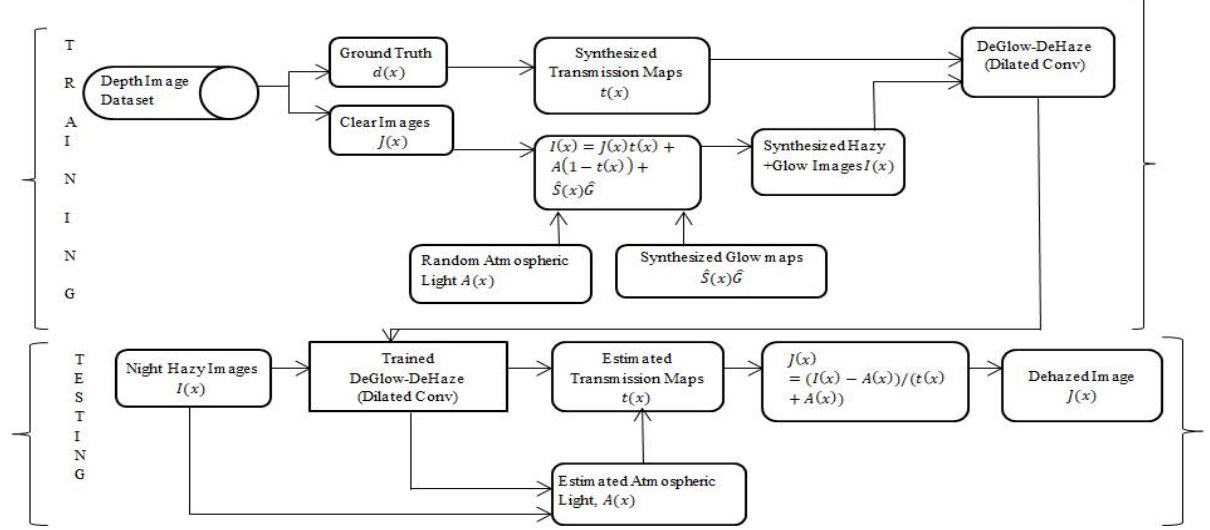


Fig.2.2.4.5.4.1.2. Pictorial representation of the workflow of the method proposed in [61]

Pros:

- The use of three dilated paths with different DF for extracting glow features in this method enables it to preserve the local image details as well as the resolution of images effectively as these paths support the aggregation of more contextual information.

Cons:

- This method performs estimation of atmospheric light by averaging 0.1% of the brightest pixels belonging to transmission map estimated using DeHaze network assuming that atmospheric light of glow free nighttime hazy image, H which is obtained as output from DeGlow network is globally uniform. This assumption mostly does not hold for nighttime images as their atmospheric light varies spatially in accordance to the intensities and colors of lights emitted by multi-colored artificial light sources.

2.2.4.5.5. Method designed following Improved atmospheric scattering model

2.2.4.5.5.1. Fusion based method

Lin et al [62]: The authors of this method performed dehazing of images following the logic of the total variation model which states that an image is composed of structure layer and texture layer where the structure layer contains haze and brightness of images and the texture layer contains their details and noise.

They initially estimated the structure layer of an image by minimizing the objective function stated in (2.2.4.5.5.1.1) using the half-quadratic splitting technique [87] and then computed the texture layer by subtracting the estimated structure layer from the image as stated in (2.2.4.2.3.1).

$$\min_{St(x)} \sum_x (St(x) - I(x))^2 + \lambda |\nabla St(x)| \quad (2.2.4.5.5.1.1)$$

After estimating the structure layer, the authors further decomposed it into glow layer and background layer and then performed glow removal using the target layer separation method similarly as proposed in [61],[85],etc. to remove the distortions introduced by undesirable glow effects on image structure. Following glow removal from images, the authors estimated atmospheric light by optimizing W , the output obtained by white balancing the mean of background layer as stated in (2.2.4.5.5.1.2) using mean-pooling and bicubic interpolation algorithm [88] as shown in Fig.2.2.4.5.5.1.1.

$$W = 1 - \left(\max_{c \in \{r, g, b\}} \overline{St_B^c} - \min_{c \in \{r, g, b\}} \overline{St_B^c} \right) \quad (2.2.4.5.5.1.2)$$

Input: $W(x)$, normalization parameter ζ , window size R
Initialize: $\zeta = 0.7, R = 50$
Step: $W'(x) = \text{meanPooling}(W(x))$
$W'(x) = \text{bicubicInterpolation}(W'(x))$
$W'_{\min}(x) = \min(W'(x))$
$W'_{\max}(x) = \max(W'(x))$
$A(x) = \zeta \cdot \frac{W'(x) - W'_{\min}(x)}{W'_{\max}(x) - W'_{\min}(x)} + (1 - \zeta) \cdot W'_{\max}(x)$
Output: $A(x)$

Fig.2.2.4.5.5.1.1. Pseudo-code of the atmospheric light estimation method [62]

Following atmospheric light estimation, the authors optimized the texture layer as stated in (2.2.4.5.5.1.3). Subsequently, they estimated transmission map as proposed in DCP [38] and refined it using GIF.

$$\widehat{T}e(x) = \frac{255}{1 + e^{0.015 \cdot (-Te(x) + 100)}} \cdot A(x) \quad (2.2.4.5.5.1.3)$$

Finally, the authors obtained output haze-free image by fusing the optimized restored structure layer and optimized texture layer as shown in Fig.2.2.4.5.5.1.2.

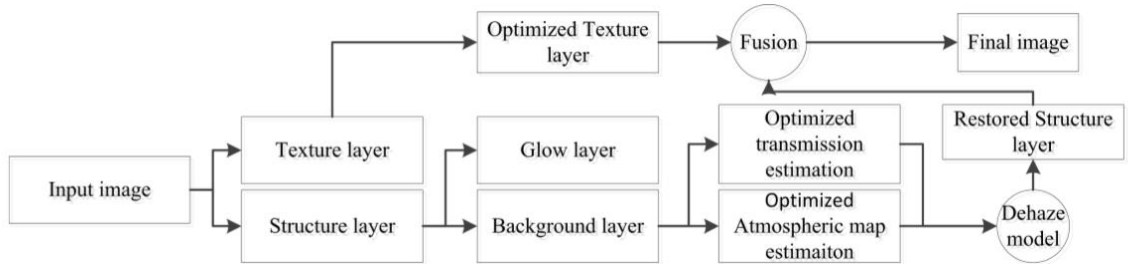


Fig.2.2.4.5.5.1.2. Block diagram of the method [62]

Pros:

- This method primarily focuses on minimizing the texture loss in outputs which enables it to restore the texture information of images and preserve image details efficiently.

Cons:

- This method performs dehazing of nighttime images considering that entire image noise is present in texture layer of images but in reality, some noise also exists in their structure layer which gets enhanced during restoration of background layer leading to the production of noisy outputs.

2.2.4.5.6. Methods designed following Relaxed atmospheric scattering model

2.2.4.5.6.1. Colour line model based method

Santra *et al.* [64]: The authors in [64] designed this method considering the properties of the colour line model [89] which states that the colors of each patch of an image should ideally lie on a line passing through the origin of the plane defined by its RGB vectors. However, in reality colors spread to form an elongated cluster due to the noise introduced by sensors and other camera related distortions. In case of hazy images, it further gets shifted from the origin by an amount α_{shift} in \hat{A} direction due to the influence of additive airlight. Hence, to enable the model stated in (2.2.4.3.1) to evaluate the magnitude and direction of atmospheric light at each pixel position, the authors have re-defined (2.2.4.3.1) as follows:

$$I(x) = J(x)t(x) + m(x)\hat{A}(x)(1 - t(x)) \quad (2.2.4.5.6.1.1)$$

$$I(x) = J(x)t(x) + \alpha_{shift}(x)\hat{A}(x) \quad (2.2.4.5.6.1.2)$$

Where

$$\alpha_{shift}(x) = m(x)(1 - t(x))$$

Thereafter, they performed dehazing of images by neutralizing the effect of airlight on each pixel by moving its color line in the opposite direction of \hat{A} by an amount α_{shift} but computing \hat{A} and α_{shift} for each pixel often increases the method's execution time. To overcome this limitation, the authors computed \hat{A} and α_{shift} patch-wise instead of performing pixel-wise computation. Intersection of two non-parallel patches having same \hat{A} but possessing distinct color lines l_1 and l_2 is shown in Fig.2.2.4.5.6.1.1.

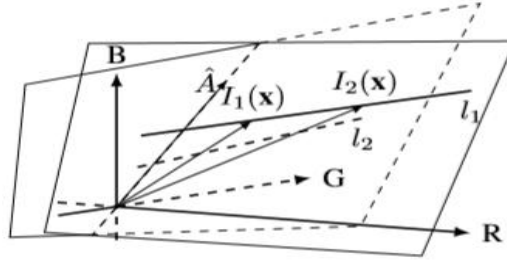


Fig.2.2.4.5.6.1.1. Intersection of two non-parallel patch planes with distinct color lines l_1 and l_2 [64]

For evaluating α_{shift} and \hat{A} , the authors initially divided each input hazy image into several patches with 50% overlap and estimated color line ($Line = \rho D + P_0$) of each patch by applying RANSAC which provides two points (I_1, I_2) defining the color line of each patch as shown in Fig.2.2.4.5.6.1.1 and a set of inlier points. They estimated parameters of $Line$ and the normal, \hat{n} to the plane containing the $Line$ and the origin as follows:

$$P_0 = I_1, D = \frac{I_2 - I_1}{\|I_2 - I_1\|}, \hat{n} = \frac{I_2 \times I_1}{\|I_2 \times I_1\|} \quad (2.2.4.5.6.1.3)$$

Subsequently, they computed α_{shift} and \hat{A} of patches which satisfy the following properties by estimating the parameters of the planes mathematically defined in (2.2.4.5.6.1.4) using Hough Transform.

- Number of inlier points are greater than 50% of the total pixels lying within the patch.
- All components of D are positive.
- All the pixels of the patch should belong to a single object and have similar reflectance values.
- The distance between origin of the patch plane and the estimated color line is high.
- The color intensity variance within a patch should be large along the direction of the color line.

$$n_x \cos \theta \sin \phi + n_y \sin \theta \sin \phi + n_z \cos \phi = 0 \quad (2.2.4.5.6.1.4)$$

Where $[n_x, n_y, n_z]^T = \hat{n}$ and θ and ϕ should lie between $[0^\circ, 90^\circ]$ as \hat{A} indicates a color in RGB color space which ensures that all of its components will always be positive.

\hat{A} values of pixels belonging to patches which do not satisfy these properties are computed using the Interpolation method proposed in [90]. Finally, using the estimate values of α_{shift} and \hat{A} , the authors generated final haze free image using the following equation:

$$J(x) = \frac{I(x) - (1 - t(p))m(x)\hat{A}(x)}{1 - Y((1 - t(x))m(x)\hat{A}(x))} \quad (2.2.4.5.6.1.5)$$

Where $Y(I(x)) = 0.2989I_R(x) + 0.5870I_G(x) + 0.1140I_B(x)$

Pros:

- This method can efficiently dehaze both nighttime and daytime hazy images.
- As this method performs dehazing of pixels by neutralizing the influence of airlight on them, it can effectively handle both the spatially varying illumination characteristics as well as the glow characteristics of nighttime images effectively.

Cons:

- The good performance of this method is not always guaranteed as it sometimes provides very dark results for images containing objects having intensities and colors similar to airlight.

2.2.4.5.6.2. Pixel-wise alpha blending based method

Yu et al. [91]: The authors in [91] performed dehazing of images using the concepts of both DCP [38] and Bright Channel Prior (BCP) [92] which is simply the reverse of DCP. BCP gives accurate estimations for bright light regions in contrary to DCP which gives accurate estimations for low light regions.

The authors in this work estimated atmospheric light using the Retinex theory [93] which states that the spatially varying atmospheric light term, $A(x)$ in (2.2.4.3.1) is a spatially smooth, low-frequency term and $(R(x)t(x) + (1 - t(x)))$ is a high-frequency term. The theory also stated that atmospheric light of an image can be estimated simply by filtering the low frequency atmospheric light term, $A(x)$ in (70) using a low pass Gaussian filter.

$$I(x) = R(x)A(x)t(x) + A(x)(1 - t(x)) \quad (2.2.4.5.6.2.1)$$

$$I(x) = A(x)(R(x)t(x) + (1 - t(x))) \quad (2.2.4.5.6.2.2)$$

But as GIF performs edge-preservation more efficiently compared to Gaussian filter, the authors in [91] estimated atmospheric light by filtering the low-frequency atmospheric light term using GIF considering I^{cd} , computed as stated in (2.2.4.5.6.2.3) as the guided image.

$$I^{cd}(x) = \max_{c1 \in \{r, g, b\}} I^{c1}(x) - \min_{c2 \in \{r, g, b\}} I^{c2}(x) \quad (2.2.4.5.6.2.3)$$

The authors generated final transmission map by pixel-wise blending transmission maps, t_{BCP} and t_{DCP} obtained as proposed in [92] and [38] respectively using a brightness aware map, $alpha$ as stated in (2.2.4.5.6.2.5).

$$alpha(x) = \max_{c \in \{r, g, b\}} (I^c)^y \quad (2.2.4.5.6.2.4)$$

$$t(x) = t_{BCP}(x)alpha(x) + t_{DCP}(x)(1 - alpha(x)) \quad (2.2.4.5.6.2.5)$$

Following the estimations of transmission map and atmospheric light, the authors performed scene radiance recovery as stated in (2.2.4.5.6.2.6).

$$J(x) = \frac{I(x) - A(x)(1 - t(x))}{t(x)} \quad (2.2.4.5.6.2.6)$$

Pros:

- The use of both t_{DCP} and t_{BCP} for performing estimation of final transmission map in this method, enables it to produce output images possessing desired illumination in both low-light and bright regions.
- Estimation of atmospheric light using GIF, considering I^{cd} as guided image also enhances the edge-preservation capability of the method.

Cons:

- This method cannot produce accurate results for images possessing complex structures.

2.2.4.5.7. Method designed following Nighttime Hazy Image Model with HDP Function

2.2.4.5.7.1. Deep neural network based method (HDP-Net)

Liao *et al* [65]: The authors in [65] designed a learning based deep neural network namely, HDP-Net to dehaze nighttime hazy images. Unlike DeGlow-DeHazeNet [61] which detects and removes glow from images and then estimates transmission map of glow-free nighttime hazy image, H , HDP-Net performs end-to-end dehazing and produces output haze free image simply by subtracting predicted haze density map ' I_h ' from hazy image, I . The architecture of HDP-Net is given in Fig. 2.2.4.5.7.1.1 and the description of its layers are given in Table.2.2.4.5.7.1.1 .

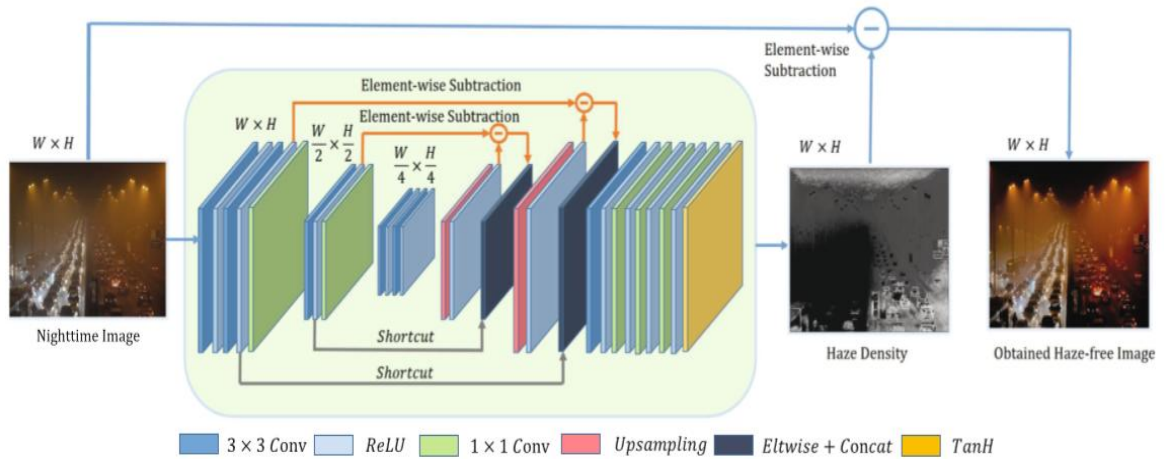


Fig. 2.2.4.5.7.1.1 HDP-Net architecture [65]

Table. 2.2.4.5.7.1.1. Description of layers of HDP-Net [65]

Formulation	Type	Input Size	Num	Filter	Stride	Pad
Feature Extraction	Conv	3x128x128	8	3x3	1	1
		8x128x128	16	3x3	1	1
		16x128x128	32	1x1	2	0
		32x64x64	32	3x3	1	1
		32x64x64	64	1x1	2	0
		64x32x32	64	3x3	1	1
		64x32x32	64	3x3	1	1
Fusion	Deconv	64x32x32	32	2x2	2	0
		32x64x64	16	2x2	2	0
Feature Extraction	Conv	16x128x128	16	3x3	1	1

Mapping	Conv	16x128x128	16	1x1	1	0
		16x128x128	8	1x1	1	0
		8x128x128	3	1x1	1	0

The layers of HDP-Net (described in Table. 2.2.4.5.7.1.1) perform different operations like: convolutional layers with kernel 3, stride 1, pad 1 perform feature extraction, those with kernel 1, stride 2 and pad 0 serve as pooling layers and perform down-sampling to generate half-size feature maps. Convolutional layers with kernel 1, stride 1 and pad 0 serve as mapping layers which perform multistage mapping to obtain a color haze density map having dimension similar to that of input image. Deconvolutional layers recover the true size of the feature maps which are fed as inputs to pooling layers by performing up-sampling.

The two shortcut connections are added to HDP-Net architecture (as shown in Fig.2.2.4.5.7.1.1) in order to enable it to concatenate low-level feature maps with high-level feature maps to retain image color and edge details while the three subtraction paths are added to HDP-Net architecture to minimize training error. They have chosen ReLU as a non-linear activation function of each convolutional layer (except the last one) due to its sparsity property to reduce the memory requirement, only for the last convolutional layer they have chosen TanH as an activation function to restrict the range of output haze density map within [-1 1] as the authors have mathematically proved that range of haze density maps of images will always lie between [-1 1]. The authors trained HDP-Net using the following cost function:

$$Loss = \sum \|J - (I - I_{conv})\|_2 \quad (2.2.4.5.7.1.1)$$

They carried out training using Stochastic Gradient Descent as an optimization algorithm and back propagation as gradient computing technique considering 0.001 as initial learning rate to enhance the training speed but decreased it gradually to obtain better convergence effects. They set the values of momentum, weight decay, power and gamma to 0.9,0.005, 0.75 and 0.001 respectively.

Pros:

- This method does not involve the estimations of atmospheric light and transmission to perform image dehazing which reduces its computation burden.
- This method performs dehazing of hazy images taking into account the difference of hue characteristics of images captured under different lighting conditions and thus reduces the chances of producing color-distorted output images.

Cons:

- Accurate prediction of haze density maps for images possessing varied illumination and glow characteristics often becomes very challenging.

A detailed discussion on the methodologies of both the daytime and nighttime image dehazing methods are carried out in this Chapter and the limitations of each of the methods are also highlighted here.

In Chapter 3 and Chapter 5, detailed discussion is carried out regarding the methodologies of novel daytime and nighttime image dehazing techniques which are designed in this thesis to solve the shortcomings of existing methods which are discussed in this Chapter.

Chapter 3

In this Chapter, detailed description of the methodology of the Bacterial Foraging and Fuzzy synergism based image dehazing method which is proposed in this thesis to overcome the shortcomings of the existing daytime image dehazing methods is given.

Some notable shortcomings of the existing daytime image dehazing methods (discussed in **Chapter 2**) are listed as follows:-

- The authors of the existing works have performed image dehazing, mostly portraying it as a simple contrast enhancement problem. They have hardly focused on the other important aspects of dehazing like edge and noise detection, noise removal and edge-sharpening while designing the methods.
- The contrast enhancement techniques adapted in most of the existing works enhance the contrast of all the pixels belonging to a hazy image identically irrespective of their extent of degradation which often leads to the production of over-saturated or poorly enhanced color distorted output images.
- In some of the works where image dehazing is carried out taking into account the edge detection aspect in addition to the contrast enhancement aspect, also do not give desired outputs because either traditional edge-detectors (like Sobel detector) or filters (like modified median filter) are used in such methods for edge-detection purpose. These edge detectors mostly detect edges along certain specified angles only. A detailed discussion on the shortcomings of several existing well-known edge detection methods is given in **Section 3.1.2.6**.
- For these reasons, output images obtained from the existing dehazing methods are mostly noisy/over-saturated/poorly enhanced/color distorted and contain halo artifacts.

So, from the above discussion, it becomes evident that none of the existing dehazing methods perform image dehazing taking into account all of its four crucial aspects like contrast enhancement, edge& noise detection, noise removal and edge-sharpening. Hence, to solve the shortcomings of existing works there is a need of a method which can perform dehazing taking into account all its crucial aspects in order to obtain desirable results.

3.1. Novel Fuzzy Logic and Bacterial Foraging (BF) synergism based daytime image dehazing method

To overcome the limitations of existing daytime image dehazing works, a novel Fuzzy Logic and BF synergism based daytime image dehazing method is designed in this thesis.

The main objective behind designing this method is to perform haze removal from images considering all crucial aspects of image dehazing problem like contrast enhancement, edge& noise detection, noise removal and edge-sharpening and also to process each pixel of an image according to its properties (like extent of degradation, extent of noise corruption, etc.) at each step of the proposed method to reduce chances of producing noisy/over-saturated/over-enhanced or poorly enhanced colour distorted output images containing halo artifacts. Pixel-wise processing of images at each step of the method also enhances its flexibility.

The proposed BF-Fuzzy synergism based method has effectively overcome the limitations of existing methods as it performs image dehazing by giving equal importance to all of its crucial aspects. It also processes each pixel according to its properties at each step of the proposed method using novel set of FI rules which are exclusively designed for processing pixels at that step. The use of different set of FI rules to process pixels at each step of the proposed method not only enhances the proposed method's flexibility but also minimizes chances of producing distorted outputs. For this purpose, Fuzzy Logic and BF algorithm are chosen to design the proposed dehazing method due to the following reasons:

- **Fuzzy Logic** since its commencement in 1965 by Zadeh [28] has proved its efficiency in solving several engineering problems. Its capability of solving the problems by utilizing the human cognitive ability in form of several FI rules makes it a perfect choice for solving the

dehazing problem. In hazy images, usually the concentration of haze varies region-wise within an image [50]. Hence, it is very essential to process each pixel of an image according to its extent of degradation, otherwise the obtained output haze free images will suffer from over-saturation/ poor enhancement etc. Most image dehazing methods like [38], [40], [43], [47] and [55-56] do not focus on this issue and enhance all the pixels belonging to hazy image identically. Hence, those methods often produce distorted results. Although the authors in [50] have tried to deal with this problem using their proposed CAP in combination with the gradient descent learning method, but it increased the computational burden of the method. On the other hand, the human cognitive system can easily understand the region-wise haze-concentration of an image without any additional information. So, by exploiting this property of human cognitive system, a set of novel FI rules (Table 3.1.1.3.1.1.) is designed to perform pixel-wise contrast enhancement of hazy images depending upon each pixel's nature of degradation. Moreover, using the human cognitive ability to differentiate between noisy and edge pixels, one set of novel FI rules is designed to perform simultaneous edge & noise detection in combination with the BFO algorithm. These rules also evaluate edge-strengths of all possible edge directions located within each image patch. BFO algorithm is used in this work to determine the most suitable direction for the movement of bacteria within each image patch. It also facilitates the selection of true edge pixels by imposing certain constraints (given in Section 3.1.2.4) on defuzzified edge-strengths which are obtained by performing defuzzification of the fuzzy membership values of output linguistic variable (D_0) of FI rules which are designed for performing edge & noise detection. This reduces chances of false edge detection. In addition to these rules, two different sets of novel FI rules are also designed to perform removal of noise present in images and sharpening of selected edges respectively. The proposed FI rules based noise filtering and edge-sharpening methods also overcome the shortcomings of existing noise removal filters and un-sharp masking method. The different sets of novel FI rules are exclusively designed to process each pixel according to its' properties at each step of the proposed dehazing method and also enhances the robustness of the method.

- **BF algorithm** [30] is chosen in this work to perform edge & noise detection in combination with FI rules because of its advantages over other bio-inspired optimization algorithms. The search strategy of BF is beneficial as unlike Genetic Algorithm, it does not assume the edges to be continuous in nature, rather it searches for edges in the whole search space. Moreover, its capability to sense, decide and act also makes it superior compared to Ant Colony optimization and Particle Swarm optimization algorithms.

HSV color space is selected in this work to carry out dehazing of images in contrary to other color spaces due to its robustness towards color distortion and its resemblance with the human vision system. Among H (hue), S (saturation) and V (value) channels of a hazy image, here the V channel is chosen to conduct dehazing as it is the only channel which satisfies the atmospheric scattering model (hazy image model) as proved in [56-57].

In this method, input hazy images are initially converted from RGB to HSV color space prior to performing image dehazing using the following equations:

$$H' = \begin{cases} \frac{G-B}{C} \text{mod} 6 & \text{if } M = R \\ \frac{B-R}{C} + 2 & \text{if } M = G \\ \frac{R-G}{C} + 4 & \text{if } M = B \end{cases} \quad (3.1.1)$$

$$H = 60 \times H' \quad (3.1.2)$$

$$V = M \quad (3.1.3)$$

$$S = \frac{M - m}{M} = \frac{C}{V} \quad (3.1.4)$$

In (3.1.1) – (3.1.4), $M = \max(R, G, B)$, $m = \min(R, G, B)$, $C = M - m$

The authors in [56-57] have proved that H channel of hazy image does not undergo any degradation due to the presence of haze, hence it needs no processing.

The authors of these works have mathematically proved that V channel is the only channel that completely satisfies the atmospheric scattering model.

Based on these findings, the proposed dehazing method is designed mainly to restore the degraded V channel (V_{old}) of hazy images.

Mathematically, V_{old} is described as follows:

$$V_{old}(i, j) = V_{new}(i, j)t(i, j) + A(1 - t(i, j)) \quad (3.1.5)$$

Comparing (1.1.1) and (3.1.6), it was concluded that the V channel of degraded hazy image completely satisfies the atmospheric scattering model. Here, the main aim is to recover V_{new} (the restored V channel) from V_{old} (the degraded V channel) using the following steps:

3.1.1. FI rules based contrast enhancement

This is the first step of the proposed method. In this step, contrast enhancement of each pixel of a hazy image is performed depending upon its extent of degradation using a set of novel FI rules (given in Table. 3.1.1.3.1.1) which are designed considering haze-concentration (HC) and Log-sigmoid transformation function (LSF) as input fuzzy linguistic variables and mapping constant (MP) as output fuzzy linguistic variable. The reasons behind choosing HC and LSF as input linguistic variables of the designed FI rules are given in Sections 3.1.1.1 and 3.1.1.2 respectively.

3.1.1.1. Reasons behind choosing HC as an input linguistic variable of the designed FI rules

The authors in [38] have stated that under the influence of additive airlight, the brightness of pixels belonging to hazy images increase and saturation decrease and hence, HC of a pixel is estimated from the difference between the magnitudes of its' brightness and saturation values i.e. ($HC(i, j) \propto v(i, j) - s(i, j)$). In other words, it is said that HC of a pixel is directly proportional to the difference between the magnitudes of its' brightness and saturation values i.e. ($HC(i, j) \propto v(i, j) - s(i, j)$).

From this observation, the following conclusions are drawn:

1. More the influence of additive airlight on a pixel \rightarrow More will be the difference between the magnitudes of its' brightness and saturation values \rightarrow More will be its HC \rightarrow More will be its visibility degradation.
2. Less the influence of additive airlight on a pixel \rightarrow Less will be the difference between the magnitudes of its' brightness and saturation values \rightarrow Less will be its HC \rightarrow Less will be its visibility degradation.

Hence, the pixels undergoing more degradation should be more contrast enhanced compared to the pixels undergoing less degradation. That is why, in this work larger MP values are used to map the pixels possessing higher HC as HC value of each pixel gives a direct estimation of its extent of degradation. For this reason, here HC is considered as one of the linguistic variables of the FI rules which are designed for performing contrast enhancement.

3.1.1.2. Reasons behind choosing LSF as an input linguistic variable of the designed FI rules

In this work, LSF is chosen as a transformation function to map the pixels belonging to the degraded V channel of a hazy image (V_{old}) to the corresponding contrast enhanced V channel (V') which is generated using the proposed FI rules based contrast enhancement method. As the objective here is to compress the high intensity range of images (as the pixels corrupted with dense haze usually possess high intensities) and to expand the low intensity range of images (as the haze free pixels usually possess low intensities), LSF is chosen as an ideal transformation function for this purpose as the basic objective of log transformation function is to compress the high intensity range of images and to expand their low intensity range [94]. Due to the significant influence of LSF on the entire contrast enhancement process, it is considered as one of the input linguistic values of the FI rules which are designed for performing contrast enhancement. LSF is calculated here using the following mathematical equation:

$$LSF(i, j) = 1/(1 + \exp(-HC(i, j))) \quad (3.1.1.2.1)$$

3.1.1.3. Details of designed FI rules based contrast enhancement method

Table 3.1.1.3.1. FI rules designed for performing contrast enhancement

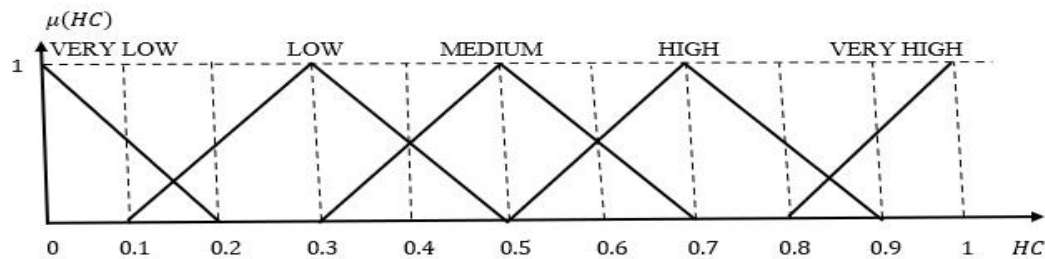
Rule No	<i>If</i>		<i>Then</i>
	HC	LSF	MP
1	VERY LOW	LESS	VERY SMALL
2	VERY LOW	AVERAGE	VERY SMALL
3	VERY LOW	HIGH	SMALL
4	LOW	LESS	SMALL
5	LOW	AVERAGE	SMALL
6	LOW	HIGH	AVERAGE
7	MEDIUM	LESS	AVEAGE
8	MEDIUM	AVERAGE	AVEAGE
9	MEDIUM	HIGH	LARGE
10	HIGH	LESS	LARGE
11	HIGH	AVERAGE	LARGE
12	HIGH	HIGH	LARGE
13	VERY HIGH	LESS	LARGE
14	VERY HIGH	AVERAGE	VERY LARGE
15	VERY HIGH	HIGH	VERY LARGE

3.1.1.3.1.1. Dynamic Range(DR) of linguistic variables used in these FI rules

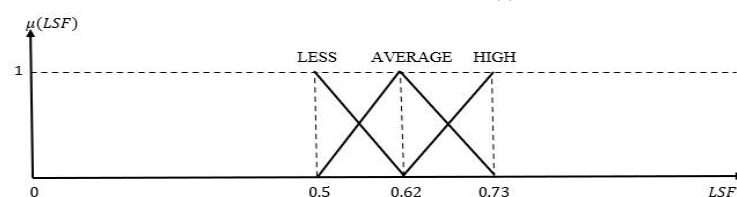
DR of HC= [0 1]. Prior to performing colour space conversion of hazy images (converting hazy images from RGB colour space to HSV colour space), pixel intensity range is restricted within [0 1] due to normalization. So, after conversion, the values of V_{old} and S_{old} will also lie between [0 1] according to (3.1.3) and (3.1.4). As in hazy images, it is proved that $V_{old} \geq S_{old}$ due to the influence of additive airlight, so $(V_{old} - S_{old})$ will always lie between [0 1].

DR of LSF= [0.5 0.73]. Lower bound= $\frac{1}{1+e^{-0}}=0.5$. Upper bound= $\frac{1}{1+e^{-1}}=0.73$. (Since, DR of HC lies between [0 1]).

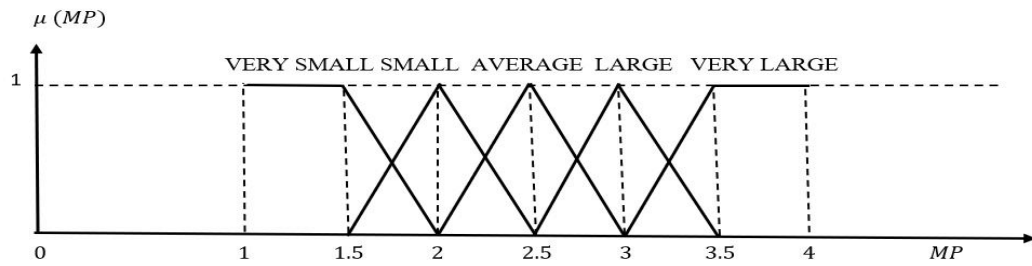
DR of MP = [1 4]. (Empirically chosen).



(a)

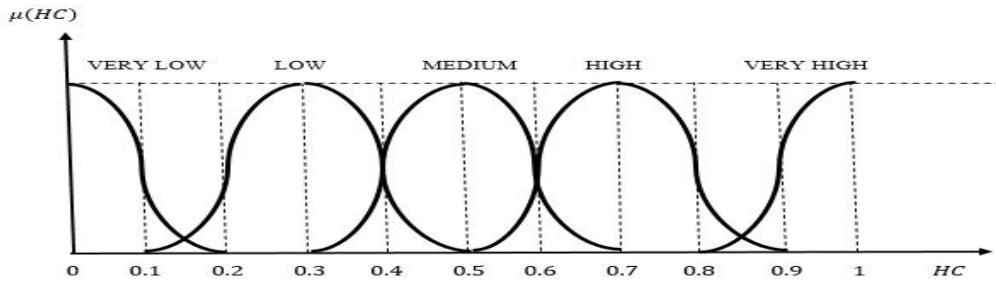


(b)

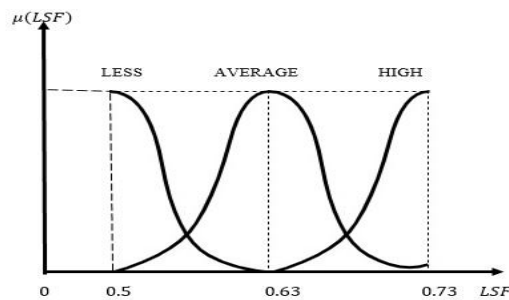


(c)

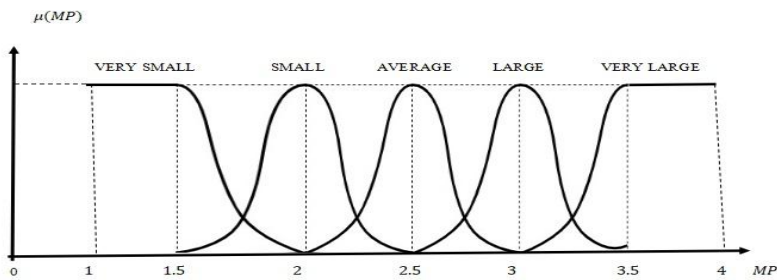
SET A



(a)



(b)



(c)

SET B

Fig.3.1.1.3.1. Graphical representations of fuzzy membership functions used to represent linguistic variables (a) *HC*, (b) *LSF* and (c) *MP*

**Note: In this proposed method, the experimental analyses are carried out twice, once using Triangular membership function (given in Set A) and once using Gaussian membership function (given in Set B) to empirically choose the most suitable membership function for representing the linguistic variables.*

After obtaining unique defuzzified output linguistic value, ' MP^* ' value for mapping each pixel belonging to the degraded V channel of a hazy image (V_{old}) to the corresponding contrast enhanced V channel (V'), contrast enhancement is performed using the following equation:

$$V'(i, j) = V_{old}(i, j) \times LSF(i, j) \times MP^*(i, j) \quad (3.1.1.3.1.2.1)$$

3.1.2. Novel BF-Fuzzy synergism based simultaneous edge & noise detection

This is the second step of the proposed method which deals with simultaneous edge & noise detection as well as noise removal using different sets of FI rules and BFO algorithm.

The BFO algorithm proposed by the authors in [30] is a powerful bio-inspired optimization algorithm. It can replicate the behavior of a single bacterium or groups of E. Coli bacteria.

BFO algorithm is used in this method to perform simultaneous edge & noise detection in combination with one set of novel FI rules. In this work, bacteria simultaneously detect both edge pixels and noisy pixels encountered by them on their path while foraging across the V' channel. BFO algorithm in combination with the FI rules determine the most suitable direction for the movement of bacteria within each 3×3 image patch and also facilitate the selection of true edges by imposing some constraints on the defuzzified edge-strengths obtained for all the eight possible edge directions within each image patch and thus eliminates any chances of false edge detection. BFO algorithm, used here comprises of three steps namely, chemotaxis, reproduction, and elimination and dispersal. The swarming step of the traditional BFO algorithm is eliminated here to reduce the computational burden as it is of no use in this application since the main aim in this work is to select most suitable edge within each image patch.

The idea of using BFO algorithm in combination with Fuzzy Logic to perform edge & noise detection in this method is partially inspired from the edge detection method proposed in [95] where the authors have also used BFO algorithm in combination with Fuzzy logic to detect edges but that method has several drawbacks which are highlighted as follows:

- In [95], the authors have performed edge detection using FI rules which they have designed considering the input linguistic variables (Fuzzy Derivative (FD) values ((FD value along any direction say, D is obtained by subtracting the pixel intensity of the centre pixel of an 3×3 image patch and the pixel located at direction ' D ' within that patch)) of the FI rules to always belong to a fuzzy set (Large). They have not considered other possible cases. This leads to erroneous edge detection and also makes the method unsuitable for practical implementation. Moreover, they have not provided any information regarding the DR of linguistic variables considered while designing the FI rules.
- They have performed edge detection considering FD values as input linguistic variables which give incorrect results because edges mean sharp transitions of pixel intensities (the transition can be from both low intensity to high intensity level or from high intensity to low intensity level), so directions possessing high membership values in 'Large' fuzzy set i.e. the edges undergoing high intensity to low intensity transitions will only be detected as edges using the method proposed in [95] since their FD values will only be positive and high. Edges having low intensity to high intensity transitions will not be detected as edges as their FD values will be negative and low.
- FI rules designed in [95] can neither detect noisy pixels nor can evaluate edge-strengths of all possible edge directions.

All these shortcomings are solved in this proposed method by designing a completely novel set of FI rules which can perform simultaneous edge & noise detection and can also evaluate edge-strength of each possible edge direction located within an image-patch. The FI rules are designed in this work by incorporating all the possible combinations which may arise during edge & noise detection to avoid any ambiguity. BFO algorithm is used in this work in combination with the FI rules to prevent false edge detection. It is used to determine the most suitable direction for the movement of bacteria within each image patch and to select true edges by imposing constraints on defuzzified edge-strengths which are evaluated for all possible eight edge directions located within an image patch. Moreover, the

drawbacks arising due to the use of FD values as input linguistic variables of the FI rules in [95] are also solved by using $|FD|$ values (magnitudes of FD values) as input linguistic variables of the novel FI rules designed in this work. The use $|FD|$ values of desired directions as input linguistic variables of FI rules facilitate the detection of the sharp transitions of pixel intensities (both from low intensity to high intensity level or from high intensity to low intensity level) as edges. The proposed BF-Fuzzy synergism based approach is completely novel as it differs from the method proposed in [95] both in terms of FI rules and methodology.

After detecting the noisy pixels, they are filtered using a novel set of FI rules based noise removal method which is designed to remove noise from each detected pixel in accordance to its extent of corruption w.r.t each neighbouring pixel located within its 3×3 neighbourhood to completely nullify any chances of occurrence of undesired edge-blurring in output images. A detailed comparative analysis of the proposed FI rules based noise removal method w.r.t several popular image denoising methods is also performed in **Section 3.1.2.5** to prove the method's efficiency. Similar comparative analysis is also carried out to prove the efficiency of the proposed BF-Fuzzy synergism based simultaneous edge & noise detection method over traditional edge-detectors in **Section 3.1.2.6**.

3.1.2.1. Search space

In this problem, bacteria forage across the entire two-dimensional search space consisting of x and y co-ordinates of pixels. Limited by the image dimensions, the search space is finite and it takes only discrete values. Bacteria are placed at random positions in the enhanced two-dimensional V' channel.

3.1.2.2. Chemotaxis

Edges signify sharp changes in pixel intensities, so traditional edge detectors often perform edge detection considering the differences in intensities of neighboring pixels which often lead to the production of erroneous results as like edge pixels, the intensities of noisy pixels too vastly differ w.r.t the intensities of neighboring pixels. Only the pixels lying in the homogeneous regions of images possess more or less similar intensities. It is also observed that in case of true edge pixels, the intensities of the neighboring pixels lying perpendicular to the edge directions vary vastly w.r.t to the intensities of the edge pixels. But this observation does not hold for noisy pixels as noise corrupts pixels randomly.

So, keeping these facts in mind, in this work a set of novel FI rules is designed to perform simultaneous edge & noise detection. The constraints (given in **Section 3.1.2.6**) imposed on the defuzzified edge-strengths obtained after performing defuzzification of the fuzzy membership values of the output linguistic variable (D_0) of these rules facilitate the selection of true edge pixels and prevent any chances of false edge detection.

In this step, initially the enhanced V' channel is divided into several overlapping 3×3 image patches. Eight possible directions within each image patch, $D_0 \in \{N, S, W, E, NE, SE, NW, SW\}$ are shown in Fig 3.1.2.2.1. Pixels considered for performing edge & noise detection along each possible edge direction say, D_0 within a 3×3 patch centered at pixel location, (x, y) is pictorially represented in Fig.3.1.2.2.2 and mathematically in (3.1.2.2.1) – (3.1.2.2.40).

NW	N	NE
W	(x, y)	E
SW	S	SE

Fig 3.1.2.2.1. Eight possible edge-directions within a 3×3 patch

$|FD|$ values of pixels considered for performing edge & noise detection along all possible edge directions within a 3×3 patch are mathematically represented as follows:

For $D_0=N$:

$$|\nabla_N(x, y)| = |I(x, y - 1) - I(x, y)| \quad (3.1.2.2.1)$$

$$|\nabla_W(x, y)| = |I(x - 1, y) - I(x, y)| \quad (3.1.2.2.2)$$

$$|\nabla_E(x, y)| = |I(x + 1, y) - I(x, y)| \quad (3.1.2.2.3)$$

$$|\nabla_W(x, y - 1)| = |I(x - 1, y - 1) - I(x, y - 1)| \quad (3.1.2.2.4)$$

$$|\nabla_E(x, y - 1)| = |I(x + 1, y - 1) - I(x, y - 1)| \quad (3.1.2.2.5)$$

For $D_0=S$:

$$|\nabla_S(x, y)| = |I(x, y + 1) - I(x, y)| \quad (3.1.2.2.6)$$

$$|\nabla_W(x, y)| = |I(x - 1, y) - I(x, y)| \quad (3.1.2.2.7)$$

$$|\nabla_E(x, y)| = |I(x + 1, y) - I(x, y)| \quad (3.1.2.2.8)$$

$$|\nabla_W(x, y + 1)| = |I(x - 1, y + 1) - I(x, y + 1)| \quad (3.1.2.2.9)$$

$$|\nabla_E(x, y + 1)| = |I(x + 1, y + 1) - I(x, y + 1)| \quad (3.1.2.2.10)$$

For $D_0=W$:

$$|\nabla_W(x, y)| = |I(x - 1, y) - I(x, y)| \quad (3.1.2.2.11)$$

$$|\nabla_N(x, y)| = |I(x, y - 1) - I(x, y)| \quad (3.1.2.2.12)$$

$$|\nabla_S(x, y)| = |I(x, y + 1) - I(x, y)| \quad (3.1.2.2.13)$$

$$|\nabla_N(x - 1, y)| = |I(x - 1, y - 1) - I(x - 1, y)| \quad (3.1.2.2.14)$$

$$|\nabla_S(x - 1, y)| = |I(x - 1, y + 1) - I(x - 1, y)| \quad (3.1.2.2.15)$$

For $D_0=E$:

$$|\nabla_E(x, y)| = |I(x + 1, y) - I(x, y)| \quad (3.1.2.2.16)$$

$$|\nabla_N(x, y)| = |I(x, y - 1) - I(x, y)| \quad (3.1.2.2.17)$$

$$|\nabla_S(x, y)| = |I(x, y + 1) - I(x, y)| \quad (3.1.2.2.18)$$

$$|\nabla_N(x + 1, y)| = |I(x + 1, y - 1) - I(x + 1, y)| \quad (3.1.2.2.19)$$

$$|\nabla_S(x + 1, y)| = |I(x + 1, y + 1) - I(x + 1, y)| \quad (3.1.2.2.20)$$

For $D_0=NW$:

$$|\nabla_{NW}(x, y)| = |I(x - 1, y - 1) - I(x, y)| \quad (3.1.2.2.21)$$

$$|\nabla_{NE}(x, y)| = |I(x + 1, y - 1) - I(x, y)| \quad (3.1.2.2.22)$$

$$|\nabla_{SW}(x, y)| = |I(x - 1, y + 1) - I(x, y)| \quad (3.1.2.2.23)$$

$$|\nabla_{NE}(x - 1, y - 1)| = |I(x, y - 2) - I(x - 1, y - 1)| \quad (3.1.2.2.24)$$

$$|\nabla_{SW}(x - 1, y - 1)| = |I(x - 2, y) - I(x - 1, y - 1)| \quad (3.1.2.2.25)$$

For $D_0=NE$:

$$|\nabla_{NE}(x, y)| = |I(x + 1, y - 1) - I(x, y)| \quad (3.1.2.2.26)$$

$$|\nabla_{NW}(x, y)| = |I(x - 1, y - 1) - I(x, y)| \quad (3.1.2.2.27)$$

$$|\nabla_{SE}(x, y)| = |I(x + 1, y + 1) - I(x, y)| \quad (3.1.2.2.28)$$

$$|\nabla_{NW}(x + 1, y - 1)| = |I(x, y - 2) - I(x + 1, y - 1)| \quad (3.1.2.2.29)$$

$$|\nabla_{SE}(x + 1, y - 1)| = |I(x + 2, y) - I(x + 1, y - 1)| \quad (3.1.2.2.30)$$

For $D_0=SW$:

$$|\nabla_{SW}(x, y)| = |I(x - 1, y + 1) - I(x, y)| \quad (3.1.2.2.31)$$

$$|\nabla_{NW}(x, y)| = |I(x - 1, y - 1) - I(x, y)| \quad (3.1.2.2.32)$$

$$|\nabla_{SE}(x, y)| = |I(x + 1, y + 1) - I(x, y)| \quad (3.1.2.2.33)$$

$$|\nabla_{NW}(x - 1, y + 1)| = |I(x - 2, y) - I(x - 1, y + 1)| \quad (3.1.2.2.34)$$

$$|\nabla_{SE}(x - 1, y + 1)| = |I(x, y + 2) - I(x - 1, y + 1)| \quad (3.1.2.2.35)$$

For $D_0=SE$:

$$|\nabla_{SE}(x, y)| = |I(x + 1, y + 1) - I(x, y)| \quad (3.1.2.2.36)$$

$$|\nabla_{NE}(x, y)| = |I(x + 1, y - 1) - I(x, y)| \quad (3.1.2.2.37)$$

$$|\nabla_{SW}(x, y)| = |I(x - 1, y + 1) - I(x, y)| \quad (3.1.2.2.38)$$

$$|\nabla_{NE}(x + 1, y + 1)| = |I(x + 2, y) - I(x + 1, y + 1)| \quad (3.1.2.2.39)$$

$$|\nabla_{SW}(x + 1, y + 1)| = |I(x, y + 2) - I(x + 1, y + 1)| \quad (3.1.2.2.40)$$

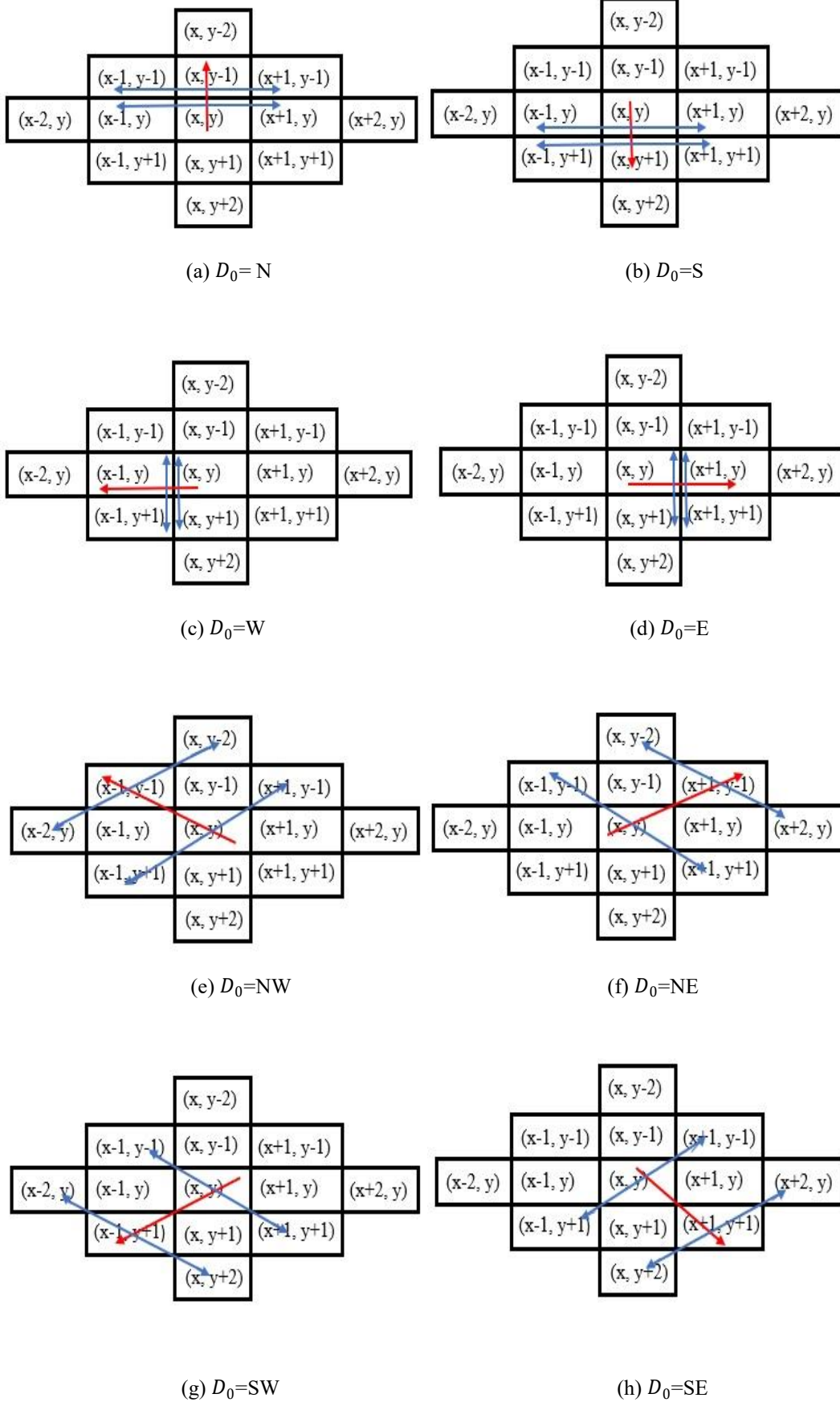


Fig.3.1.2.2.2. (a) N, (b) S, (c) W, (d) E, (e) NW, (f) NE, (g) SW, (h) SE (Red arrows represent probable edge directions, blue arrows signify locations of pixels lying perpendicular to probable edge directions)

For each possible edge direction, (D_0) five equations are obtained where the first equation is designed to calculate the FD value along the possible edge-direction, (D_0) and the other four equations are designed to calculate the FD values of the four neighboring pixels which lie perpendicular to D_0 . Here with the help of an example (Fig. 3.1.2.2.3. where $D_0 = N$ (North)) the logic following which these five equations are obtained for each D_0 is explained.

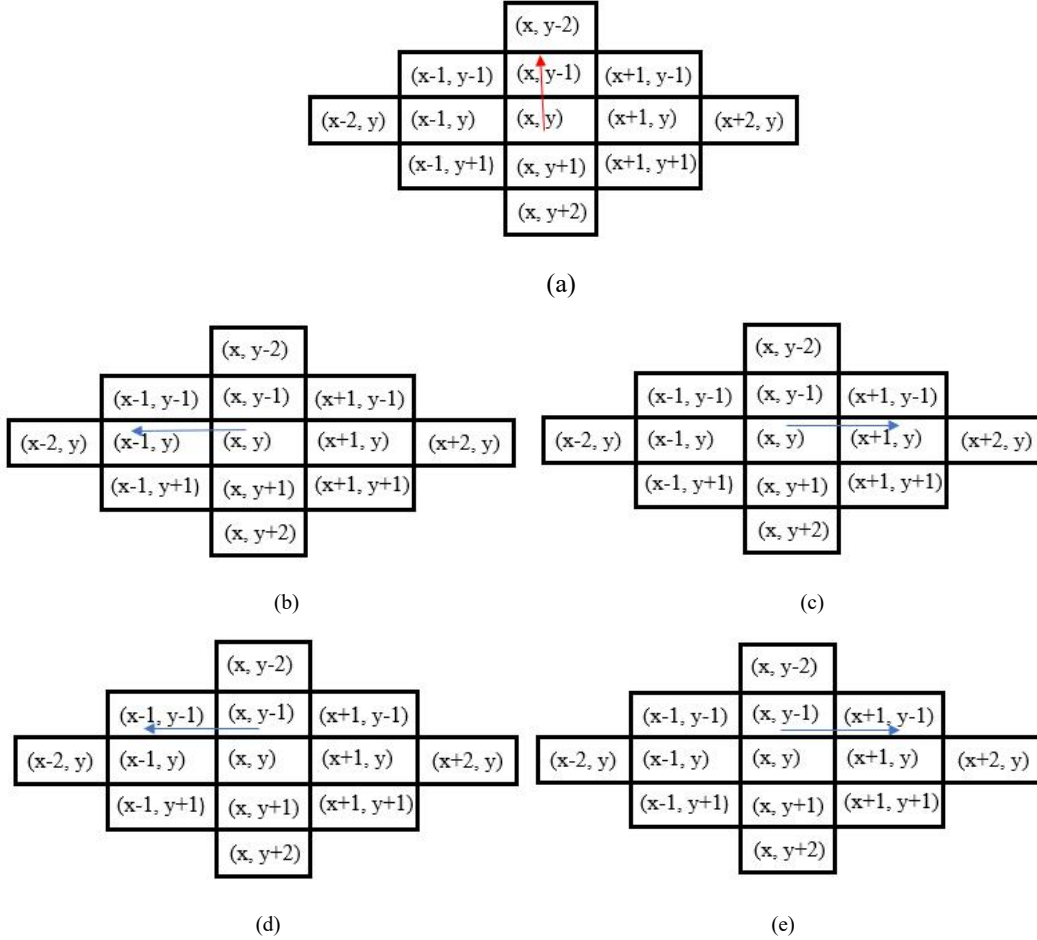


Fig. 3.1.2.2.3. (a) Possible edge-direction (marked with red arrow) and (b)-(e) Neighboring pixels which lie perpendicular to the possible edge-direction (marked with blue arrows)

From Fig. 3.1.2.2.3 (a) , (3.1.2.2.41) is designed:

$$\nabla_N(x, y) = I(x, y - 1) - I(x, y) \quad (3.1.2.2.41)$$

Form (3.1.2.2.41), it is concluded that if an edge exists along $D_0 = N$, then $I(x, y - 1)$ and $I(x, y)$ are possible edge pixels.

From Fig. 3.1.2.2.3 (b) , (3.1.2.2.42) is designed:

$$\nabla_W(x, y) = I(x - 1, y) - I(x, y) \quad (3.1.2.2.42)$$

As the pixel, $I(x - 1, y)$ lies at a pixel position which is perpendicular to $D_0 = N$ and as it is located towards the West direction of possible edge pixel, $I(x, y)$, so the corresponding FD value is marked as $\nabla_W(x, y)$ to denote the concerned neighboring pixel lies towards the West direction of possible edge pixel located at (x, y) .

From Fig. 3.1.2.2.3 (c), (3.1.2.2.43) is designed:

$$\nabla_E(x, y) = I(x + 1, y) - I(x, y) \quad (3.1.2.2.43)$$

As the pixel, $I(x + 1, y)$ lies at a pixel position which is perpendicular to $D_0 = N$ and as it is located towards the East direction of possible edge pixel, $I(x, y)$, so the corresponding FD value is marked as $\nabla_E(x, y)$ to denote the concerned neighboring pixel lies towards the East direction of possible edge pixel located at (x, y) .

From Fig. 3.1.2.2.3 (d), (3.1.2.2.44) is designed:

$$\nabla_W(x, y - 1) = I(x - 1, y - 1) - I(x, y - 1) \quad (3.1.2.2.44)$$

As the pixel, $I(x - 1, y - 1)$ lies at a position which is perpendicular to $D_0 = N$ and as it is located towards the West direction of possible edge pixel, $I(x, y - 1)$, so the corresponding FD value is marked as $\nabla_W(x, y - 1)$ to denote the concerned neighboring pixel lies towards the West of possible edge pixel located at $(x, y - 1)$.

From Fig. 3.1.2.2.3 (e), (3.1.2.2.45) is designed:

$$\nabla_E(x, y - 1) = I(x + 1, y - 1) - I(x, y - 1) \quad (3.1.2.2.45)$$

As the pixel, $I(x + 1, y - 1)$ lies at a position which is perpendicular to $D_0 = N$ and as it is located towards the East direction of possible edge pixel, $I(x, y - 1)$, so the corresponding FD value is marked as $\nabla_E(x, y - 1)$ to denote the concerned neighboring pixel lies towards the East of the possible edge pixel located at $(x, y - 1)$.

The equations to calculate the FD values for other possible edge directions are also designed in similar way. The proposed BF-Fuzzy synergism based edge & noise detection method performs detection of edge pixels and noisy pixels considering $|FD|$ (magnitude of FD values) as linguistic variables of the designed FI rules given in Table 3.1.2.2.1.

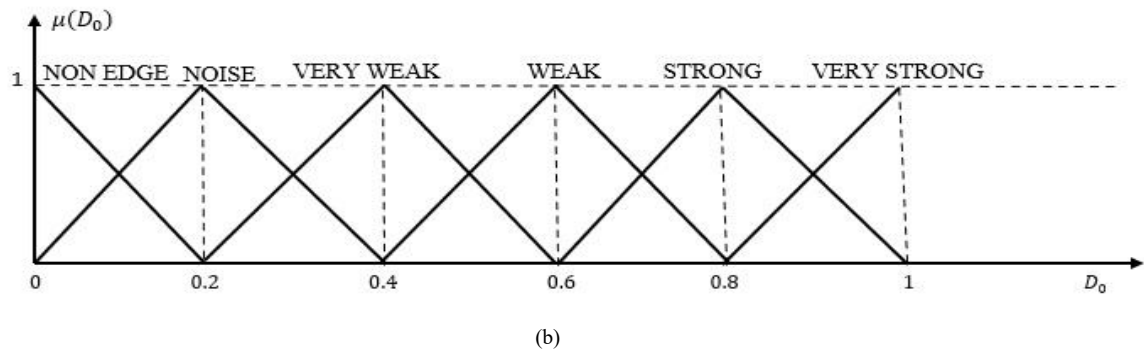
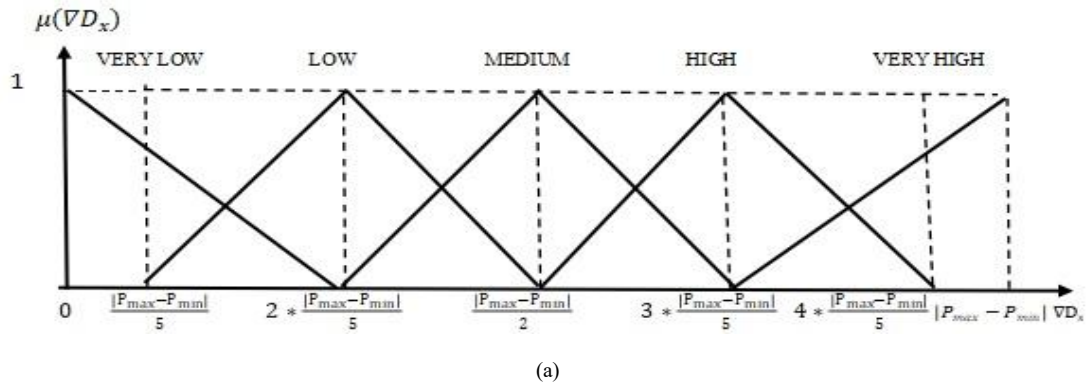
Table 3.1.2.2.1. FI rules designed for performing simultaneous edge& noise detection

Rule	<i>If</i>					<i>Then</i>
	$ \nabla D_0(p, q) $ (Magnitude of FD along possible edge-direction, D_0)	$ \nabla D_1(p, q) $ (Magnitude of FD along direction, D_1 which lies perpendicular to possible edge-direction, D_0)	$ \nabla D_2(p, q) $ (Magnitude of FD along direction, D_2 which lies perpendicular to possible edge-direction, D_0)	$ \nabla D_3(p, q) $ (Magnitude of FD along direction, D_3 which lies perpendicular to possible edge-direction, D_0)	$ \nabla D_4(p, q) $ (Magnitude of FD along direction, D_4 which lies perpendicular to possible edge-direction, D_0)	D_0 (Possible edge-strength along possible edge-direction, D_0)
1	VERY LOW	VERY LOW	VERY LOW	VERY LOW	VERY LOW	NON-EDGE
2	LOW	VERY LOW	VERY LOW	VERY LOW	VERY LOW	NON-EDGE
3	MEDIUM	VERY LOW	VERY LOW	VERY LOW	VERY LOW	NON-EDGE
4	HIGH	VERY LOW	VERY LOW	VERY LOW	VERY LOW	NOISE
5	VERY HIGH	VERY LOW	VERY LOW	VERY LOW	VERY LOW	NOISE
6	VERY HIGH	VERY HIGH	VERY HIGH	VERY HIGH	VERY HIGH	VERY STRONG
7	VERY HIGH	VERY HIGH	VERY HIGH	VERY HIGH	HIGH	VERY STRONG
8	VERY HIGH	VERY HIGH	VERY HIGH	HIGH	HIGH	VERY STRONG
9	VERY HIGH	VERY HIGH	HIGH	HIGH	HIGH	STRONG
10	VERY HIGH	HIGH	HIGH	HIGH	HIGH	STRONG
11	HIGH	HIGH	HIGH	HIGH	HIGH	STRONG
12	HIGH	HIGH	HIGH	HIGH	MEDIUM	STRONG
13	HIGH	HIGH	HIGH	MEDIUM	MEDIUM	STRONG
14	HIGH	HIGH	MEDIUM	MEDIUM	MEDIUM	WEAK
15	HIGH	MEDIUM	MEDIUM	MEDIUM	MEDIUM	NOISE
16	MEDIUM	MEDIUM	MEDIUM	MEDIUM	MEDIUM	NON-EDGE
17	VERY LOW	MEDIUM	MEDIUM	MEDIUM	MEDIUM	NON-EDGE
18	VERY LOW	LOW	MEDIUM	MEDIUM	MEDIUM	NON-EDGE
19	VERY LOW	LOW	LOW	MEDIUM	MEDIUM	NON-EDGE
20	VERY LOW	LOW	LOW	LOW	MEDIUM	NON-EDGE
21	VERY LOW	LOW	LOW	LOW	LOW	NON-EDGE
22	VERY HIGH	MEDIUM	MEDIUM	MEDIUM	MEDIUM	NOISE
23	VERY HIGH	VERY HIGH	HIGH	MEDIUM	MEDIUM	VERY WEAK
24	VERY HIGH	VERY HIGH	HIGH	HIGH	MEDIUM	WEAK
25	VERY HIGH	VERY HIGH	VERY HIGH	VERY HIGH	MEDIUM	VERY STRONG
26	HIGH	HIGH	HIGH	HIGH	MEDIUM	STRONG
27	HIGH	HIGH	MEDIUM	LOW	MEDIUM	VERY WEAK

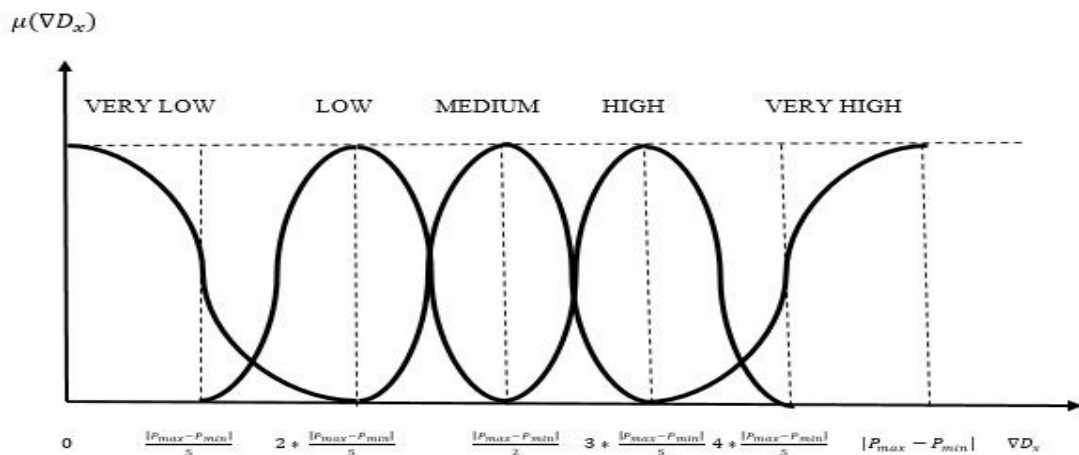
28	VERY LOW	LOW	VERY LOW	VERY LOW	VERY LOW	NON-EDGE
29	HIGH	HIGH	LOW	LOW	MEDIUM	VERY WEAK
30	HIGH	HIGH	LOW	LOW	LOW	VERY WEAK

Example: PR_1 (FI rule: 1): IF $|\nabla_{D_0}(p, q)|$ is **VERY LOW** and $|\nabla_{D_1}(p, q)|$ is **VERY LOW** and $|\nabla_{D_2}(p, q)|$ is **VERY LOW** and $|\nabla_{D_3}(p, q)|$ is **VERY LOW** and $|\nabla_{D_4}(p, q)|$ is **VERY LOW** THEN D_0 is **NON EDGE**.

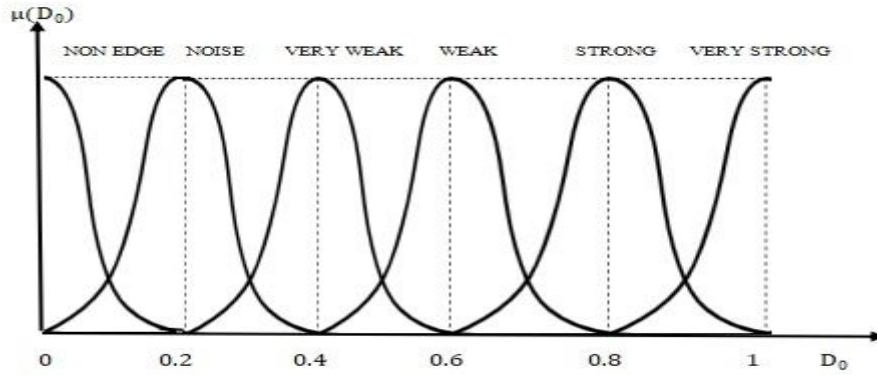
*Note: PR_1 (below Table..3.1.2.2.1.) is given just as an example to help the readers understand the way of representation of FI rules .



SET A



(a)



(b)

SET B

Fig. 3.1.2.2.4. Graphical representations of fuzzy membership functions used for representing linguistic variables of FI rules designed for edge & noise detection (a) $|\nabla D_x|$ and (b) D_0

P_{min} and P_{max} represent the minimum and maximum intensities of all the pixels belonging to the enhanced two-dimensional V' channel.

Let the dimension of the enhanced two-dimensional V' channel be $m \times n$.

$$P_{min} = \min (I(1,1), I(1,2), \dots, I(m, n - 1), I(m, n)) \quad (3.1.2.2.46)$$

$$P_{max} = \max (I(1,1), I(1,2), \dots, I(m, n - 1), I(m, n)) \quad (3.1.2.2.47)$$

Where, $I(i, j)$: Intensity of pixel located at the i^{th} row and j^{th} column of the two-dimensional V' channel.

In Fig.3.1.2.2.4. (a), DR of the input linguistic variable is chosen to be $[0, |P_{min} - P_{max}|]$ to cover the entire $|FD|$ range. In Fig.3.1.2.2.4. (a), DR of the output linguistic variable is chosen to be $[0 1]$, so that the defuzzified value of edge-strength estimated for each possible edge-direction remains restricted within this range.

The FI rules designed for edge & noise detection (given in Table.3.1.2.2.1.) are fired for each of the eight possible edge directions within a 3×3 image patch. After performing defuzzification of the output linguistic variable, D_0 of these rules, eight dd_0 (defuzzified edge-strength of any edge-direction say, D_0) values (one for each direction) are obtained. Among these eight possible directions, if the dd_0 value of any direction suggests that it possess the highest membership value in the **NOISE** fuzzy set, then noise removal is done using the following procedure.

Let the dd_0 value of any direction say, ' D_x ' suggests that it possess the highest membership value in the **NOISE** fuzzy set. Since, in this work, the estimation of each $|FD|$ involves two pixels hence, a certain edge-direction can possess the highest membership value in the **NOISE** fuzzy set in two cases as explained below.

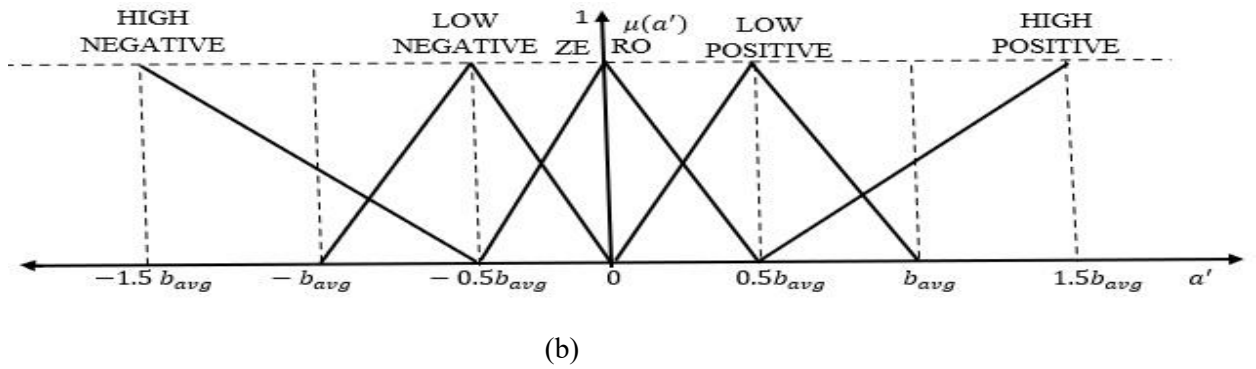
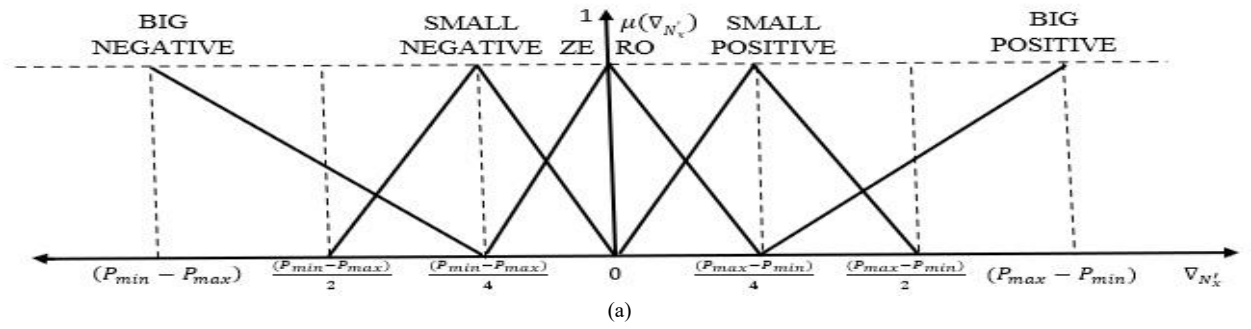
$$|\nabla_{D_x}(i, j)| = |I(s, t) - I(i, j)| \quad (3.1.2.2.48)$$

Where, (s, t) and (i, j) denote the pixel locations. The pixel located at (s, t) exists towards the D_x direction of the center pixel located at (i, j) . $|\nabla_{D_x}(i, j)|$ denotes the magnitude of the difference of $I(s, t)$ and $I(i, j)$. So, $|\nabla_{D_x}(i, j)|$ gets influenced when either of these pixels get corrupted with noise.

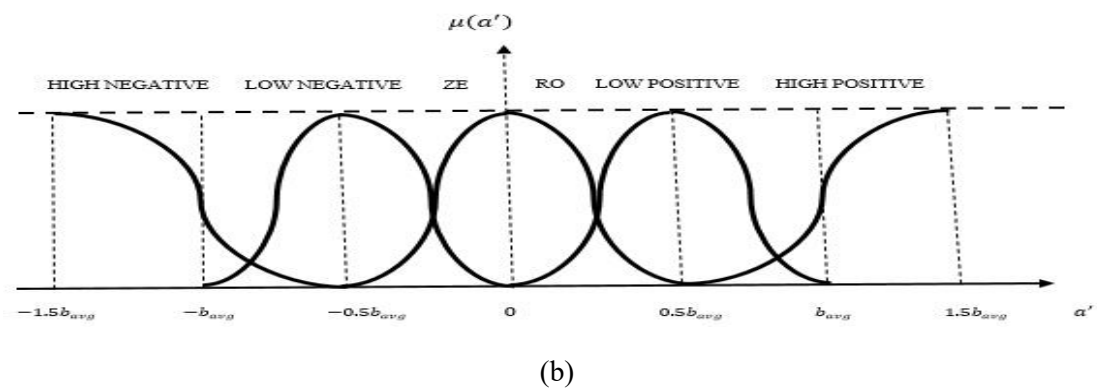
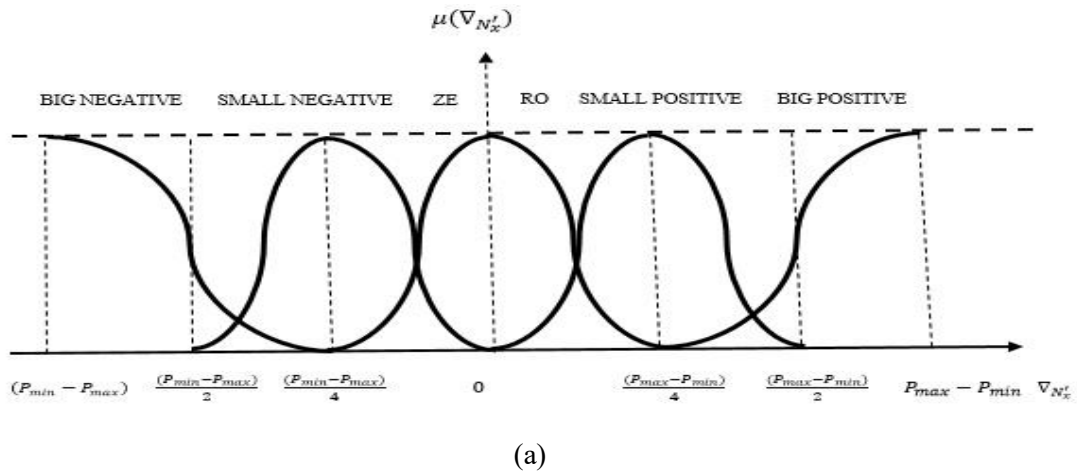
Here, initially considering the pixel located in the D_x direction as noisy pixel, it is filtered using the set of novel FI rules (given in Table.3.1.2.2.2.) which are designed for performing noise removal.

Table. 3.1.2.2.2. FI rules designed for filtering noisy pixels

Rule	If								$Then$
	∇_{N_0} (FD along direction D_0)	∇_{N_1} (FD of noisy pixel w.r.t Neighbour 1)	∇_{N_2} (FD of noisy pixel w.r.t Neighbour 2)	∇_{N_3} (FD of noisy pixel w.r.t Neighbour 3)	∇_{N_4} (FD of noisy pixel w.r.t Neighbour 4)	∇_{N_5} (FD of noisy pixel w.r.t Neighbour 5)	∇_{N_6} (FD of noisy pixel w.r.t Neighbour 6)	∇_{N_7} (FD of noisy pixel w.r.t Neighbour 7)	a' (Correction term)
1	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	HIGH NEGATIVE
2	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG NEGATIVE	BIG NEGATIVE	HIGH POSITIVE
3	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	ZERO
4	BIG NEGATIVE	SMALL NEGATIVE	SMALL NEGATIVE	SMALL NEGATIVE	SMALL NEGATIVE	SMALL POSITIVE	SMALL POSITIVE	SMALL POSITIVE	LOW NEGATIVE
5	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	SMALL NEGATIVE	SMALL NEGATIVE	SMALL NEGATIVE	SMALL NEGATIVE	HIGH NEGATIVE
6	BIG NEGATIVE	SMALL NEGATIVE	SMALL NEGATIVE	SMALL NEGATIVE	SMALL NEGATIVE	SMALL NEGATIVE	SMALL NEGATIVE	BIG NEGATIVE	LOW NEGATIVE
7	BIG POSITIVE	SMALL POSITIVE	SMALL POSITIVE	SMALL POSITIVE	SMALL POSITIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	LOW POSITIVE
8	BIG NEGATIVE	SMALL NEGATIVE	SMALL NEGATIVE	SMALL NEGATIVE	SMALL NEGATIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	LOW NEGATIVE
9	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	SMALL POSITIVE	SMALL POSITIVE	SMALL POSITIVE	SMALL POSITIVE	HIGH POSITIVE
10	BIG POSITIVE	SMALL POSITIVE	SMALL POSITIVE	SMALL POSITIVE	SMALL POSITIVE	SMALL POSITIVE	BIG POSITIVE	BIG POSITIVE	HIGH POSITIVE
11	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	SMALL POSITIVE	HIGH NEGATIVE
12	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	SMALL NEGATIVE	HIGH POSITIVE
13	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	HIGH NEGATIVE	HIGH POSITIVE
14	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG NEGATIVE	SMALL POSITIVE	HIGH POSITIVE
15	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG NEGATIVE	ZERO	HIGH POSITIVE
16	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	SMALL NEGATIVE	HIGH NEGATIVE
17	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	SMALL POSITIVE	HIGH NEGATIVE
18	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG POSITIVE	HIGH NEGATIVE
19	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	HIGH POSITIVE
20	SMALL POSITIVE	SMALL POSITIVE	SMALL POSITIVE	SMALL POSITIVE	SMALL NEGATIVE	SMALL NEGATIVE	SMALL NEGATIVE	SMALL NEGATIVE	ZERO
21	SMALL NEGATIVE	SMALL NEGATIVE	SMALL NEGATIVE	SMALL NEGATIVE	SMALL POSITIVE	SMALL POSITIVE	SMALL POSITIVE	SMALL POSITIVE	ZERO
22	SMALL POSITIVE	SMALL POSITIVE	SMALL POSITIVE	SMALL POSITIVE	SMALL POSITIVE	SMALL NEGATIVE	SMALL NEGATIVE	SMALL NEGATIVE	LOW POSITIVE
23	SMALL NEGATIVE	SMALL NEGATIVE	SMALL NEGATIVE	SMALL NEGATIVE	SMALL NEGATIVE	SMALL POSITIVE	SMALL POSITIVE	SMALL POSITIVE	LOW NEGATIVE
24	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	ZERO	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	LOW POSITIVE
25	BIG POSITIVE	BIG POSITIVE	BIG POSITIVE	ZERO	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	LOW NEGATIVE
26	BIG POSITIVE	SMALL POSITIVE	SMALL POSITIVE	SMALL POSITIVE	SMALL POSITIVE	BIG NEGATIVE	BIG NEGATIVE	BIG NEGATIVE	LOW POSITIVE



SET A



SET B

Fig. 3.1.2.2.5. Graphical representations of fuzzy membership functions used for representing linguistic variables of FI rules designed for filtering of noisy pixels (a) $\nabla N'_x$ and (b) a'

After filtering the pixel located at D_x direction, the FI rules which are designed for performing edge & noise detection (given in Table 3.1.2.2.1) are fired for ' D_x ' direction only. If the obtained defuzzified value still suggests that ' D_x ' has the highest membership value in the **NOISE** fuzzy set, then the filtered pixel is replaced with its original intensity value (intensity value prior to filtering) and the center pixel ($I(x, y)$) is filtered accordingly. Else, the FI rules designed for edge & noise direction are fired for all eight possible edge directions.

The proposed noise removal procedure is explained here with an example. Suppose *SW* direction (as marked in red in Fig.3.1.2.2.6.) has the highest membership in the **NOISE** fuzzy set then noise removal is done in the following way:

Initially, the pixel located in the *SW* direction ($(x - 1, y + 1)$) is considered to be noisy and it is filtered using the FI rules which are designed for performing noise removal (given in Table. 3.1.2.2.2), as discussed below:

(x-2, y-2)	(x-1, y-2)	(x, y-2)	(x+1, y-2)	(x+2, y-2)
(x-2, y-1)	(x-1, y-1)	(x, y-1)	(x+1, y-1)	(x+2, y-1)
(x-2, y)	(x-1, y)	(x, y)	(x+1, y)	(x+2, y)
(x-2, y+1)	(x-1, y+1)	(x, y+1)	(x+1, y+1)	(x+2, y+1)
(x-2, y+2)	(x-1, y+2)	(x, y+2)	(x+1, y+2)	(x+2, y+2)

Fig.3.1.2.2.6. Pixels considered for noise filtering if *SW* has highest membership in **NOISE** fuzzy set

CASE 1:

$$\nabla_{N_0}(x - 1, y + 1) = I(x, y) - I(x - 1, y + 1) \quad (3.1.2.2.49)$$

$$\nabla_{N_1}(x - 1, y + 1) = I(x - 1, y) - I(x - 1, y + 1) \quad (3.1.2.2.50)$$

$$\nabla_{N_2}(x - 1, y + 1) = I(x - 2, y) - I(x - 1, y + 1) \quad (3.1.2.2.51)$$

$$\nabla_{N_3}(x - 1, y + 1) = I(x - 2, y + 1) - I(x - 1, y + 1) \quad (3.1.2.2.52)$$

$$\nabla_{N_4}(x - 1, y + 1) = I(x - 2, y + 2) - I(x - 1, y + 1) \quad (3.1.2.2.53)$$

$$\nabla_{N_5}(x - 1, y + 1) = I(x - 1, y + 2) - I(x - 1, y + 1) \quad (3.1.2.2.54)$$

$$\nabla_{N_6}(x - 1, y + 1) = I(x, y + 1) - I(x - 1, y + 1) \quad (3.1.2.2.55)$$

$$\nabla_{N_7}(x - 1, y + 1) = I(x, y + 2) - I(x - 1, y + 1) \quad (3.1.2.2.56)$$

$$b_{avg} = \frac{\sum_{i=0}^7 |\nabla_{N_i}(x-1, y+1)|}{8} \quad (3.1.2.2.57)$$

When the pixel located at $(x - 1, y + 1)$ is considered as the possible noisy pixel, then it is filtered depending upon its extent of noise corruption w.r.t each of its neighboring pixels located with the 3×3 neighborhood which is defined considering $I(x - 1, y + 1)$ as the center pixel as shown in Fig.3.1.2.2.7.

(x-2, y)	(x-1, y)	(x, y)
(x-2, y+1)	(x-1, y+1)	(x, y+1)
(x-2, y+2)	(x-1, y+2)	(x, y+2)

Fig.3.1.2.2.7. 3×3 neighbourhood of the possible noisy pixel $I(x - 1, y + 1)$

After filtering $I(x - 1, y + 1)$ using the FI rules which are designed for performing noise removal (given in Table.3.1.2.2.2), the FI rules designed for performing edge & noise detection (given in Table.3.1.2.2.1) are fired for SW direction only. If the defuzzified SW value suggests that SW no longer possess highest membership value in the **NOISE** fuzzy set, then the FI rules which are designed for performing edge & noise detection are fired for all the eight possible edge directions in an 3×3 image patch which is defined considering $I(x, y)$ as the center pixel as shown in Fig.3.1.2.2.8.

Else filtered $I(x - 1, y + 1)$ is replaced with its original intensity (intensity value prior to filtering) and noise filtering operation is repeated considering $I(x, y)$ as the noisy pixel. The b_{avg} value needed for performing noise removal in this case is computed as stated below:

$(x-1, y-1)$	$(x, y-1)$	$(x+1, y-1)$
$(x-1, y)$	(x, y)	$(x+1, y)$
$(x-1, y+1)$	$(x, y+1)$	$(x+1, y+1)$

Fig.3.1.2.2.8. 3×3 neighbourhood of the possible noisy pixel $I(x, y)$

CASE 2:

$$\nabla_{N_0}^*(x, y) = I(x - 1, y + 1) - I(x, y) \quad (3.1.2.2.58)$$

$$\nabla_{N_1}^*(x, y) = I(x - 1, y) - I(x, y) \quad (3.1.2.2.59)$$

$$\nabla_{N_2}^*(x, y) = I(x - 1, y - 1) - I(x, y) \quad (3.1.2.2.60)$$

$$\nabla_{N_3}^*(x, y) = I(x, y - 1) - I(x, y) \quad (3.1.2.2.61)$$

$$\nabla_{N_4}^*(x, y) = I(x + 1, y - 1) - I(x, y) \quad (3.1.2.2.62)$$

$$\nabla_{N_5}^*(x, y) = I(x + 1, y) - I(x, y) \quad (3.1.2.2.63)$$

$$\nabla_{N_6}^*(x, y) = I(x + 1, y + 1) - I(x, y) \quad (3.1.2.2.64)$$

$$\nabla_{N_7}^*(x, y) = I(x, y + 1) - I(x, y) \quad (3.1.2.2.65)$$

$$b_{avg} = \frac{\sum_{i=0}^7 |\nabla_{N_i}^*(x, y)|}{8} \quad (3.1.2.2.66)$$

b_{avg} values needed for performing noise removal of any possible noisy pixel depends upon its respective extent of noise corruption w.r.t each of its neighboring pixels located within the 3×3 neighborhood which is defined considering the possible noisy pixel as the center pixel as discussed above. Hence, b_{avg} values obtained from (3.1.2.2.57) and (3.1.2.2.65) are different as they are calculated considering different 3×3 neighborhoods of different possible noisy pixels.

After filtering $I(x, y)$, the FI rules which are designed for performing edge & noise detection (given in Table.3.1.2.2.1) for SW direction only. If it still suggests that SW still possess highest membership value in **NOISY** fuzzy set, then both $I(x - 1, y + 1)$ and $I(x, y)$ are replaced with their corresponding filtered values. Else only $I(x, y)$ is replaced with its' filtered intensity value and edge & noise detection is performed along all the eight possible directions using FI rules given in 3.1.2.2.1.

The main aim here is to remove the noise present in the V channel of hazy image. The noise removal operation performed here is mathematically defined as follows:

$$I_{filtered}(i, j) = I_{noisy}(i, j) + a^*(i, j) \quad (3.1.2.2.67)$$

Where, $I_{noisy}(i, j)$ and $I_{filtered}(i, j)$ denote the intensities of pixel corrupted with noise located at pixel position, (i, j) and corresponding filtered pixel located at the same pixel position respectively. $a^*(i, j)$ is the defuzzified value of correction term, a' which is computed using the proposed FI rules based noise removal method to perform filtering of the noisy pixel located at pixel position, (i, j) . The

defuzzification is performed here using the Centre of Area method whose mathematical expression is given in **Chapter 1**.

After performing edge & noise detection and noise removal (if required) within a 3×3 image patch, eight dD_0 values are obtained for all eight possible edge directions within that image-patch (one for each direction). Out of all those eight possible edge directions, bacteria move along the direction having the maximum dD_0 value as stated below:

$$\theta^i(j+1, k, l) = \theta^i(j, k, l) + C(i) \frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}} \quad (3.1.2.2.68)$$

In (3.1.2.2.68), $\theta^i(j+1, k, l)$ and $\theta^i(j, k, l)$ represent the locations of the bacteria before and after tumble respectively. $C(i)$ and $\Delta(i)$ denote the step size and directional vector respectively.

3.1.2.3. Reproduction and Elimination step

Here the elimination step is performed to remove bacteria, which have deviated from their path. The reproduction step is performed to replace the same number of eliminated bacteria with healthy bacteria using a bacteria split ration of 2:1. The selection of edge pixels and elimination of unhealthy bacteria are done using certain constraints.

3.1.2.4. Constraints

If after tumbling, bacteria reach to a location which possesses highest membership value either in the **STRONG** or **VERY STRONG** fuzzy sets, then the pixel located at that position will be selected as edge pixel.

If after tumbling, bacteria reach to a location possessing highest membership value in the **NON EDGE** fuzzy set, then the bacteria are considered to be unhealthy and are eliminated.

The use of these constraints completely eliminates any chance of false edge detection. The pseudo code of the proposed BF-Fuzzy synergism based edge & noise detection method is given in Fig.3.1.2.1.

- Step 1. Initialize all bacteria at random positions.
 Step 2. For each bacterium ($i=1: S$ (total number of bacteria))
 For each chemotactic step ($j=1: N_c$ (total number of chemotactic steps))
 For each reproduction step ($k=1: N_{re}$ (total number of reproduction steps))
 For each elimination and dispersal step ($l=1: N_{ed}$ (total number of elimination and dispersal steps))
- i. Find the pixels located in the 3×3 neighbourhood of the initial position of the bacteria.
 - ii. Evaluate dD_0 values for all the possible eight directions within the 3×3 neighbourhood.
 - iii. If dD_0 value of any direction suggests that direction has the highest membership value in the **NOISE** fuzzy set, then perform noise removal using the FI rules based noise filtering method and go to step ii.
 - iv. Move the bacteria to the direction having maximum dD_0 value within the 3×3 window using directional vector $\Delta(i)$ and step size $C(i)$.
 - v. Select the pixels located in the directions having the highest membership values either in the **STRONG** or **VERY STRONG** fuzzy sets as edge pixels.
 - vi. Eliminate the bacteria, reaching to locations having the highest membership values in the **NON EDGE** fuzzy set.
 - vii. Replace the total number of eliminated bacteria with same number of healthy bacteria.

Fig.3.1.2.1. Pseudo code of the BF and Fuzzy synergism based edge & noise detection method

3.1.2.5. Comparative analyses of noise removal methods

In this Section, comprehensive comparative analysis of the filtering strategies of several existing, well-known noise removal filters is performed and how the proposed FI rules based noise removal method

can satisfactorily overcome their shortcomings is discussed. The filtering strategies of some well-known noise removal filters are described below:

1. Mean filter [94]: This filter performs noise removal of all the pixels present in an image by substituting each pixel present in an image with the average value of the intensities of all the pixels lying within its pre-defined neighborhood of size, $W \times W$.

2. Median filter (MF) [96] performs removal of noise present in images by substituting each pixel present in an image with the median value of the intensities of all the pixels lying within its pre-defined neighborhood of size, $W \times W$.

3. Hybrid Median filter (HMF) [97] performs noise suppression by filtering all the pixels belonging to an image irrespective of whether it is a noisy pixel or a noise-free pixel using two steps.

Step 1. Initially, HMF divides the neighboring pixels lying within a filter window of pre-defined size, 3×3 into two groups where the first group comprises of pixels which are located at pixel locations marked as 'R' in Fig. 3.1.2.5.1 and the second group comprises of pixels which are located at pixel locations marked as 'D' in Fig. 3.1.2.5.1.

D	R	D
R	C	R
D	R	D

Fig.3.1.2.5.1. 3×3 filter window

Step 2. In this step, HMF substitutes the intensity of 'C' (center pixel) of the filter window with the median value (M) which is computed as stated in (3.1.2.5.1.1).

$$M = \text{median}(MR, MD, C) \quad (3.1.2.5.1.1)$$

Where, MR and MD are the median values of the pixels belonging to the first group and the second group respectively.

4. Modified Hybrid Median filter (MHMF) [98] performs suppression of noise present in images almost similarly as proposed in HMF [97] but with certain modifications. Instead of computing M as stated in (3.1.2.5.1.1), the authors who have proposed MHMF have substituted C with M' which is computed using the following equation:

$$M = \text{median}(MR, MD, M') \quad (3.1.2.5.1.2)$$

Where, ' M' ' is the median value of all the pixels which are located at pixel locations marked as 'R' and 'D' in Fig.3.1.2.5.1.

Limitations of MF, HMF and MHMF: As these filters perform noise suppression by filtering all the pixels present in an image irrespective of determining whether it is a noisy pixel or a noise-free pixel, hence, they often lead to the production of undesired outputs.

5. Rank-order based Adaptive Median filter (RAMF) [99] : Unlike MF, HMF and MHMF, RAMF performs removal of noise present in images by filtering noisy pixels only. RAMF performs detection and filtering of noisy pixels in two steps:

Step 1: If the intensity of the center pixel of a filter window of size $W \times W$ lies between the minimum and maximum intensities of the pixels located within that window, RAMF considers that center pixel as a noise free pixel and retains its intensity value without any modification. Else, if the intensity of the center pixel of any filter window becomes equal to either minimum or maximum intensity value of the pixels located within that window, then RAMF marks it as a noisy pixel and substitutes its intensity value with a new intensity value which is computed as stated in Step 2.

Step 2. If the median value of the intensities of the pixels located within a pre-defined filter window of size $W \times W$ lies between the maximum and minimum values of the intensities of the pixels located within that window, then RAMF substitutes the intensity of the center pixel of that window which is identified as a noisy pixel with that median value. Else, RAMF increases the size of the filter window by one position on all the sides, i.e. the new size of the filter window becomes $(W + 1) \times (W + 1)$ and computes the median value of the new window and check whether the newly computed median value lies between the minimum and maximum intensities of the pixels located within the window of size $(W + 1) \times (W + 1)$. If yes, then RAMF substitutes the intensity of the center pixel with the new median value. Else, it further increases the size of the filter and repeats the same process until either correct median value is obtained or maximum pre-defined window size $W_m \times W_m$ is reached.

Limitation of RAMF: This filter performs noise detection by marking the center pixel of any filter window of pre-defined size as noisy pixel if its' intensity value becomes equal to either the minimum and maximum intensities of the pixels located within that filter window. This method of noise detection often gives erroneous results as pixels possessing minimum and maximum intensities locally (within a filter window) may not always be noisy. They may be a part of the (low intensity/high intensity) homogeneous regions which generally possess large clusters of pixels possessing similar intensities.

6. Progressive Switching Median filter (PSMF) [100] performs removal of noise from a noise corrupted image in an iterative manner. Initially, PSMF considers all the pixels belonging to a noisy image to be noise-free and performs noise detection using a filter window of size 3×3 . If the difference between the intensity of the center pixel of a filter window and the median of all the pixels located within that window exceeds a pre-defined (manually selected) threshold, then PSMF marks it as a noisy pixel and replaces it with the median value of the intensities of all the pixels located within that window. The same operation is repeated for all iterations (total number of iterations is chosen manually at the beginning of the filtering process).

7. Improved Progressive Switching Median filter (IPSMF) [101] performs noise suppression similarly as proposed in PSMF but instead of substituting the value of the center pixel of any filter window (which is identified as a noisy pixel) with the median value of the intensities of pixels lying within that window, IPSMF substitutes it with half of value of the sum obtained by adding the median and mean values of all the pixels lying within the filter window.

Limitations of PSMF and IPSMF: These filters perform noise detection by comparing the difference between the intensity of the center pixel of any filter window and the median value of all the pixels located within that window with a pre-defined (manually selected) threshold which enhances the chance of performing erroneous noise detection as this entire noise detection process is dependent on the threshold value. Selecting correct threshold value to perform noise detection is very challenging.

Moreover, all these filters substitute the intensities of all the pixels present in an image (in case of MF, HMF and MHMF) or the detected noisy pixels (in case of RAMF, PSMF and IPSMF) with the median value of the pixels located within a filter window calculated in some way or the other as discussed above which often lead to the production of undesirable results as calculated median value may be noisy itself and it may not always be the ideal substitute of the intensity of the noisy pixels. The proposed FI rules based noise removal method can efficiently overcome the drawbacks of all these existing, well-known noise removal filters.

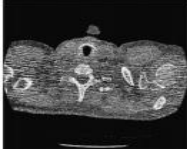

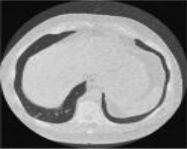
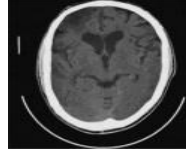
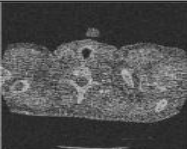
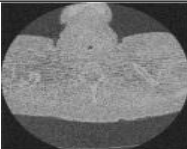

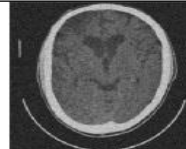
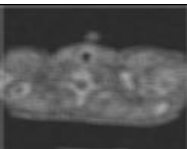
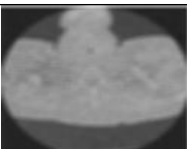
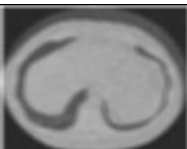
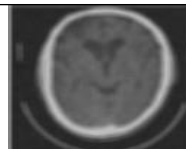
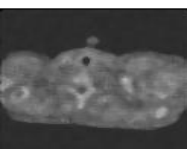

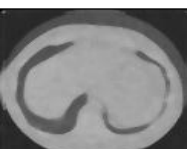
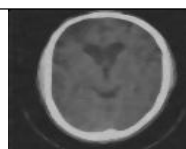
- a. The proposed noise removal method does not perform filtering of all the noisy as well as noise free pixels belonging to an image like MF, HMF and MHMF filters, instead it performs filtering of the pixels only when they are detected as noisy pixels. Moreover, as the proposed method performs simultaneous edge & noise detection using a set of novel FI rules (given in Table.3.1.2.2.1) which are designed considering the $|FD|$ values of pixels as input linguistic variables, it completely eliminates any chances of erroneous noise detection which in turn reduces the chances of producing undesired results.
- b. The noise removal method designed here detects noisy pixels taking into the account the $|FD|$ values of the center pixel of any filter window w.r.t the neighboring pixels lying within that window. So unlike RAMF filter, the designed filter completely eliminates any chances of

erroneously detecting pixels belonging to either low intensity homogeneous regions or high intensity homogeneous regions as noisy pixels. Moreover, unlike PSMF and IPSMF filters, the proposed noise removal method does not use any threshold (manually selected) to perform detection of noisy pixels, which also helps it to overcome any drawbacks which may arise in the methods proposed in [100] and [101] due to improper selection of the threshold values.

- c. Most of the existing, well-known noise removal filters substitute either all the pixels belonging to an image (in case of MF, HMF and MHMF filters) or the detected noisy pixels (in case of RAMF, PSMF and IPSMF filters) with the median value of the intensities of the neighboring pixels calculated in some way or the other as discussed above, but those median values may not always be the ideal substitute of the noisy pixels as the intensities of pixels whose values get selected as the median values may be noisy pixels themselves. In the proposed noise removal method, the detected noisy pixels are substituted with new intensity values which are computed using the unique defuzzified values of the correction term a' as stated in (3.1.2.2.67). As in the proposed method, an unique $(a')^*$ is calculated for filtering each noisy pixel depending upon its extent of noise corruption w.r.t to each of its neighboring pixel, located within its 3×3 neighborhood using the novel set of FI rules (given in Table. 3.1.2.2. 2), the computed intensity value of the filtered pixel obtained in the proposed method acts as much reliable substitute of the intensity of the detected noisy pixel compared to the filtered values used for substituting the intensities of noisy pixels in existing filters.

Comparative analysis of the proposed FI rules based noise removal method w.r.t to various well-known noise removal methods is carried out both qualitatively as well as quantitatively in Table 3.2.1.5.1 and Table 3.2.1.5.2 respectively. The original CT images which are used for performing the comparative analyses are taken from the source mentioned in [102] which is a publicly available database. The noisy CT images which are used for performing comparative analyses are of size 512×512 and are generated by introducing noise having noise level of 20 in original images.

Table 3.2.1.5.1. Comparative qualitative results obtained using CT images

Methods	Image 1	Image 2	Image 3	Image 4
Original Image				
Noisy Image				
Mean filter [94]				
Median filter [96]				

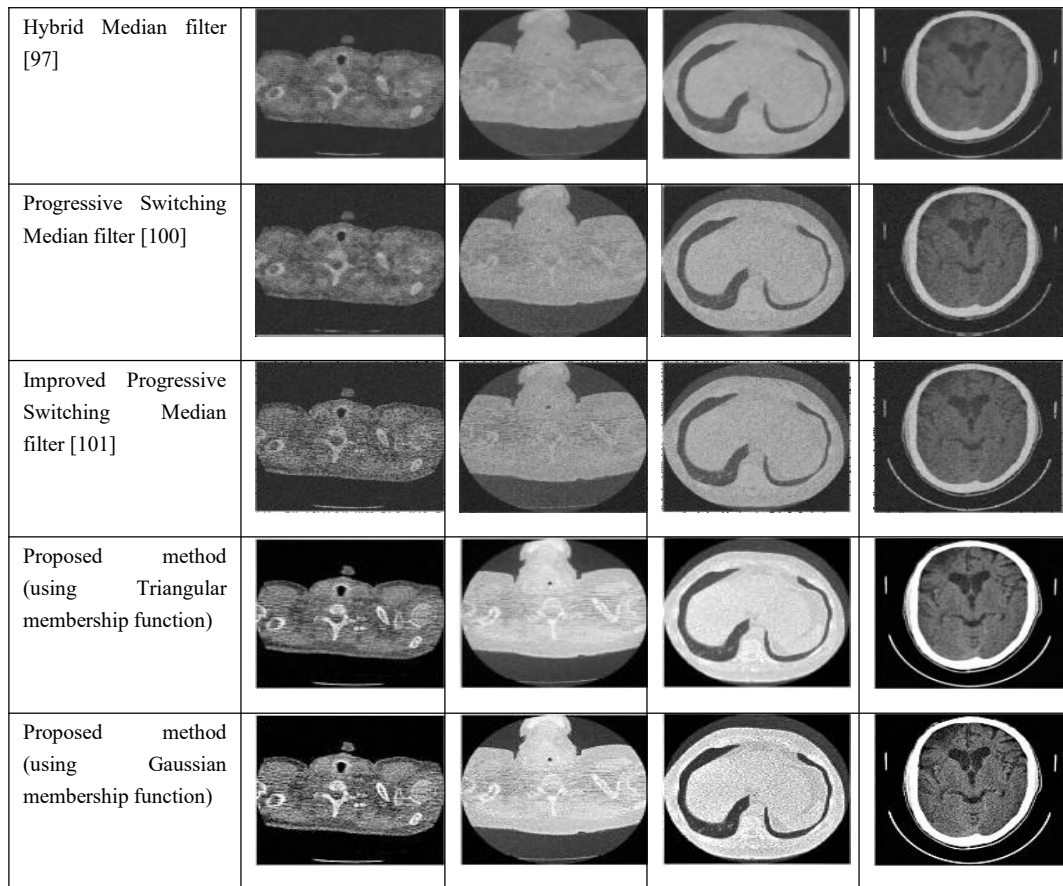


Table 3.2.1.5.2. Comparative qualitative results obtained using CT images

Methods	Image 1		Image 2		Image 3		Image 4	
	PSNR	Entropy	PSNR	Entropy	PSNR	Entropy	PSNR	Entropy
Mean filter [94]	14.626	0.6229	14.277	0.6085	15.131	0.3865	15.778	0.4460
Median filter [96]	14.924	0.4920	13.911	0.3453	14.595	0.3399	16.507	0.3458
Hybrid Median filter [97]	14.518	0.0421	14.567	0.3424	16.426	0.1337	16.587	0.2678
Progressive Switching Median filter [100]	14.367	0.2538	14.203	0.1609	15.356	0.1188	16.795	0.1143
Improved Progressive Switching Median filter [101]	14.323	0.1074	14.435	0.1467	16.367	0.0745	16.772	0.1095
Proposed method (using Triangular membership function)	15.591	0.985	16.904	0.1386	16.963	0.0669	17.06	0.906
Proposed method (using Gaussian membership function)	15.967	0.804	16.952	0.1174	17.711	0.583	18.329	0.898

The parameters used for performing quantitative analysis are Peak Signal-to-Noise Ratio (PSNR) [103] and Entropy [104]. Brief definitions of these parameters are given below:

PSNR: This parameter is used to measure the quality of restored haze-free image. Hence, high values of this parameter are always desirable as those values suggest better image restoration. Mathematically, this parameter is represented as follows:

$$PSNR = 10\log_{10}\left(\frac{Max^2}{MSE}\right) \quad (3.1.2.5.1.3)$$

The terms on the right and side of the equation for e.g. Max represents the maximum intensity of the pixel present in an image while MSE denote Mean Signal-to-Noise Ratio.

Entropy: Entropy of an image gives a measure of the amount of uncertainty present in the information content of an image.

3.1.2.6 .Comparative analysis of edge detection methods

In this Sub-section, a detailed discussion is carried out regarding some of the most common, well-known edge detectors like Roberts, Prewitt, Sobel and Canny edge detectors [94]. The 3×3 kernels which are used by the Roberts, Prewitt, Sobel edge detectors to perform edge detection are given in Table. 3.1.2.6.1.

Table.3.1.2.6.1. 3×3 kernels used for performing edge detection by Roberts, Prewitt and Sobel edge detectors

No	Edge detector	Masks used																																				
1	Roberts	<div style="display: flex; justify-content: center; align-items: center;"> <table border="1" style="margin-right: 20px;"> <tr><td>-1</td><td>0</td></tr> <tr><td>0</td><td>1</td></tr> </table> <table border="1"> <tr><td>0</td><td>-1</td></tr> <tr><td>1</td><td>0</td></tr> </table> </div> <p style="text-align: center;">2×2 kernels used for performing edge-detection along diagonal directions</p>	-1	0	0	1	0	-1	1	0																												
-1	0																																					
0	1																																					
0	-1																																					
1	0																																					
2	Prewitt	<table border="1" style="width: 100%; text-align: center;"> <tr> <td>-1</td><td>-1</td><td>-1</td> <td>-1</td><td>0</td><td>1</td> <td>0</td><td>1</td><td>1</td> <td>-1</td><td>-1</td><td>0</td> </tr> <tr> <td>0</td><td>0</td><td>0</td> <td>-1</td><td>0</td><td>1</td> <td>-1</td><td>0</td><td>1</td> <td>-1</td><td>0</td><td>1</td> </tr> <tr> <td>1</td><td>1</td><td>1</td> <td>-1</td><td>0</td><td>1</td> <td>-1</td><td>-1</td><td>0</td> <td>0</td><td>1</td><td>1</td> </tr> </table> <p style="text-align: center;">(a)Horizontal (b) Vertical (c) Diagonal (d) Diagonal</p>	-1	-1	-1	-1	0	1	0	1	1	-1	-1	0	0	0	0	-1	0	1	-1	0	1	-1	0	1	1	1	1	-1	0	1	-1	-1	0	0	1	1
-1	-1	-1	-1	0	1	0	1	1	-1	-1	0																											
0	0	0	-1	0	1	-1	0	1	-1	0	1																											
1	1	1	-1	0	1	-1	-1	0	0	1	1																											
3	Sobel	<table border="1" style="width: 100%; text-align: center;"> <tr> <td>-1</td><td>-2</td><td>-1</td> <td>-1</td><td>0</td><td>1</td> <td>0</td><td>-1</td><td>2</td> <td>-2</td><td>-1</td><td>0</td> </tr> <tr> <td>0</td><td>0</td><td>0</td> <td>-2</td><td>0</td><td>2</td> <td>-1</td><td>0</td><td>1</td> <td>-1</td><td>0</td><td>1</td> </tr> <tr> <td>1</td><td>2</td><td>1</td> <td>-1</td><td>0</td><td>1</td> <td>-2</td><td>1</td><td>0</td> <td>0</td><td>1</td><td>2</td> </tr> </table> <p style="text-align: center;">(a)Horizontal (b) Vertical (c) Diagonal (d) Diagonal</p>	-1	-2	-1	-1	0	1	0	-1	2	-2	-1	0	0	0	0	-2	0	2	-1	0	1	-1	0	1	1	2	1	-1	0	1	-2	1	0	0	1	2
-1	-2	-1	-1	0	1	0	-1	2	-2	-1	0																											
0	0	0	-2	0	2	-1	0	1	-1	0	1																											
1	2	1	-1	0	1	-2	1	0	0	1	2																											

From Table.3.1.2.6.1 , it becomes evident that the existing edge detectors (Roberts, Prewitt and Sobel) use different 3×3 kernels to perform edge detection along horizontal, vertical and diagonal edge directions. Moreover, these edge detectors also use manually selected threshold say, (T) to minimize

false edge detection. The pixels whose intensities are either equal to or greater than the threshold are included in the edge maps which are estimated by these edge detectors while the pixels having intensities less than the threshold are excluded from estimated edge maps.

These edge detectors compute magnitude and direction of possible edges existing at pixel position, (x, y) , located along the center pixel of the 3×3 image patch given in Fig.3.1.2.6.1 using the following equations:

$$M(x, y) = |g_x| + |g_y| \quad (3.1.2.6.1)$$

$$\alpha(x, y) = \tan^{-1} \left[\frac{g_x}{g_y} \right] \quad (3.1.2.6.2)$$

$(x-1, y-1)$	$(x, y-1)$	$(x+1, y-1)$
$(x-1, y)$	(x, y)	$(x+1, y)$
$(x-1, y+1)$	$(x, y+1)$	$(x+1, y+1)$

Fig.3.1.2.6.1. 3×3 image patch centered at pixel position, (x, y)

Roberts operator calculates the magnitude of edge at pixel position, (x, y) , the center pixel of the 3×3 image-patch (given in Fig. 3.1.2.6.1) using the following equation:

$$M(x, y) = |I(x + 1, y + 1) - I(x, y)| + |I(x, y + 1) - I(x + 1, y)| \quad (3.1.2.6.3)$$

In (3.1.2.6.3),

$$g_x = I(x + 1, y + 1) - I(x, y) \quad (3.1.2.6.4)$$

$$g_y = I(x, y + 1) - I(x + 1, y) \quad (3.1.2.6.5)$$

Prewitt operator calculates the magnitude of edge at pixel position, (x, y) , the center pixel of the 3×3 image-patch (given in Fig.3.1.2.6.1) using the following equation:

$$M(x, y) = \left| (I(x - 1, y + 1) + I(x, y + 1) + I(x + 1, y + 1)) - (I(x - 1, y - 1) + I(x, y - 1) + I(x + 1, y - 1)) \right| + \left| (I(x + 1, y - 1) + I(x + 1, y) + I(x + 1, y + 1)) - (I(x - 1, y - 1) + I(x - 1, y) + I(x - 1, y + 1)) \right| \quad (3.1.2.6.6)$$

In (3.1.2.6.6),

$$g_x = I(x - 1, y + 1) + I(x, y + 1) + I(x + 1, y + 1) - I(x - 1, y - 1) - I(x, y - 1) - I(x + 1, y - 1) \quad (3.1.2.6.7)$$

$$g_y = I(x + 1, y - 1) + I(x + 1, y) + I(x + 1, y + 1) - I(x - 1, y - 1) - I(x - 1, y) - I(x - 1, y + 1) \quad (3.1.2.6.8)$$

Sobel operator calculates the magnitude of edge at pixel position, (x, y) , the center pixel of the 3×3 image-patch (given in Fig.3.1.2.6.1) using the following equation:

$$M(x, y) = \left| (I(x - 1, y + 1) + I(x, y + 1) + I(x + 1, y + 1)) - (I(x - 1, y - 1) + I(x, y - 1) + I(x + 1, y - 1)) \right| + \left| (I(x + 1, y - 1) + I(x + 1, y) + I(x + 1, y + 1)) - (I(x - 1, y - 1) + I(x - 1, y) + I(x - 1, y + 1)) \right| \quad (3.1.2.6.9)$$

In (3.1.2.6.9),

$$g_x = (I(x - 1, y + 1) + I(x, y + 1) + I(x + 1, y + 1)) - (I(x - 1, y - 1) + I(x, y - 1) + I(x + 1, y - 1)) \quad (3.1.2.6.10)$$

$$g_y = (I(x + 1, y - 1) + (I(x + 1, y) + I(x + 1, y)) + I(x + 1, y + 1) - (I(x - 1, y - 1) + (I(x - 1, y) + I(x - 1, y)) + I(x - 1, y + 1) \quad (3.1.2.6.11)$$

Limitations of Robert, Prewitt and Sobel edge detectors: Roberts edge detector is the simplest among all the edge detectors whose details are discussed here. It performs edge detection using the 2×2 kernels given in Table.3.1.2.6.1 but as these kernels cannot capture the information from either side of the center pixel (variation of intensities of the center pixel w.r.t to the neighboring pixels located on its either side), hence, Roberts edge detector often performs erroneous edge map estimation.

Prewitt and Sobel edge detectors are designed to mitigate these drawbacks. As these edge detectors use 3×3 kernels (shown in Table 3.1.2.6.1.) to perform edge detection, so unlike Roberts edge detector, these edge detectors can perform edge detection by taking into account the information from both the sides of the center pixel.

The performance efficiencies of these edge detectors are entirely dependent on the manually selected threshold (T) as it decides whether a pixel will be included as an edge pixel in the estimated edge map or not. Hence, selecting proper threshold value that too manually is very necessary to perform accurate edge map estimation using these detectors. The selection of the accurate value of ' T ' is a very challenging task.

To reduce the dependency of the entire edge detection process on a single threshold value, Canny edge detector performs edge detection using the hysteresis thresholding method which performs edge detection using two thresholds namely, (T_L) (low threshold) and (T_H) (high threshold). The edge detection operation performed by the Canny edge detector is discussed below:

Canny edge detector detects edges along four directions namely, horizontal (d_1), vertical (d_2), -45° (d_3) and 45° (d_4) using similar 3×3 kernels as used by Prewitt and Sobel operators (shown in Table 3.1.2.6.1).

For e.g. For detecting the presence of edge at the center pixel of the 3×3 image-patch (given in Fig.3.1.2.6.1) using Canny edge detector, one have to perform the entire edge detection operation using the following steps:

Step 1. Initially, the magnitude and direction of edge at pixel position, (x, y) are calculated as stated in (3.1.2.6.1) and (3.1.2.6.2), where the values of g_x and g_y values are chosen according to the kernels used for performing edge detection.

Step 2. The calculated edge direction, $\alpha(x, y)$ is set to its closest d_k value. (Where, $k \in \{1,2,3,4\}$)

Step 3. The calculated $M(x, y)$ is compared with the intensities of its two neighbors located along the horizontal direction (here, $I(x, y - 1)$ and $I(x, y + 1)$) (if after Step 2, $\alpha(x, y)$ is set to d_1). If $M(x, y)$ is lesser than any of these neighbors then set $g_N(x, y) = 0$ in the corresponding edge map. Else set $g_N(x, y) = M(x, y)$.

To avoid the presence of false edges in the computed edge map, Canny edge detector uses hysteresis thresholding method where it computes two more edge maps $g_{NH}(x, y)$ and $g_{NL}(x, y)$ from $g_N(x, y)$ by using two thresholds T_H (high threshold) and T_L (low threshold) as stated below.

$$g_{NH}(x, y) = g_N(x, y) \leq T_H \quad 3.1.2.6.12$$

$$g_{NL}(x, y) = g_N(x, y) \geq T_L \quad 3.1.2.6.13$$

Pixels included in $g_{NH}(x, y)$ are termed as strong edge pixels and pixels included in $g_{NL}(x, y)$ are termed as weak edge pixels. Sometimes, due to the selection of very high T_H , some gaps occur within the strong edges. The final edge map, $g'_N(x, y)$ is formed by including all edge pixels belonging to $g_{NH}(x, y)$ and only those pixels from $g_{NL}(x, y)$ which are connected to the pixels included in $g_{NH}(x, y)$ in order to bridge the gaps between strong edges.

In Table.3.1.2.6.2 and Table.3.1.2.6.3, it is shown how the thresholds influence the qualities of the estimated edge maps. These tables are included in this work to show how the edge estimation capabilities of well-known edge detectors largely depend on thresholds.

Table. 3.1.2.6.2. Effect of threshold (T) on estimated edge maps


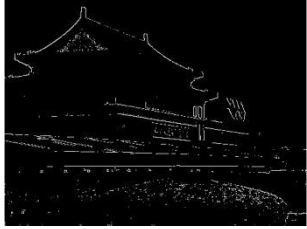
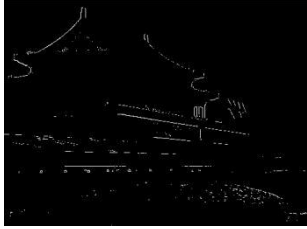
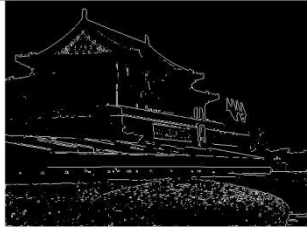
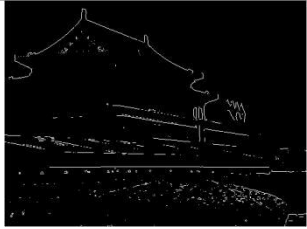
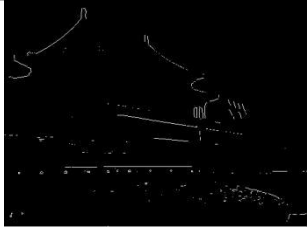
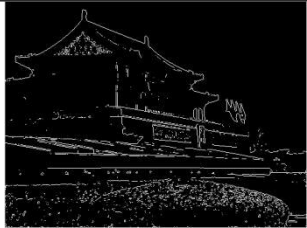
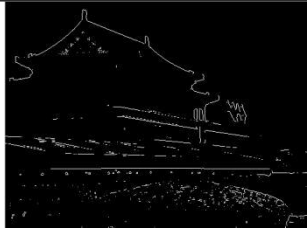
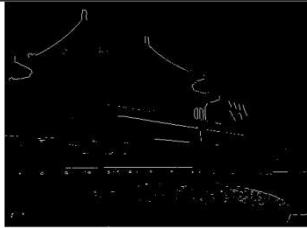
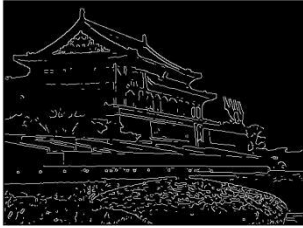
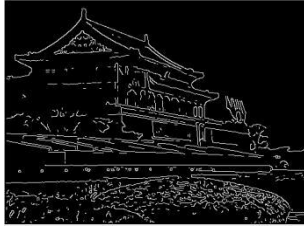

Edge detector	$T=0.04$	$T=0.06$	$T=0.08$
Roberts			
Prewitt			
Sobel			

Table.3.1.2.6.3. Effect of hysteresis thresholding on estimated edge maps

Edge detector	$T_L = 0.04$ $T_H = 0.06$	$T_L = 0.06$ $T_H = 0.08$	$T_L = 0.08$ $T_H = 0.1$
Canny			

Edge maps given in Table. 3.1.2.6.2 and Table. 3.1.2.6.3 depict the influence of thresholds on them. The most notable shortcomings of the existing, well-known edge detectors are:

- These edge detectors cannot differentiate between edge and noise pixels and hence, they often erroneously include noisy pixels into the estimated edge maps owing due to the similarities in some of the properties of noisy pixels and edge pixels.
- The use of manually selected thresholds to perform edge detection by these edge detectors often leads to the inclusion of false edges in estimated edge maps.
- The quality of estimated edge maps largely depends on the thresholds selected for performing edge detection.
- The edge detection operation performed by the well-known edge detectors are kernel-dependent as a particular kernel can detect edge only along the direction for which it is designed. Different kernels are required to detect edges along different directions.

The proposed BF-Fuzzy synergism edge & noise detection method overcome these shortcomings as discussed below:

The FI rules (given in Table. 3.1.2.2.1) which is designed to perform edge detection and edge strength evaluation can also detect noisy pixels simultaneously along with edge pixels. Moreover, apart from selecting edge pixels within any image-patch, the proposed method performs filtering of noise pixels (if any) within that patch using another set of novel FI rules (given in Table.3.1.2.2.2). Hence, possibilities of erroneously including noisy pixels as edge pixels in the edge maps estimated by the proposed edge detection method are almost negligible.

b. The use of constraints (given in Subsection 3.1.2.4) to select final edge pixels within each image-patch in the proposed edge detection method completely reduces the chances of inclusion of false edges in estimated edge maps.

c. The proposed edge detection method does not involve the use of any threshold to perform edge detection. Hence, there are no chances of incorrect edge map estimation due to improper selection of threshold values.

d. The proposed method does not use pre-defined, direction-specific kernels to perform edge detection, instead it detects edges along all eight possible edge directions within each image-patch efficiently.

Some examples of the edge maps estimated using the proposed BF- Fuzzy synergism based method is given below:

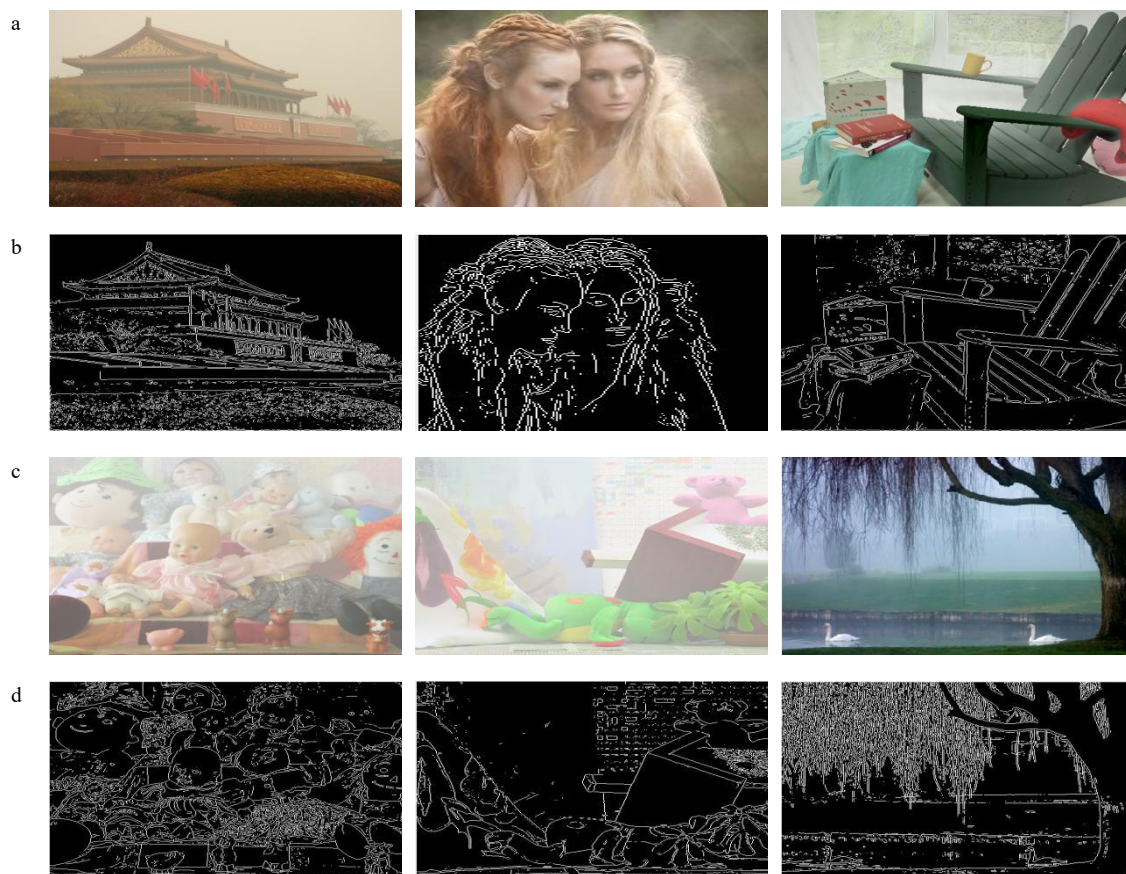


Fig.3.1.2.6.2. Some examples of edge maps estimated using the proposed method

3.1.3. FI rules based edge-sharpening

The proposed FI rules based edge-sharpening procedure overcome the limitations of existing un-sharp masking method [94].

Mathematically, the un-sharp masking method is explained using the following equations:

$$g_{mask}(i, j) = f(i, j) - \overline{f(i, j)} \quad (3.1.3.1)$$

$$g(i, j) = f(i, j) + c * g_{mask}(i, j) \quad (3.1.3.2)$$

In (3.1.3.1) and (3.1.3.2), $f(i, j)$, $\overline{f(i, j)}$, and $g(i, j)$ represent the intensities of pixels located at pixel position, (i, j) in original image, smoothed image (which is generated by replacing each pixel belonging to the original image with the average of all the pixels located within its pre-defined neighborhood) and sharpened image respectively. The term, g_{mask} in (3.1.3.1) denotes the edge map of the original image, f which is estimated by the un-sharp masking method as stated in (3.1.3.1).

The limitations of the un-sharp masking method are:

- The un-sharp masking method performs detection of edge pixels simply as stated in(3.1.3.1) . Hence, it often erroneously includes noisy pixels in estimated edge maps due to some similarities in their properties with the edge pixels. Moreover, the un-sharp masking method also performs sharpening of the noisy pixels which are erroneously included in the estimated edge maps as stated in (3.1.3.2) which boosts the presence of noise in output images.
- The output sharpened images produced by the un-sharp masking method often contain dark halo-artifacts at some pixel positions, where the intensities of pixels belonging to the sharpened images become lesser than their corresponding intensities in original images.

In this work, the first limitation of the un-sharp masking method is overcome by performing edge map estimation using the proposed BF-Fuzzy synergism based edge & detection method as it completely eliminates any chances of the presence of noise in the estimated edge maps as discussed earlier.

The second drawback i.e. dark halo-artifact arise in output sharpened images obtained using un-sharp masking method due to the following reasons:

- As shown in (3.1.3.1) in un-sharp masking method, $g_{mask}(x, y)$ is calculated by subtracting $f(x, y)$ and $\overline{f(x, y)}$. Again, $\overline{f(x, y)}$ is calculated by patch-wise averaging $f(x, y)$. So, there are chances when $\overline{f(x, y)} > f(x, y)$ at some pixel locations where values of $g_{mask}(x, y)$ will be negative.
- According to (3.1.3.2), the sharpened image $g(x, y)$ is obtained by adding $g_{mask}(x, y)$ with $f(x, y)$ that too after multiplication with the sharpening factor 'c'. High values of 'c' increase sharpening and enhance chances of the presence of halo-artifacts and vice versa.

In the proposed method, a set of novel FI rules (given in Table 3.1.3.1) is designed to perform sharpening of selected edges. These FI rules perform sharpening of each selected edge using unique defuzzified sharpening factor, 'c*' which is computed in accordance to the defuzzified edge-strength of that selected edge. As the proposed FI rules based edge-sharpening method facilitates sharpening of only selected edges that too with unique sharpening factors, the chances of occurrence of undesirable patterns or halo-artifacts in output haze free images due to over-sharpening become almost negligible.

Table 3.1.3.1. FI rules designed to perform edge-sharpening

Rule	If	Then
	dd_0	c
1.	SMALL	HIGH
2.	AVERAGE	MEDIUM
3.	BIG	LOW

For dd_0 : $DR = [0.6 \ 1]$. As a pixel is selected as an edge pixel if it is located at a position which has the highest membership value either in the **STRONG** or **VERY STRONG** fuzzy sets. So, it is evident from Fig. 3.1.2.2.1 (b), that the DR of dd_0 should lie between $[0.6 \ 1]$.

For c: $DR = [1 \ P_{max}]$. (Empirically chosen)

Mathematically edge-sharpening is done in this work using the following equation:

$$V_{new}(i, j) = V'(i, j) + c^*(i, j) * edgemap(i, j) \quad (3.1.3.1)$$

Where $V'(i, j)$ represents the intensity of the pixel located at the position, (i, j) of the enhanced V channel and $c^*(i, j)$ denotes the defuzzified sharpening factor which is used to sharpen the edge pixel located at (i, j) in the edge map of V' channel which is estimated using novel BF-Fuzzy synergism based edge & noise detection method. Graphical representations of the fuzzy membership functions used for representing the linguistic variables in the FI rules are given in Fig.3.1.3.1.

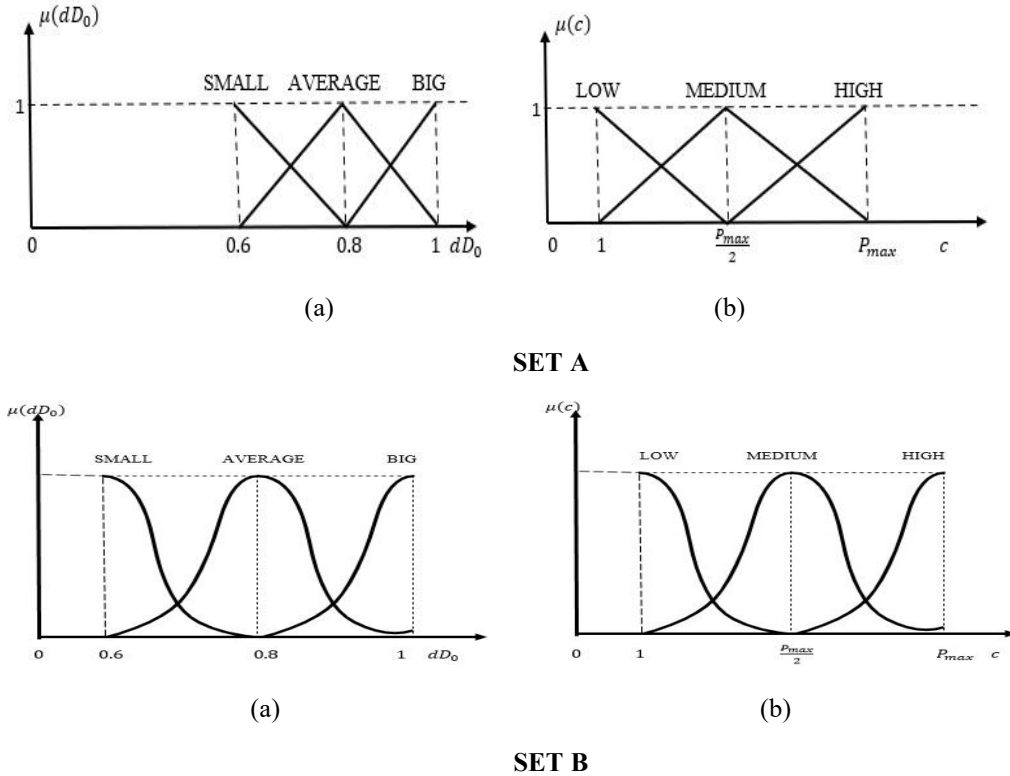


Fig. 3.1.3.1. Graphical representations of fuzzy membership functions used for representing linguistic variables of FI rules designed for edge-sharpening (a) dD_0 , (b) c

3.1.4. Block diagram

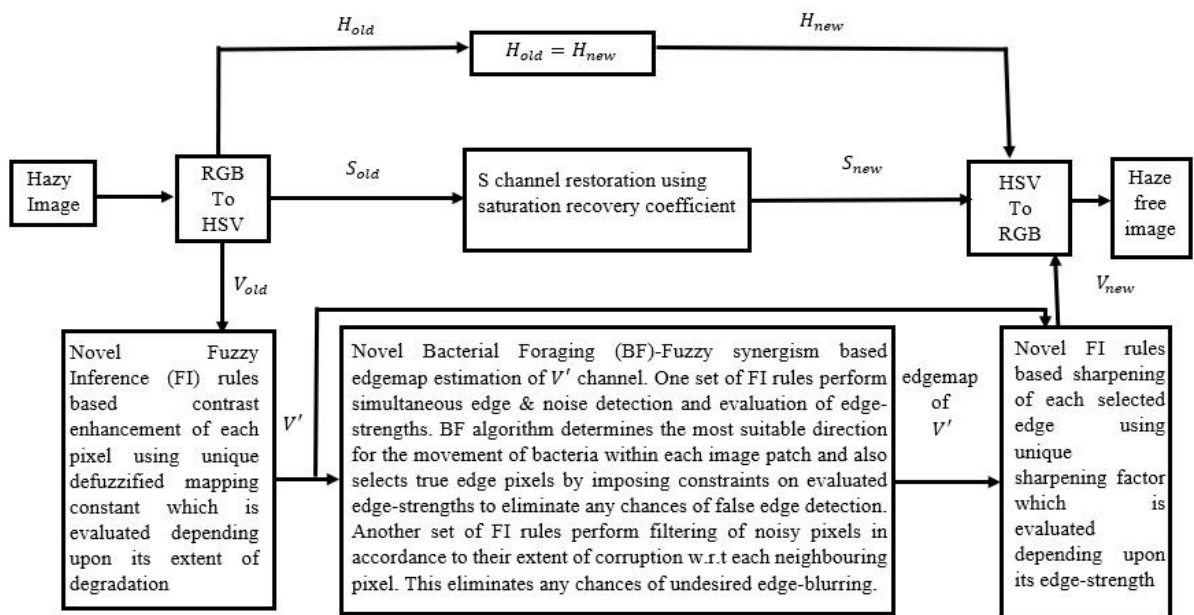


Fig.3.1.4.1. Block diagram of the proposed method

Chapter 4

This chapter presents comprehensive comparative qualitative and quantitative analyses' of results which are obtained by applying the proposed BF -Fuzzy synergism based daytime image dehazing method as well as several well-known daytime image dehazing methods. These comparative analyses are carried out using various popular benchmark databases.

4.1. Databases

This Section contains brief descriptions of the databases which are chosen to perform comparative analyses.

4.1.1. Waterloo IVC Dehazed Image Database [105]:

This is the first database which is designed for performing subjective (qualitative) and objective (quantitative) analyses of real world hazy images in 2015 by a group of researchers working in the University of Waterloo. This database contains 25 hazy images, out of which 22 are hazy images of real world outdoor scenes while the other 3 are images of real world indoor scenes in which haze is stimulated homogeneously. For each hazy image, the authors who have designed this database have provided eight haze free images which they have obtained by applying eight different existing dehazing algorithms on hazy images. The subjective results which are obtained by applying the proposed method and several other existing dehazing methods on real world images belonging to this database are given in Table 4.3.1.

4.1.2. Middlebury Database [106]:

This database comprises of stereo images required for performing comparative analyses of various computer vision algorithms. This database contains GT image corresponding to each hazy image.

4.1.3. D-HAZY Database [107]:

This database comprises of 1400+ pairs of hazy and Ground Truth images of the same scene. The authors who have designed this database have initially acquired the images of various scenes and depth maps from Middlebury database [106] and NYU Depth database [108] and then synthetically generated the hazy images of those scenes by exploiting the properties of atmospheric scattering model and utilizing the depth information. This database contains GT image corresponding to each hazy image.

4.1.4. I-HAZE Database [109]:

This database contains 35 pairs of hazy and haze free images of real world indoor scenes. Unlike the hazy images present in Middlebury and D-HAZY databases, the hazy images present in this database are not synthetically generated, instead they are captured in the presence of haze created by artificial haze machines. This database contains GT image corresponding to each hazy image.

4.1.5. O-HAZE Database [110]:

This database contains 45 pairs of hazy and haze free images of real world outdoor scenes. Like the hazy images present in I-HAZE Database, the hazy images present in this database are also captured in the presence of haze created by artificial haze machines. This database contains GT image corresponding to each hazy image.

Images from all these databases are used for performing comparative analyses here for two reasons:

- a. These databases contain hazy images possessing varied characteristics (both real world hazy images as well as synthetically generated hazy images). Hence, performing comparative analyses using these images also helps one to prove that the proposed method can provide satisfactory results for all types of images irrespective of their characteristics.
- b. The availability of Ground Truth images corresponding to hazy images in most of the databases also provide an opportunity to compare the original scene restoration capabilities of the dehazing methods by comparing the scene characteristics of dehazed outputs with

corresponding GT images both qualitatively as well as by performing quantitative analyses using both full-reference and no-reference parameters.

The comparative qualitative and quantitative analyses results which are obtained using images from Middlebury, D-HAZY, I-HAZE and O-HAZE databases in Table 4.3.2-Table 4.3.9.

4.2. Quantitative parameters

This section contains brief descriptions of the quantitative parameters which are chosen to perform comparative quantitative analyses.

4.2.1. Full-reference quantitative parameters

- **Structural Similarity Index (SSIM) [111]:** This parameter compares the local patterns of pixels' intensities that have been normalized for both luminance and contrast. Its value lies in the range of [0 1]. Higher SSIM value indicates better similarities between two images. In this work, SSIM values are calculated by comparing output images with corresponding GT images.
- **CIEDE2000 [112]:** This parameter measures the color difference between dehazed outputs and corresponding GT images. Its value lies in the range of [0 100]. Lower CIEDE 2000 value indicates lesser color difference between two images.
- **DEHAZEfr [113]:** This quantitative parameter is designed to measure the similarities between the GT images and dehazed outputs based on three components namely, image structure recovering, over-enhancement of low-contrast areas and color rendition. Higher outcomes of this parameter suggest better results.

4.2.2. No-reference quantitative parameters

- **Haze Improvement (HI) [114]:** This quantitative parameter is designed to evaluate the performance of dehazing methods by comparing Contrast-to-Noise ratio (CNR) values of hazy and dehazed outputs of the same scene. This metric is mathematically defined as follows:

$$HI = CNR_{dehazed\ output} - CNR_{hazy} \quad (4.2.2.2.1)$$

'CNR' is mathematically represented as:

$$CNR = \frac{\mu_A - \mu_B}{\sigma_B} \quad (4.2.2.2.2)$$

In (4.2.2.2.2), the terms μ_A , μ_B signify mean attenuation coefficients of any target object lying within the region of interest and image background lying outside the region of interest respectively, whereas σ_B represents the background noise which is calculated as the standard deviation of intensities of pixels lying outside the region of interest. Higher HI value of this metric indicates better results.

















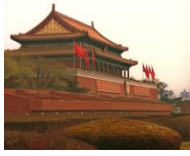















- **Fog Aware Density Estimator (FADE) [115]:** This parameter is designed based on the natural scene characteristics and fog aware statistical features. It measures the fog density of an image. Hence, low outcome of this parameter is desirable as it suggests less haze content.
- **BRISQUE [116]:** This parameter measures the amount of information loss in images. Hence, low outcome of this parameter is desirable as it suggests less information loss.

In this thesis, the efficiency and robustness of the proposed dehazing method is proved by performing comparative qualitative and quantitative analyses of output haze free images obtained from the proposed method as well as several existing state-of-the-art methods like [38],[43]-[47], [51]-[53] and [117]-[118].

4.3. Experimental results

The comparative qualitative and quantitative results obtained by applying the proposed BF-Fuzzy Logic synergism based daytime image dehazing methods as well as the well-known image dehazing methods on images belonging to the benchmark databases are given in this Section.

Table 4.3.1. Comparative qualitative analyses results obtained using Waterloo IVC Dehazed Image database

Methods	Image 1	Image 2	Image 3	Image 4
Hazy image (#)				
He <i>et al.</i> [38] (#)				
Salazar-Colores <i>et al.</i> [47](ω)				
Tarel <i>et al.</i> [43] (#)				
Meng <i>et al.</i> [44] (#)				
Xiao <i>et al.</i> [45] (#)				
Kim <i>et al.</i> [46] (#)				
Tang <i>et al.</i> [51] (#)				





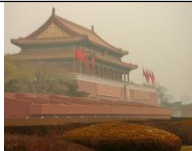




























Cai <i>et al.</i> [52] (⊖)				
Ren <i>et al.</i> [53](⊖)				
Galdran <i>et al.</i> [117](⊖)				
Ngo <i>et al.</i> [118] (⊖)				
Proposed method (using Triangular membership function)				
Proposed method (using Gaussian membership function)				

Table 4.3.2. Comparative qualitative analyses results obtained using Middlebury Image database









































Methods	Image 5	Image 6	Image 7	Image 8
Hazy images(η)				
He <i>et al.</i> [38](⊖)				
Salazar-Colores <i>et al.</i> [47] (⊖)				

Tarel et al. [43] (ω)				
Meng et al. [44](ω)				
Zhu et al. [50](ω)				
Cai et al. [52] (ω)				
Ren et al. [53](ω)				
Galdran et al. [117](ω)				
Ngo et al. [118] (ω)				
Ground Truth (η)				
Proposed method (using Triangular membership function)				
Proposed method (using Gaussian membership function)				

Table 4.3.3. Comparative quantitative analyses results obtained using Middlebury Image database

Methods	Image5			Image 6			Image7			Image8		
	SSIM	CIEDE 2000	HI	SSIM	CIEDE 2000	HI	SSIM	CIEDE 2000	HI	SSIM	CIEDE 2000	HI
He <i>et al.</i> [38]	0.858	10.78	20.98	0.878	7.43	20.38	0.816	13.71	45.69	0.882	15.30	181.10
Salazar-Colores <i>et al.</i> [47]	0.574	14.61	29.93	0.643	7.04	36.37	0.597	10.68	30.79	0.596	9.46	48.47
Tarel <i>et al.</i> [43]	0.851	15.20	25.70	0.792	18.37	38.55	0.760	18.59	74.23	0.876	8.843	226.58
Meng <i>et al.</i> [44]	0.882	11.12	11.70	0.822	12.07	29.63	0.805	14.15	70.34	0.766	23.26	212.68
Zhu <i>et al.</i> [50]	0.762	19.35	1.01	0.874	11.94	13.93	0.884	9.36	69.37	0.805	14.64	173.01
Cai <i>et al.</i> [52]	0.805	17.62	19.45	0.916	9.17	12.65	0.917	7.25	58.42	0.841	20.75	160.77
Ren <i>et al.</i> [53]	0.837	13.48	9.76	0.893	8.65	18.05	0.936	6.03	51.22	0.870	10.27	203.57
Galdran <i>et al.</i> [117]	0.841	12.30	14.09	0.887	13.96	22.89	0.921	7.53	65.41	0.893	11.78	189.07
Ngo <i>et al.</i> [118]	0.852	11.56	17.78	0.905	8.93	28.95	0.896	8.07	69.08	0.906	14.09	212.78
Proposed method (using Triangular membership function)	0.936	7.98	30.89	0.941	5.78	45.90	0.958	4.06	83.67	0.952	6.54	251.89
Proposed method (using Gaussian membership function)	0.940	6.89	32.74	0.967	5.09	42.67	0.965	3.89	90.41	0.963	4.89	266.90
Methods	Image5			Image 6			Image7			Image8		
	FAD E	BRISQ UE	DEHA ZEfr	FAD E	BRISQ UE	DEHA ZEfr	FADE	BRISQ UE	DEHA ZEfr	FADE	BRISQ UE	DEHA ZEfr
He <i>et al.</i> [38]	0.458	25.43	0.807	0.661	7.92	0.896	0.326	26.21	0.658	1.589	29.88	0.539
Salazar-Colores <i>et al.</i> [47]	0.338	17.41	0.774	0.618	23.55	0.847	0.367	40.32	0.511	0.489	24.70	0.451
Tarel <i>et al.</i> [43]	0.789	25.27	0.594	0.978	22.20	0.700	0.418	31.95	0.599	1.185	21.48	0.398
Meng <i>et al.</i> [44]	0.559	10.63	0.832	0.460	12.25	0.776	0.461	35.06	0.635	0.892	25.68	0.382
Zhu <i>et al.</i> [50]	0.647	21.56	0.582	0.892	22.87	0.588	0.547	34.65	0.432	1.648	25.33	0.398
Cai <i>et al.</i> [52]	0.537	10.70	0.613	0.869	26.63	0.627	0.541	32.58	0.490	1.547	24.25	0.405
Ren <i>et al.</i> [53]	0.572	9.38	0.636	0.620	21.56	0.746	0.379	36.39	0.476	1.279	22.02	0.446
Galdran <i>et al.</i> [117]	0.584	17.85	0.752	0.905	26.84	0.778	1.566	33.67	0.573	1.626	24.76	0.438
Ngo <i>et al.</i> [118]	0.566	15.59	0.753	0.673	28.66	0.712	1.357	38.99	0.585	1.239	26.99	0.440
Proposed method (using Triangular membership function)	0.269	9.05	0.856	0.359	7.78	0.906	0.288	19.88	0.677	0.668	18.89	0.563
Proposed method (using Gaussian membership function)	0.221	7.69	0.881	0.313	6.56	0.912	0.237	16.25	0.702	0.581	15.67	0.587

Table 4.3.4. Comparative qualitative analyses results obtained using D-HAZY database

Methods	Image 9	Image 10	Image 11	Image 12
Hazy images(η)				
He et al. [38](ω)				
Salazar-Colores et al.[47] (ω)				
Tarel et al. [43] (ω)				
Meng et al. [44](ω)				
Zhu et al. [50](ω)				
Cai et al. [52] (ω)				
Ren et al. [53](ω)				
Galdran et al. [117] (ω)				
Ngo et al. [118] (ω)				

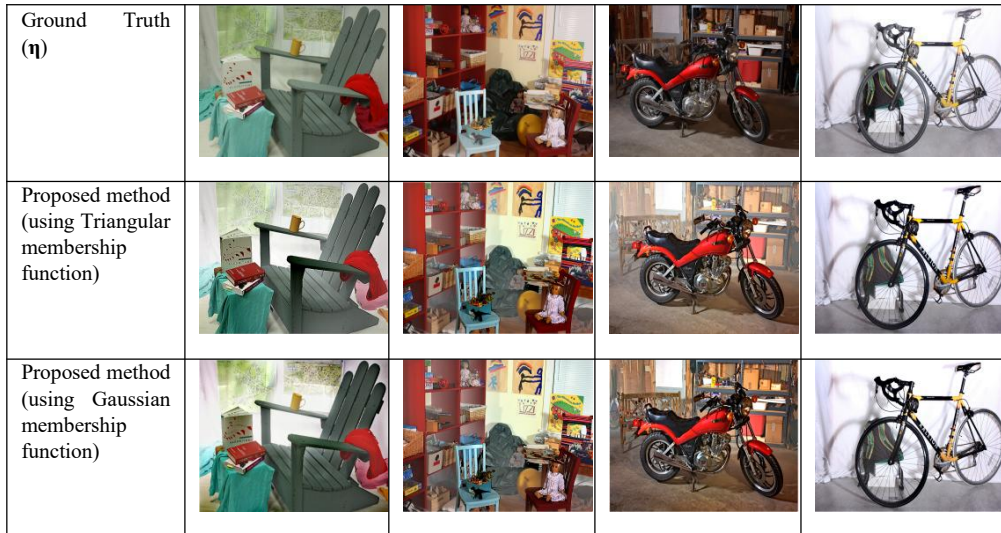


Table 4.3.5. Comparative quantitative analyses results obtained using D-HAZY database

Methods	Image9			Image 10			Image11			Image12		
	SSIM	CIEDE 2000	HI	SSIM	CIEDE 2000	HI	SSIM	CIEDE 2000	HI	SSIM	CIEDE 2000	HI
He <i>et al.</i> [38]	0.858	10.78	20.98	0.878	7.43	20.38	0.816	13.71	45.69	0.882	15.30	181.10
Salazar-Colores <i>et al.</i> [47]	0.574	14.61	29.93	0.643	7.04	36.37	0.597	10.68	30.79	0.596	9.46	48.47
Tarel <i>et al.</i> [43]	0.851	15.20	25.70	0.792	18.37	38.55	0.760	18.59	74.23	0.876	8.843	226.58
Meng <i>et al.</i> [44]	0.882	11.12	11.70	0.822	12.07	29.63	0.805	14.15	70.34	0.766	23.26	212.68
Zhu <i>et al.</i> [50]	0.762	19.35	1.01	0.874	11.94	13.93	0.884	9.36	69.37	0.805	14.64	173.01
Cai <i>et al.</i> [52]	0.805	17.62	19.45	0.916	9.17	12.65	0.917	7.25	58.42	0.841	20.75	160.77
Ren <i>et al.</i> [53]	0.837	13.48	9.76	0.893	8.65	18.05	0.936	6.03	51.22	0.870	10.27	203.57
Galdran <i>et al.</i> [117]	0.864	16.89	15.77	0.867	10.95	26.89	0.944	8.67	65.88	9.21	14.75	187.54
Ngo <i>et al.</i> [118]	0.827	19.53	18.53	0.901	11.62	32.55	0.925	7.88	75.91	9.17	12.99	199.62
Proposed method (using Triangular membership function)	0.916	8.74	32.08	0.932	7.19	42.69	0.952	5.09	87.04	0.947	7.03	243.86
Proposed method (using Gaussian membership function)	0.928	7.82	37.99	0.945	6.32	47.66	0.977	3.81	98.56	0.963	6.01	267.01
Methods	Image9			Image 10			Image11			Image12		
	FADE	BRISQ UE	DEHA ZEfr	FADE	BRISQ UE	DEHA ZEfr	FADE	BRISQ UE	DEHA ZEfr	FADE	BRISQ UE	DEHA ZEfr
He <i>et al.</i> [38]	0.468	28.18	0.737	0.305	14.60	0.716	0.213	22.18	0.752	0.497	27.29	0.596
Salazar-Colores <i>et al.</i> [47]	0.329	24.22	0.732	0.298	28.88	0.680	0.264	32.44	0.685	0.414	16.34	0.921
Tarel <i>et al.</i> [43]	0.973	11.87	0.659	0.395	23.53	0.614	0.312	38.78	0.646	0.748	24.89	0.771
Meng <i>et al.</i> [44]	0.589	25.06	0.694	1.096	22.45	0.339	0.338	17.37	0.606	0.923	31.06	0.656
Zhu <i>et al.</i> [50]	1.043	15.02	0.720	0.706	22.34	0.689	0.406	26.37	0.706	0.921	18.07	0.923
Cai <i>et al.</i> [52]	0.870	16.32	0.680	0.551	26.23	0.648	0.279	25.87	0.689	0.914	17.85	0.954
Ren <i>et al.</i> [53]	1.211	8.87	0.686	0.705	24.46	0.602	0.500	17.93	0.673	0.969	16.39	0.926
Galdran <i>et al.</i> [117]	1.084	17.88	0.780	0.562	28.49	0.701	0.365	34.65	0.414	0.870	26.92	0.192
Ngo <i>et al.</i> [118]	0.850	21.63	0.792	0.507	22.63	0.735	0.273	29.61	0.395	0.711	23.66	0.167
Proposed method (using Triangular membership function)	0.284	7.72	0.805	0.276	12.78	0.756	0.185	16.38	0.778	0.398	14.62	1.09

Proposed method (using Gaussian membership function)	0.259	6.98	0.843	0.255	10.11	0.778	0.177	15.99	0.815	0.375	11.89	1.26
--	-------	------	-------	-------	-------	-------	-------	-------	-------	-------	-------	------

Table 4.3.6. Comparative qualitative analyses results obtained using I-HAZE database


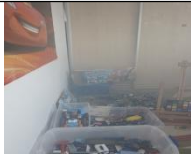
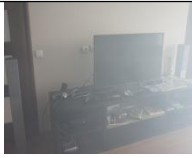
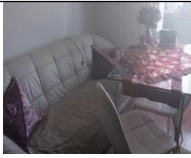





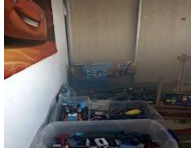


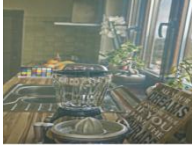













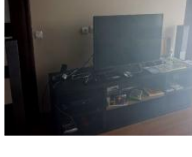

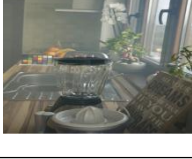
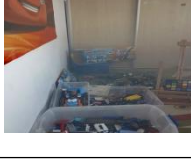
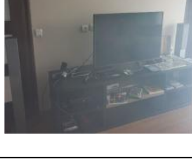
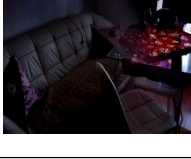
Methods	Image13	Image 14	Image 15	Image 16
Hazy images(η)				
He et al. [38](ω)				
Salazar-Colores et al.[47] (ω)				
Tarel et al. [43] (ω)				
Meng et al. [44](ω)				
Zhu et al. [50](ω)				
Cai et al. [52] (ω)				
Ren et al. [53](ω)				



























Table 4.3.7. Comparative quantitative analyses results obtained using I-HAZE database

Methods	Image13			Image 14			Image15			Image16		
	SSIM	CIEDE 2000	HI	SSIM	CIEDE 2000	HI	SSIM	CIEDE 2000	HI	SSIM	CIEDE 2000	HI
He et al.[38]	0.778	18.08	-9.25	0.617	15.66	5.75	0.661	18.04	-15.67	0.332	21.63	-7.81
Salazar-Colores et al.[47]	0.552	14.68	12.63	0.696	13.17	-4.59	0.132	5.83	-37.09	0.714	15.01	-28.55
Tarel et al.[43]	0.683	23.91	7.91	0.559	17.89	8.52	0.419	9.08	4.59	0.495	16.99	17.24
Meng et al.[44]	0.819	17.25	38.60	0.790	29.57	10.65	0.704	23.91	11.74	0.402	27.94	4.90
Zhu et al.[50]	0.874	11.49	27.02	0.794	13.89	12.72	0.772	6.38	12.88	0.594	16.45	16.08
Cai et al.[52]	0.631	20.86	32.08	0.608	15.04	15.48	0.817	5.11	9.81	0.662	12.85	29.23
Ren et al.[53]	0.899	10.74	40.47	0.806	16.01	10.56	0.866	5.06	10.01	0.748	14.06	14.29
Galdran et al. [117]	0.872	14.77	37.92	0.758	14.96	11.77	0.883	6.71	14.85	0.791	12.64	33.67
Ngo et al. [118]	0.818	12.89	34.89	0.714	15.05	14.58	0.801	8.47	16.38	0.847	11.26	28.58
Proposed method (using Triangular membership function)	0.942	8.07	48.91	0.961	12.39	16.01	0.907	4.75	17.62	0.975	10.87	35.32
Proposed method (using Gaussian membership function)	0.965	6.89	52.57	0.975	9.91	18.38	0.921	3.94	22.85	0.984	7.14	42.88
Methods	Image13			Image 14			Image15			Image16		
	FADE	BRISQU E	DEHA Zefr	FADE	BRISQ UE	DEHA Zefr	FADE	BRISQ UE	DEHA Zefr	FADE	BRISQU E	DEHA Zefr

He <i>et al.</i> [38]	0.360	23.35	0.485	0.542	20.64	0.561	0.664	31.89	0.475	0.519	31.43	0.761
Salazar-Colores <i>et al.</i> [47]	0.418	17.48	0.640	0.448	26.45	0.768	0.578	25.98	0.513	0.505	24.75	0.788
Tarel <i>et al.</i> [43]	0.421	31.05	0.634	0.691	21.90	0.628	0.643	23.69	0.457	0.487	31.05	0.786
Meng <i>et al.</i> [44]	0.453	29.33	0.438	0.685	23.88	0.549	0.781	29.08	0.484	0.595	16.89	0.797
Zhu <i>et al.</i> [50]	0.557	12.50	0.619	1.113	23.47	0.738	1.454	25.82	0.496	1.225	24.42	0.696
Cai <i>et al.</i> [52]	0.325	25.04	0.652	0.693	23.03	0.781	0.766	19.73	0.430	0.773	6.31	0.565
Ren <i>et al.</i> [53]	0.917	30.40	0.575	1.847	19.03	0.744	1.779	23.59	0.467	1.117	17.62	0.593
Galdran <i>et al.</i> [117]	0.628	28.61	0.753	1.017	24.87	0.840	2.266	27.04	0.491	0.737	26.93	0.276
Ngo <i>et al.</i> [118]	0.629	22.85	0.640	0.938	29.73	0.780	2.012	24.85	0.483	0.779	29.11	0.368
Proposed method (using Triangular membership function)	0.327	10.01	0.775	0.406	18.06	0.942	0.518	17.92	0.761	0.472	4.82	0.805
Proposed method (using Gaussian membership function)	0.273	8.69	0.857	0.389	16.82	1.054	0.447	15.39	0.984	0.401	3.09	0.925

Table 4.3.8. Comparative qualitative analyses results obtained using O-HAZE database

Methods	Image17	Image 18	Image 19	Image 20
Hazy images(η)				
He et al. [38](ω)				
Salazar-Colores et al.[47] (ω)				
Tarel et al. [43] (ω)				
Meng et al. [44](ω)				
Zhu et al. [50](ω)				























Cai et al. [52] (ω)				
Ren et al. [53](ω)				
Galdran et al. [117](ω)				
Ngo et al. [118](ω)				
Ground Truth (η)				
Proposed method (using Triangular membership function)				
Proposed method (using Gaussian membership function)				

Table 4.3.9. Comparative quantitative analyses results obtained using O-HAZE database

Methods	Image17			Image 18			Image19			Image20		
	SSIM	CIEDE 2000	HI	SSIM	CIED E 2000	HI	SSIM	CIEDE 2000	HI	SSIM	CIEDE 2000	HI
He <i>et al.</i> [38]	0.395	33.71	-18.6	0.523	15.79	6.90	0.592	32.78	3.84	0.776	19.54	-27.84
Salazar-Colores <i>et al.</i> [47]	0.631	26.58	7.857	0.763	13.96	15.18	0.817	20.34	14.86	0.773	14.17	8.73
Tarel <i>et al.</i> [43]	0.519	28.67	-0.62	0.586	11.67	-4.89	0.685	29.61	10.25	0.628	35.75	-35.15
Meng <i>et al.</i> [44]	0.360	26.95	-8.63	0.639	14.05	17.36	0.487	42.84	-10.59	0.723	21.45	-44.73
Zhu <i>et al.</i> [50]	0.622	21.38	12.37	0.684	10.96	20.61	0.735	27.69	19.54	0.831	14.29	4.67
Cai <i>et al.</i> [52]	0.708	19.86	19.44	0.721	10.01	23.29	0.771	26.72	22.85	0.844	12.78	8.17
Ren <i>et al.</i> [53]	0.652	19.32	15.66	0.745	9.58	21.50	0.824	22.19	24.03	0.881	12.34	9.45
Galdran <i>et al.</i> [117]	0.678	22.67	17.29	0.731	10.68	22.17	0.858	24.75	26.92	0.906	13.84	8.61
Ngo <i>et al.</i> [118]	0.692	28.40	19.82	0.759	11.20	27.93	0.831	27.18	28.66	0.893	11.59	12.64
Proposed method (using Triangular membership function)	0.747	17.09	24.79	0.826	8.07	25.81	0.877	16.29	35.18	0.924	9.57	14.88
Proposed method (using Gaussian membership function)	0.755	16.63	27.50	0.837	7.75	29.55	0.942	14.43	38.88	0.937	8.18	15.79
Methods	Image17			Image 18			Image19			Image20		
	FADE	BRISQ UE	DEHA ZEfr	FADE	BRIS QUE	DEHA ZEfr	FADE	BRISQ UE	DEHA ZEfr	FADE	BRISQ UE	DEHA ZEfr
He <i>et al.</i> [38]	0.214	13.86	0.443	0.459	23.68	0.356	0.345	6.92	0.508	0.152	21.25	0.494
Salazar-Colores <i>et al.</i> [47]	0.198	14.89	0.600	0.388	31.52	0.836	0.378	9.25	0.717	0.275	27.05	0.579
Tarel <i>et al.</i> [43]	0.221	23.25	0.640	0.409	43.83	0.807	0.227	43.12	0.616	0.323	42.90	0.622
Meng <i>et al.</i> [44]	0.147	17.35	0.670	0.405	40.34	0.746	0.349	14.53	0.667	0.457	14.84	0.556
Zhu <i>et al.</i> [50]	0.414	5.13	0.554	0.867	24.24	0.814	0.550	27.23	0.445	0.630	18.37	0.532
Cai <i>et al.</i> [52]	0.413	16.42	0.515	1.152	24.29	0.809	0.709	20.20	0.430	0.712	19.73	0.518
Ren <i>et al.</i> [53]	0.375	9.49	0.611	0.967	24.13	0.692	0.565	18.95	0.552	0.683	17.80	0.600
Galdran <i>et al.</i> [117]	0.365	16.82	0.629	0.240	36.74	0.863	0.478	17.84	0.505	0.594	32.88	0.505
Ngo <i>et al.</i> [118]	0.311	21.84	0.534	0.160	33.18	0.871	0.502	12.71	0.504	0.397	28.18	0.504
Proposed method (using Triangular membership function)	0.127	4.46	0.705	0.221	20.66	0.906	0.218	5.59	0.809	0.116	12.87	0.678
Proposed method (using Gaussian membership function)	0.108	3.81	0.778	0.187	15.81	1.112	0.176	4.38	0.885	0.102	8.93	0.701

Table 4.3.10. Meaning of symbols used in Table 4.3.1-Table 4.3.9

Symbol	Meaning
#	Images are taken from Waterloo IVC Dehazed Image Database
ω	Images are obtained by running the codes of different image dehazing methods. The codes of these methods are available in the websites of the respective authors who have proposed them.
η	Images are taken from D-HAZY Database
α	Images are taken from I-HAZE Database
β	Images are taken from O-HAZE Database

**The codes of He et al. [38], Zhu et al. [50], Tarel et al. [43], Meng et al.[44], Cai et al. [52], Ren et al. [53], Salazar-Colores et al. [47], Galdran et al. [117] and Ngo et al. [118] are downloaded from the sources mentioned in [119]-127] respectively.*

The effectiveness of the proposed BF-Fuzzy synergism based dehazing method in restoring the visibility of images captured during hazy weather conditions using information derived from single hazy image by performing contrast enhancement, edge& noise detection, noise removal and edge-sharpening is demonstrated in this Chapter. The use of different sets of novel FI rules to process each pixel at each stage of the proposed method depending upon that pixel's properties has also enabled it to overcome the significant shortcomings of existing dehazing methods as well as the limitations of several existing well-known edge detectors, noise removal methods and un-sharp masking method and thus reduces chances of producing noisy, over-saturated, over enhanced, over sharpened output images containing halo artifacts. The proposed novel BF-Fuzzy synergism based edge& noise detection method and the set of constraints used for performing edge map estimation facilitates the selection of true edges and almost eliminates chances of false edge detection. Moreover, comparative qualitative and quantitative analyses' results obtained by applying various dehazing methods on images which are acquired from several publicly available databases also proved the excellence of the proposed method over existing works.

Refined Color Channel Transfer Prior (which is the modified version of the existing Color Channel Transfer Prior) is proposed in this thesis. This prior is introduced as a pre-processing step of daytime image dehazing methods to enable those methods to perform dehazing of nighttime hazy images effectively alongside daytime hazy images. Detailed description of the Refined Color Channel Transfer Prior is given in this part of the thesis. Comparative qualitative and quantitative analyses of outputs obtained before and after the introduction of the prior as a pre-processing step of the existing methods are also provided in this part of the thesis to prove the efficiency of the proposed prior.

Chapter 5

Refined Color Channel Transfer Prior (RCCT) is a modified version of the existing Color Channel Transfer Prior (CCT) whose details are given in **Section. 2.2.4.5.1.1** (Image color transfer theory based methods) of **Chapter 2** of this thesis.

CCT is designed based on the observation that in every hazy image there is one color channel whose extent of attenuation and degradation is much higher compared to the other two channels present in an image. Pictorial representation of the three channels of a nighttime hazy image are given in Fig.5.1 to validate this observation.

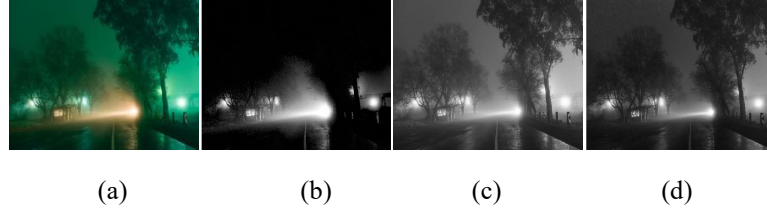


Fig.5.1.Attenuation of color channels (a) Hazy image, (b), (c) and (d): R , G and B channels of (a) respectively

Based on their observation, the authors in [73] have designed CCT prior in opponent color space, aiming to restore the information loss occurring in severely degraded color channel of images by transferring information from other comparatively less degraded color channels.

The authors have chosen opponent color space for performing the entire color conversion process because of its advantages over RGB color space, opponent color space does not requires estimation of the chromatic loss direction to perform chromatic loss compensation as the blue-yellow and red-green components of this color space are mixed where as these components in RGB color space are independent in nature.

CCT prior is mainly designed as a pre-processing step of daytime image dehazing methods in order to enable those methods to dehaze nighttime hazy images efficiently alongside nighttime hazy images.

As discussed in **Chapter 2**, CCT performs color transfer between source image and reference image using the color transfer method proposed in [67] i.e. with the set of mathematical equations (2.2.4.5.1.1.1) (given in **Chapter 2**) similarly as done by the authors in [66] but instead of using any daytime hazy image as reference image while performing color transfer as done by the authors in [66] and [70], Ancuti *et. al.* have performed color transfer using a reference image which is derived from the source image itself in CCT prior. The use of such generated reference images enables the CCT prior to provide better quality outputs compared to the methods proposed in [66] and [70] as the outputs provided by these methods often contain various artifacts owing to the variations in the scene characteristics between source and reference images as discussed in **Chapter 2**.

The mathematical equation used for generating the reference image from the source image in CCT prior is given as follows:

$$r(x) = Gray(x) + Detail(x) + Saliency(x)I(x) \quad (2.2.4.5.1.1.12)$$

This equation is also given in **Chapter 2**, hence the equation number is kept unchanged here to avoid any ambiguity.

In (2.2.4.5.1.1.12), the term '*Gray*' represents a uniform gray image whose luminance value is fixed close to 0.5 whereas the term '*Detail*' represents the detail layer of the image which is obtained by subtracting the Gaussian blurred version of the original image from the original image itself. This term is related to the transfer of information regarding the details of the original image to the generated reference image.

The term '*Saliency*' here signifies the saliency map of an image which is estimated using the method proposed by Achanta *et al.* in [74]. The product of this map and the original image provides a color bias which helps in restoration of original colors in output images.

An example of color transferred outputs for a nighttime hazy image (source image) obtained while using two different daytime hazy images as well as the reference image generated using the CCT method respectively are given in Fig.5.2.



Fig.5.2. Hazy (source) image, (b)-(c): Daytime hazy images, (d)-(e): Color transferred outputs obtained from [66] using (b) & (c) as reference images , (f) Color transferred output obtained from CCT

The images given in Fig.5.2 clearly depict that the colour transferred outputs obtained from [66] vary in accordance to the variations of color characteristics of daytime hazy (reference) images. Although CCT prior solves this limitation, but from Fig.5.2 (f), it becomes evident that CCT mostly produces distorted colour transferred outputs (containing over-enhanced glow regions and poorly enhanced low light regions) which have significant influence on the visual quality of dehazed outputs obtained from several dehazing methods where CCT is introduced as pre-processing step.

In order to solve this limitation of CCT prior, in proposed RCCT prior initially a constant 'k' which is termed as the control parameter is introduced in (2.2.4.5.1.1.12). Here, the name of the parameter is given as control parameter as it is used to restrict the intensity range of color transferred outputs within a certain limit (which is computed according to the image properties) to reduce the chances of occurrence of poorly illuminated low light regions and over-enhanced glow regions in outputs.

Following the introduction of control parameter, 'k' in (2.2.4.5.1.1.12), the modified equation becomes,

$$r(x) = G(x) + D(x) + k \times Sa_{new}(x)s_{new}(x) \quad (5.1)$$

The control parameter is multiplied with the product of saliency map of the source image and source image itself as this product term being the color bias term is responsible for determining the color and intensities of pixels in the generated reference images. Hence, restricting the value of this product for each pixel will also lead to the restriction of the color and intensity value of that pixel in the generated reference images.

Initially, a trial and error is adapted to select the most suitable value for 'k'. The color transferred outputs obtained using different constant values of 'k' are given in Fig.5.3.

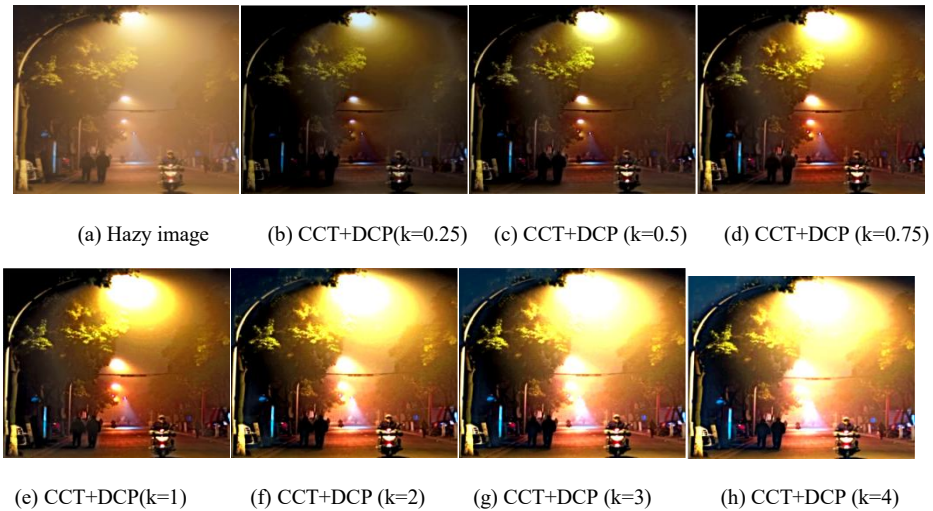


Fig.5.3. Color transferred outputs obtained using different values of 'k'

The above results suggest that desired outputs cannot be obtained using constant control parameter's values as fixing the control parameter to any value less than one although succeeds in restricting the intensities of the over-enhanced glow regions but do not give ideal results for regions having poor illumination. Similarly if the control parameter is fixed to any value greater than one, then in such cases the extent of over enhancement of glow regions in generated reference images increases.

To overcome this limitation, a Fuzzy Logic based reference image generation technique is proposed. This reference image generation technique allows to restrict the intensity range of each pixel in generated reference image depending upon its' properties. The novelty of RCCT prior lies in the proposed Fuzzy Logic based reference image generation technique.

5.1. Fuzzy Logic based reference image generation technique

In the proposed technique, reference image is generated from the source image using a slightly modified version of (5.1), i.e. by substituting the constant control parameter term in (5.1) with a variable term which is mathematically represented as follows:

$$r(x) = G(x) + D(x) + k(x) \times Sa_{new}(x) s_{new}(x) \quad (5.1.1)$$

The control parameter value for each pixel is calculated depending upon its saliency and intensity values.

As an image usually contains large number of pixels depending upon its dimension, hence calculating an unique control parameter value corresponding to each pixel is not only tedious but is also time-consuming.

To reduce the complexity of this problem, in this work, an image is initially divided into 'n' number of super pixels using the Simple Linear Iterative clustering method [86] before performing the computation of control parameter values.

The theory based on which the super-pixel concept is developed states that the pixels belonging to a super-pixel usually possess similar properties like intensities, brightness, texture, etc. Hence, from this concept, it can be said that computing an unique control parameter value for all the pixels belonging to same super-pixel will serve the purpose in this case. Here the value of 'n' is kept fixed at 500.

Following the segmentation of an image into 'n' number of super-pixels and before computing control parameter's values, some approximations are done which are mathematically defined as follows:

For example, the ' i^{th} ' super-pixel contains 'm' number of pixels, then in such case a value, ' in_i ' is calculated as mathematically defined in (5.1.2) which is given as follows:

$$in_i = \frac{\sum_{i=1}^m \left(\frac{(R_1 + G_1 + B_1)}{3} + \frac{(R_2 + G_2 + B_2)}{3} + \dots + \frac{(R_m + G_m + B_m)}{3} \right)}{m} \quad (5.1.2)$$

The intensities of all the 'm' pixels in the ' i^{th} ' super-pixel are substituted with the computed ' in_i ' values.

Similarly, the saliency values of all the 'm' pixels belonging to the ' i^{th} ' super-pixel are substituted with the computed ' Sa_i ' values. The computation of ' Sa_i ' values are done according to the following mathematical equation:

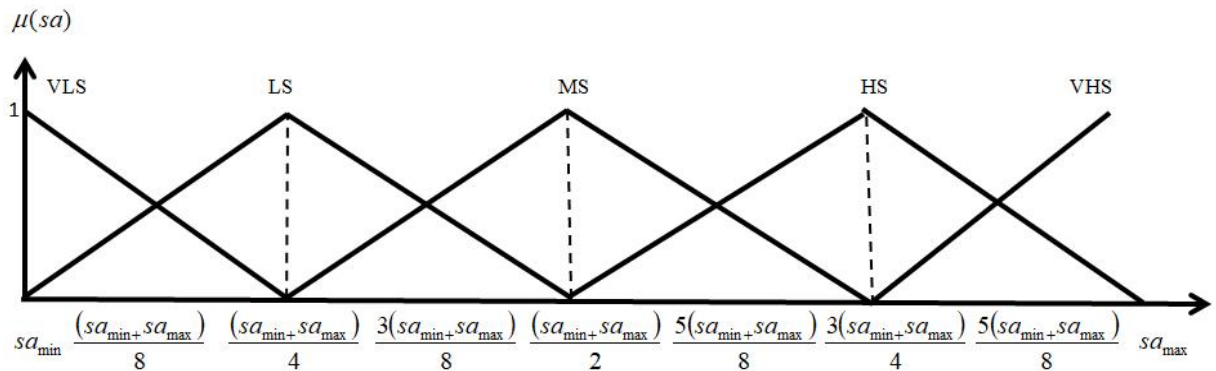
$$Sa_i = \frac{\sum_{i=1}^m (Sa_1 + Sa_2 + \dots + Sa_m)}{m} \quad (5.1.3)$$

Similar computations are also carried out for other super-pixels too.

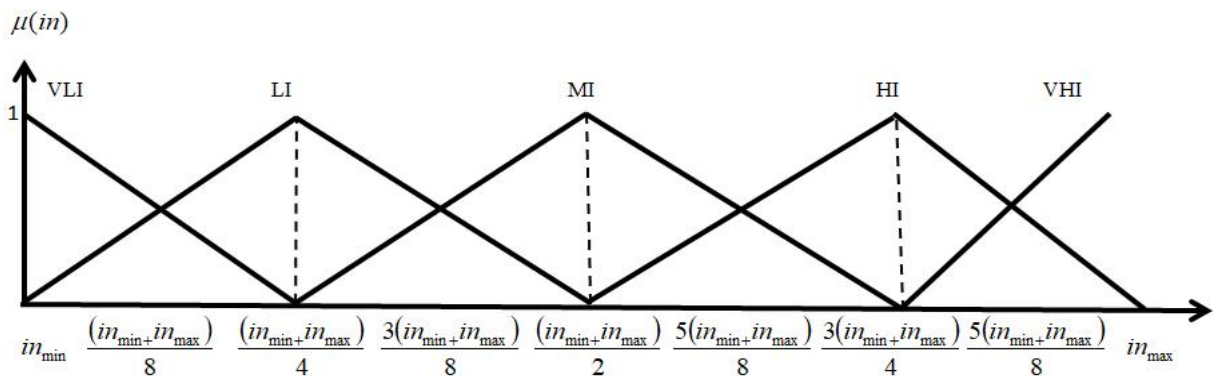
Following these substitutions, an unique control parameter value is computed for each super-pixel using the novel set of FI rules which is given in Table 5.1.1.

Table 5.1.1. FI rules designed for reference image generation

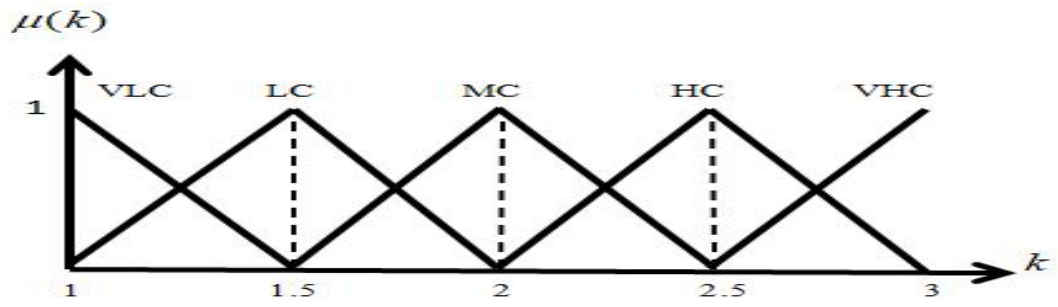
Rule No	If		Then
	<i>sa</i>	<i>in</i>	<i>k</i>
1	VERY LOW SALIENCY	VERY LOW INTENSITY	VERY LOW CONTROL
2	VERY LOW SALIENCY	LOW INTENSITY	LOW CONTROL
3	VERY LOW SALIENCY	MEDIUM INTENSITY	MEDIUM CONTROL
4	VERY LOW SALIENCY	HIGH INTENSITY	HIGH CONTROL
5	VERY LOW SALIENCY	VERY HIGH INTENSITY	VERY HIGH CONTROL
6	LOW SALIENCY	VERY LOW INTENSITY	VERY LOW CONTROL
7	LOW SALIENCY	LOW INTENSITY	LOW CONTROL
8	LOW SALIENCY	MEDIUM INTENSITY	MEDIUM CONTROL
9	LOW SALIENCY	HIGH INTENSITY	HIGH CONTROL
10	LOW SALIENCY	VERY HIGH INTENSITY	VERY HIGH CONTROL
11	MEDIUM SALIENCY	VERY LOW INTENSITY	VERY LOW CONTROL
12	MEDIUM SALIENCY	LOW INTENSITY	LOW CONTROL
13	MEDIUM SALIENCY	MEDIUM INTENSITY	MEDIUM CONTROL
14	MEDIUM SALIENCY	HIGH INTENSITY	HIGH CONTROL
15	MEDIUM SALIENCY	VERY HIGH INTENSITY	VERY HIGH CONTROL
16	HIGH SALIENCY	VERY LOW INTENSITY	VERY LOW CONTROL
17	HIGH SALIENCY	LOW INTENSITY	LOW CONTROL
18	HIGH SALIENCY	MEDIUM INTENSITY	MEDIUM CONTROL
19	HIGH SALIENCY	HIGH INTENSITY	HIGH CONTROL
20	HIGH SALIENCY	VERY HIGH INTENSITY	VERY HIGH CONTROL
21	VERY HIGH SALIENCY	VERY LOW INTENSITY	VERY LOW CONTROL
22	VERY HIGH SALIENCY	LOW INTENSITY	LOW CONTROL
23	VERY HIGH SALIENCY	MEDIUM INTENSITY	MEDIUM CONTROL
24	VERY HIGH SALIENCY	HIGH INTENSITY	HIGH CONTROL
25	VERY HIGH SALIENCY	VERY HIGH INTENSITY	VERY HIGH CONTROL



(a)

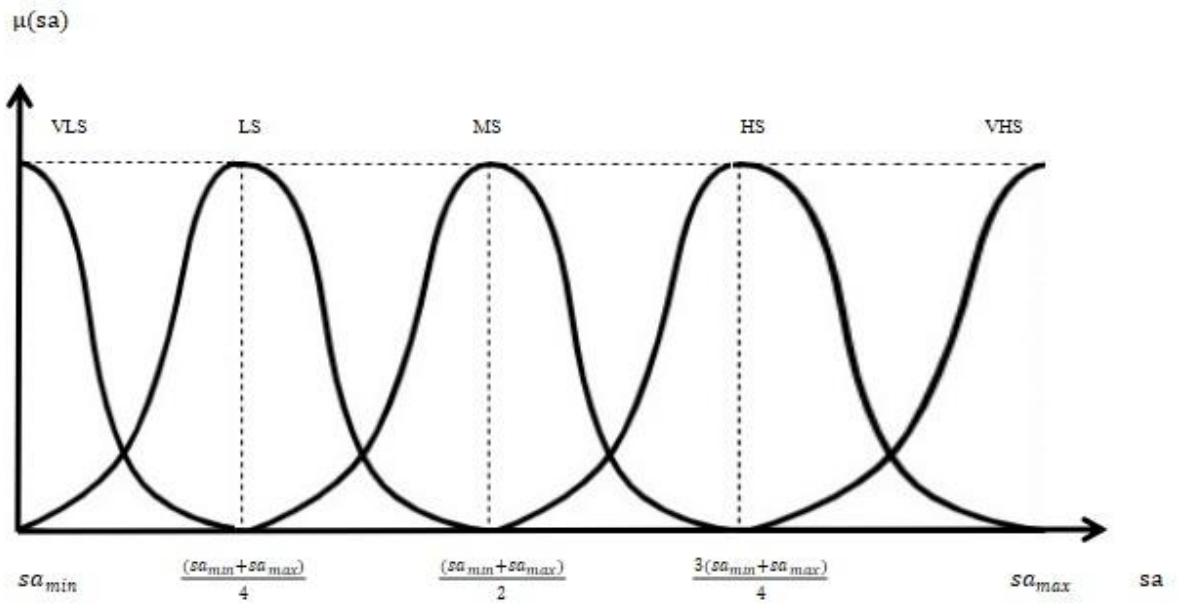


(b)

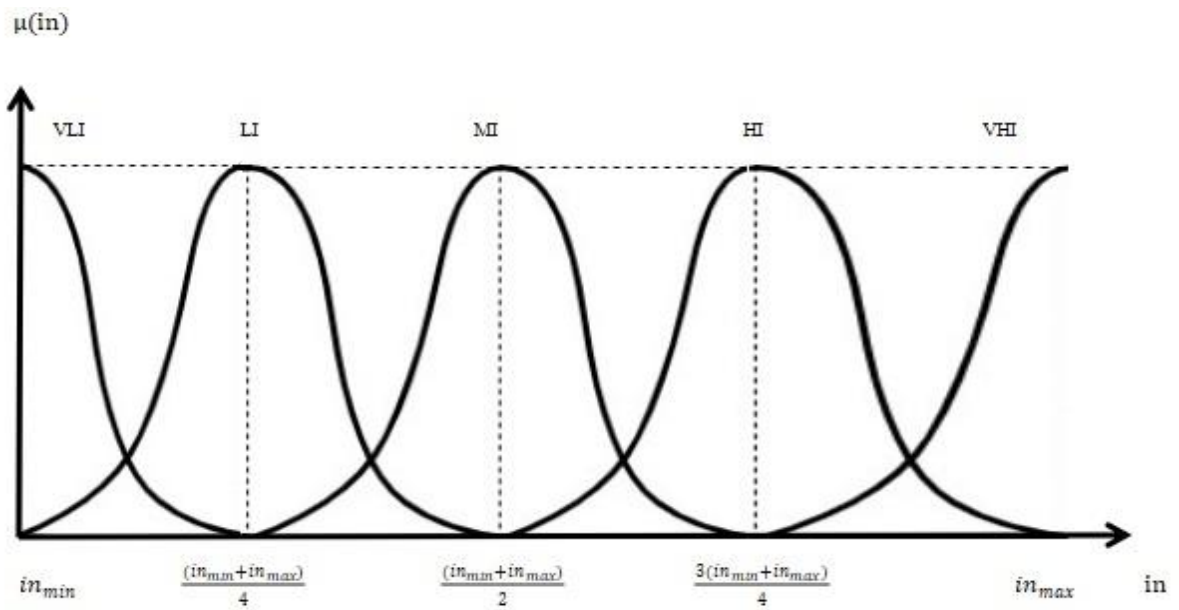


(c)

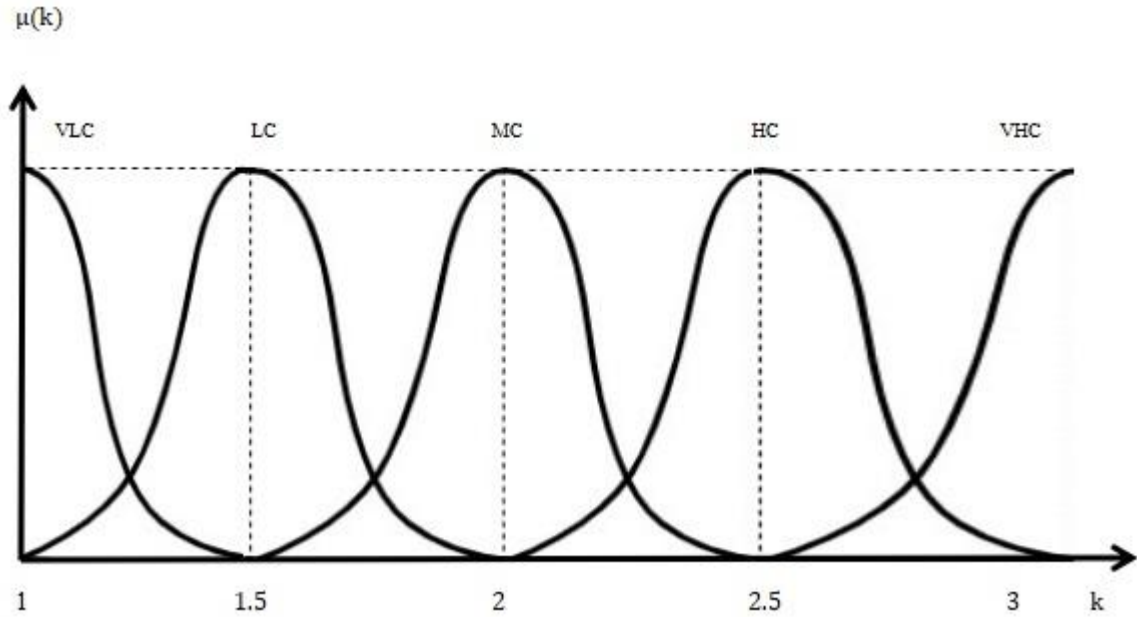
SET A



(a)



(b)



(c)

SET B

Fig.5.1.1. Graphical representation of fuzzy membership values of (a) sa, (b) in and (c) k

Abbreviations used in the above figure represent the following terms:

Linguistic values considered for representing input linguistic variable, (sa) are VLS: VERY LOW SALIENCY, LS: LOW SALIENCY, MS: MEDIUM SALIENCY, HS: HIGH SALIENCY, VHS: VERY HIGH SALIENCY.

Linguistic values considered for representing input linguistic variable, (intensity) are VLI: VERY LOW INTENSITY, LI: LOW INTENSITY, MI: MEDIUM INTENSITY, HI: HIGH INTENSITY, VHI: VERY HIGH INTENSITY.

Linguistic values considered for representing output linguistic variable, are VLC: VERY LOW CONTROL, LC: LOW CONTROL, MC: MEDIUM CONTROL, HC: HIGH CONTROL, VHC: VERY HIGH CONTROL.

DR considered for each linguistic variable are:

Linguistic variable, in = [in_{min} in_{max}] where in_{min} = min(in₁, in₂, ..., in_m) and in_{max} = max(in₁, in₂, ..., in_m)

Linguistic variable, sa = [sa_{min} sa_{max}] where sa_{min} = min(sa₁, sa₂, ..., sa_m) and sa_{max} = max(sa₁, sa₂, ..., sa_m)

Linguistic variable, k = [1 3] (empirically chosen)

5.2. Experimental Results

This Section contains comprehensive comparative qualitative and quantitative analyses' of results obtained by applying several state-of-the art daytime image dehazing methods as well as nighttime image dehazing methods on real world nighttime hazy images (before and after introduction of CCT and RCCT as pre-processing steps of the daytime image dehazing methods).

Details of quantitative parameters used here for performing quantitative analysis are given in Section 4.2 of Chapter 4.

Table 5.2.1. Comprehensive qualitative analysis results obtained using real world images

Methods	Image 1	Image 2	Image 3	Image 4
Hazy image				
He <i>et al.</i> [38]				
CCT+ [38]				
RCCT(using membership functions in Set A)+ [38]				
RCCT(using membership functions in Set B)+ [38]				
Meng <i>et al.</i> [44]				
CCT+ [44]				
RCCT(using membership functions in Set A)+ [44]				
RCCT(using membership functions in Set B)+ [44]				
Cai <i>et al.</i> [52]				
CCT+ [52]				

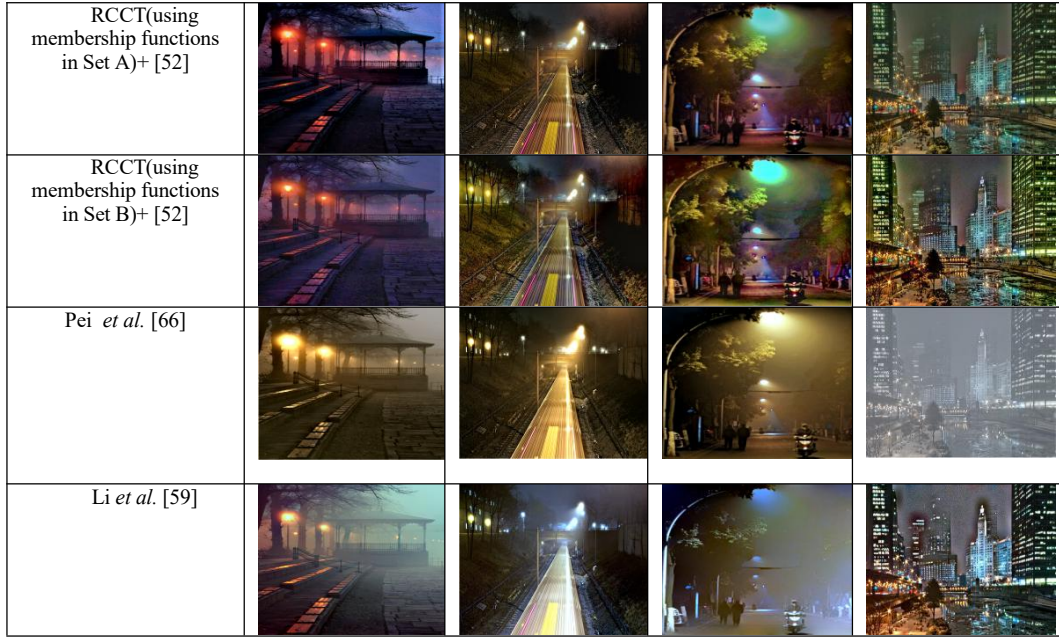


Table 5.2.2. Comprehensive quantitative analysis results obtained using real world images

Methods	Image 1			Image 2			Image 3			Image 4		
	BRISQUE	FADE	HI	BRISQUE	FADE	HI	BRISQUE	FADE	HI	BRISQUE	FADE	HI
He et al. [38]	15.7	0.607	7.50	7.25	0.314	7.55	28.70	0.420	7.41	25.87	0.494	26.95
CCT+ [38]	33.7	0.164	24.82	37.74	0.135	24.68	32.57	0.241	24.88	17.83	0.167	32.88
RCCT(using membership functions in Set A)+ [38]	3.07	0.148	32.59	7.254	0.135	32.22	18.82	0.209	32.84	12.89	0.143	37.54
RCCT(using membership functions in Set B)+ [38]	2.55	0.125	35.89	6.883	0.126	36.83	15.77	0.186	42.28	9.15	0.127	42.66
Meng et al. [44]	17.20	0.411	12.91	15.67	0.275	13.00	30.85	0.456	12.75	33.71	0.180	18.74
CCT+ [44]	31.83	0.278	28.90	14.93	0.253	28.69	24.22	0.314	29.01	28.49	0.114	22.65
RCCT(using membership functions in Set A)+ [44]	2.07	0.212	34.64	12.19	0.120	34.47	21.16	0.295	34.71	24.91	0.104	34.75
RCCT(using membership functions in Set B)+ [44]	1.88	0.278	28.95	8.96	0.098	25.78	18.89	0.201	39.06	18.58	0.085	47.89
Cai et al. [52]	29.95	0.365	7.67	35.51	0.208	7.72	35.69	0.366	7.60	28.48	0.265	11.75
CCT+ [52]	38.87	0.233	25.48	38.15	0.216	25.33	32.99	0.312	25.54	22.37	0.133	18.03
RCCT(using membership functions in Set A)+ [52]	2.18	0.206	33.49	15.22	0.157	33.10	23.85	0.234	33.76	17.94	0.125	25.43
RCCT(using membership functions in Set B)+ [53]	2.04	0.199	38.06	12.98	0.136	38.75	17.50	0.205	41.75	15.77	0.103	31.67
Pei et al. [66]	6.06	0.268	4.14	23.14	0.189	4.01	30.44	0.299	3.10	32.74	0.546	26.98
Li et al. [59]	3.86	0.506	3.17	26.64	0.245	3.04	36.55	1.075	2.13	28.95	0.417	14.66

The comparative analyses' results show how introduction of the newly proposed RCCT enables daytime dehazing methods to dehaze nighttime hazy images effectively and clearly depict the excellence of the proposed RCCT prior over such existing priors.

In Chapter 6, three new databases (S-HAZE, N-HAZE and D&N-HAZE) are introduced to facilitate reliable benchmarking of image dehazing methods. The presence of Ground Truth images corresponding to each hazy image in these databases not only supports subjective comparison of the obtained outputs with the actual scene content of images or in other words it can be said that the presence of Ground Truth images corresponding to each hazy image in these databases help in evaluation of the original scene restoration capabilities of the image dehazing methods but also allows researchers to perform quantitative analysis of the methods using both no-reference as well as full-reference quantitative parameters.

Chapter 6

Three new databases (S-HAZE, N-HAZE and D&N-HAZE) which are created as a part of this thesis work are introduced in this Chapter. Detailed description of the designed databases along with few examples of the images included in these databases are provided in this Chapter. The logic following which these databases are created are also given in details in this Chapter and the advantages of performing benchmarking of image dehazing methods using these databases are also highlighted here.

6.1. S-HAZE database

This database comprises of ‘synthetic’ daytime hazy images as well as GT image corresponding to each hazy image. The term ‘synthetic’ is used while describing the type of hazy images present in this database as the images are created synthetically by introducing haze in real world daytime images using the “Dehaze” filter option which is available under the “Effects” section in the “Adobe Lightroom App.” Two hazy images of varying haze density are included corresponding to each real world daytime image in this database. These hazy images are created by fixing the value of the “Dehaze” filter to +70 and +100 values respectively. The purpose of including two hazy images with similar scene contents but varying haze density in this database is to facilitate evaluation of the variations in performance efficiencies of image dehazing methods in accordance to the variations of haze density of images.

This database is created mainly focusing on sky regions of images because of its immense significance in the image dehazing field. Sky regions of daytime images usually possess high intensity pixels because of the presence of sunlight during daytime. As sunlight is considered as the main source of illumination in daytime images and as the influence of sunlight on sky regions is comparatively higher than the other regions of images, so sky regions are often chosen as the ideal location of images for the estimation of atmospheric light by many authors working in the image dehazing field (For.e.g. the authors who have proposed DCP [38] have estimated atmospheric light by averaging the intensities of 0.1% of the brightest pixels of the images assuming those pixels lie in the sky regions of images). In image dehazing methods which are designed following DCP concept as well as in other existing methods too, the authors have considered sky regions of images as ideal location for evaluating atmospheric light but they have mostly assumed that sky regions of images are white and contain no details which is not always true as sky regions of images usually possess varied characteristics like it may be white or blue or it mainly contain cloud or it may be clear and so on. In this database, images having different types of sky regions are included to evaluate the variations in performances of image dehazing methods in accordance to the variations of nature of sky regions of images and to evaluate the original scene restoration capabilities of the image dehazing methods (especially sky regions). Apart from the characteristics of sky regions, the other factor which is considered while selecting the images which are to be included in this database is the percentage occupancy (i.e. out of the total region of images how much region is occupied by sky-regions) . Hence, the images included in this database also have varied portions of sky regions like large sky regions, little sky regions and no sky regions in addition to different types of sky regions.

The inclusion of GT images in this database allows researchers to perform comparative quantitative analyses using both full-reference and no-reference quantitative parameters. An example of such quantitative analyses using two hazy images of varying haze density are given in Table 6.1.1.2 (using full-reference quantitative parameters) and Table 6.1.1.3 (using no-reference parameters). Some examples of images included in S-HAZE database are given in Table 6.1.1.1, Table 6.1.1.4 and Table 6.1.1.5.

The full-reference parameters considered for performing quantitative analysis in Table 6.1.1.2 are given below:

- **Structural Similarity Index (SSIM) [111]**
- **CIEDE2000 [112]**
- **DEHAZEfr [113]**

The details of these parameters are given in **Section 4.2.1 of Chapter 4**.

- **Peak Signal-to-Noise Ratio (PSNR) [103]**

The detail of this parameter is given in **Section 3.1.2.5 of Chapter 3**.

Brief details of full-reference parameters considered for performing quantitative analysis in Table 6.1.1.2 are given below:

- **Haze Improvement (HI) [114]**
- **Fog Aware Density Estimator (FADE) [115]**

The details of these parameters are given in **Section 4.2.2 of Chapter 4**.

- **e (Rate of new visible edge in restored images) [128]:** This parameter is designed to evaluate the edge restoration capability of the designed image dehazing methods. Mathematically, it is expressed as follows:

$$e = \frac{n_r - n_0}{n_0} \quad (6.1.1)$$

In (6.1.1), the terms n_r and n_0 are the cardinal numbers which is used to represent the sets of edges visible in dehazed outputs as well as original hazy images respectively.

- **\bar{r} (Gradient ratio at visible edges)[128]:** This parameter is designed to estimate the contrast restoration capabilities of image dehazing methods. Mathematically, it is expressed as follows:

$$\bar{r} = \exp\left[\frac{1}{n_r} \sum_{P_i \in \theta_r} \log r_i\right] \quad (6.1.2)$$

P_i denote the i^{th} pixel included in the set, θ_r which contains visible edges in dehazed outputs.

- **n_s [128] :** This parameter defines the number of saturated pixels present in dehazed outputs which are formed as a result of dehazing. Mathematically, this parameter is expressed as follows:





$$\sigma = \frac{n_s}{\dim_x \times \dim_y}$$

The terms, \dim_x and \dim_y denote the height and width of dehazed outputs.

6.1.1. Images belonging to S-HAZE database

Some examples of images having varied portions and types of sky regions which are included in S-HAZE database are given in the following Tables.

Table 6.1.1.1. Images having large sky regions

Methods	Image 1(light haze)	Image1(dense haze)	Image 2(light haze)	Image2(dense haze)
Hazy Images				


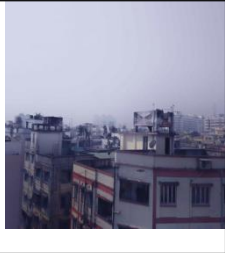
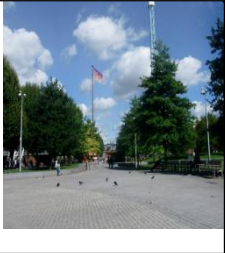
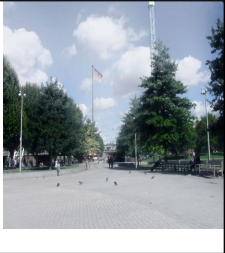

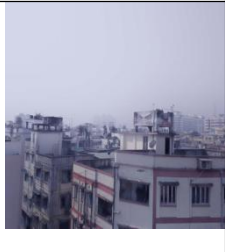

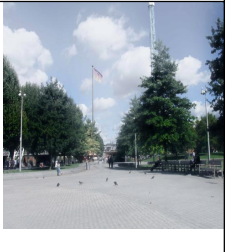
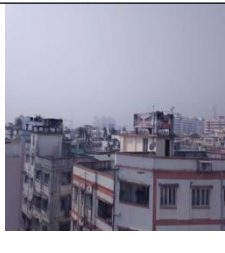
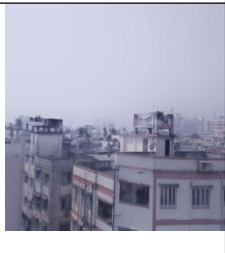






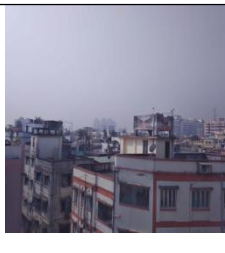
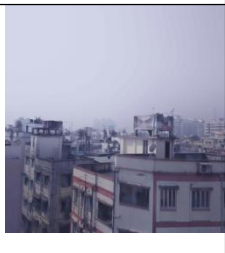



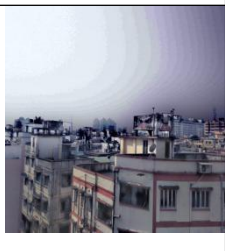



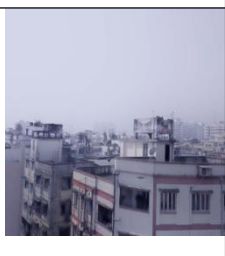
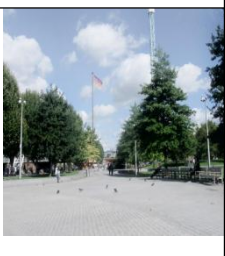

He <i>et al.</i> [38]				
Cai <i>et al.</i> [52]				
Ren <i>et al.</i> [53]				
Meng <i>et al.</i> [44]				
Zhu <i>et al.</i> [50]				
Berman <i>et al.</i> [49]				
He <i>et al.</i> [129](CO)				



Table 6.1.1.2. Comparative quantitative analysis results (using full-reference quantitative parameters)













Methods	Image 1(light haze)				Image 1(dense haze)			
	SSIM	PSNR	DEHAZEfr	CIEDE2000	SSIM	PSNR	DEHAZEfr	CIEDE2000
He <i>et al.</i> [38]	0.886	24.505	0.8717	14.349	0.791	18.587	0.7088	24.385
Cai <i>et al.</i> [52]	0.909	23.407	0.8770	22.131	0.779	16.154	0.6487	35.757
Ren <i>et al.</i> [53]	0.873	19.885	0.824	24.578	0.748	14.669	0.593	42.330
Meng <i>et al.</i> [44]	0.724	14.365	0.597	42.489	0.657	12.956	0.575	50.799
Zhu <i>et al.</i> [50]	0.886	24.162	0.862	12.596	0.771	17.029	0.655	30.179
Berman <i>et al.</i> [49]	0.819	16.211	0.587	7.782	0.836	17.921	0.632	28.977
He <i>et al.</i> [129](CO)	0.900	20.825	0.848	29.322	0.767	15.033	0.604	40.970
He <i>et al.</i> [129](CODHWT1)	0.899	20.865	0.854	29.293	0.768	15.041	0.608	40.938









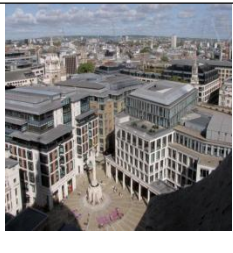










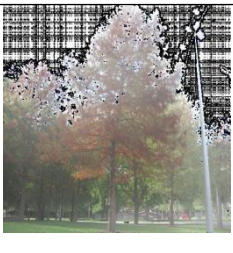



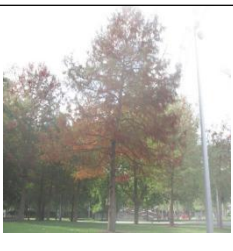



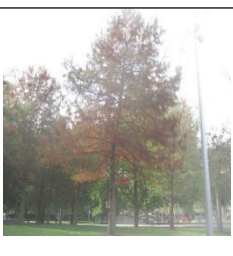
He <i>et al.</i> [129](CODHWT2)	0.899	20.890	0.858	29.215	0.767	15.046	0.608	40.917
Galdran <i>et al.</i> [117]	0.802	19.917	0.856	0.705	0.709	21.071	0.733	6.347
Salazar-Colores <i>et al.</i> [47]	0.878	16.969	0.903	39.170	0.757	13.662	0.735	50.224

Table 6.1.1.3. Comparative quantitative analysis results (using no-reference quantitative parameters)

Methods	Image 1(light haze)					Image 1(dense haze)				
	e	\bar{r}	n_s	FADE	HI	e	\bar{r}	n_s	FADE	HI
He <i>et al.</i> [38]	-0.849	5.184	0	1.628	-12.35	-0.774	5.082	0	2.167	0.41
Cai <i>et al.</i> [52]	-0.852	6.749	0	1.627	-10.16	-0.785	5.753	0	2.193	10.63
Ren <i>et al.</i> [53]	-0.857	6.975	0	2.322	-6.59	-0.801	5.538	0	2.979	2.45
Meng <i>et al.</i> [44]	-0.845	7.042	0.006	1.215	-39.36	-0.753	6.095	0	1.527	-11.39
Zhu <i>et al.</i> [50]	-0.857	6.386	0	2.615	-23.62	-0.791	5.517	0	2.325	-4.67
Berman <i>et al.</i> [49]	-0.839	7.230	0.032	0.462	-11.03	-0.763	6.680	0.007	1.068	10.77
He <i>et al.</i> [129](CO)	-0.852	7.019	0.003	1.573	-3.42	-0.790	5.966	0	2.541	7.66
He <i>et al.</i> [129](CODHWT1)	-0.851	6.952	0.003	1.539	-1.55	-0.789	5.934	0	2.558	7.73
He <i>et al.</i> [129](CODHWT2)	-0.851	6.944	0.009	1.531	-3.75	-0.789	5.912	0	2.557	7.74
Galdran <i>et al.</i> [117]	-0.853	7.784	0.006	1.442	30.55	-0.768	5.775	0	1.820	15.38
Salazar-Colores <i>et al.</i> [47]	-0.851	7.524	0	1.305	-12.35	-0.764	6.277	0	1.742	0.41

Table 6.1.1.4. Images having little sky regions

Methods	Image 1(light haze)	Image1(dense haze)	Image 2(light haze)	Image2(dense haze)
Hazy Images				
He <i>et al.</i> [38]				
Cai <i>et al.</i> [52]				

Ren et al. [53]				
Meng et al.[44]				
Zhu et al.[50]				
Berman et al. [49]				
He et al.[129](CO)				
He et al.[129](CODHWT1)				
He et al.[129](CODHWT2)				




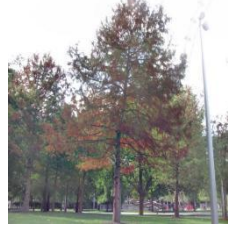



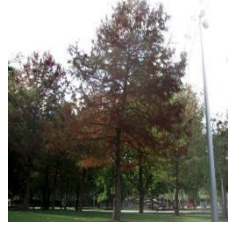



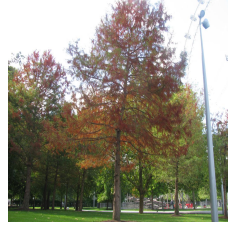





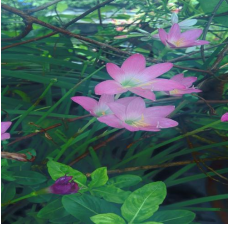



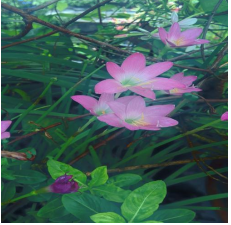






























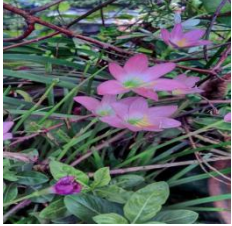
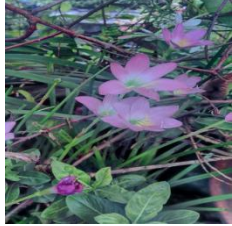





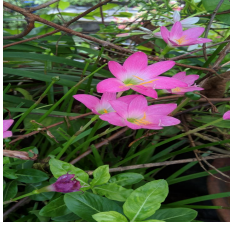
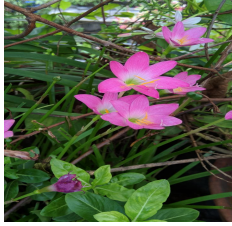


Galdran <i>et al.</i> [117]				
Salazar-Colores <i>et al.</i> [47]				
Ground Truth				

Table 6.1.1.5. Images having no sky regions

Methods	Image 1(light haze)	Image1(dense haze)	Image 2(light haze)	Image2(dense haze)
Hazy Images				
He <i>et al.</i> [38]				
Cai <i>et al</i> [52]				

Ren et al. [53]				
Meng et al.[44]				
Zhu et al.[50]				
Berman et al. [49]				
He et al.[129](CO)				
He et al.[129](CODHWT1)				
He et al.[129](CODHWT2)				

Galdran <i>et al.</i> [117]				
Salazar-Colores <i>et al.</i> [47]				
Ground Truth				

6.2. N-HAZE database

This database comprises of real world nighttime hazy images which are captured in the presence of haze created by artificial haze machines. Apart from hazy images, a GT image corresponding to each hazy image is included in this database to facilitate the researchers to perform subjective evaluation of visual quality of dehazed outputs with the actual scene content of images (i.e. GT images) as well as to perform quantitative analyses using both no-reference and full-reference quantitative parameters.

Before the creation of the N-HAZE database, comparative qualitative and quantitative analyses of nighttime image dehazing methods are performed using either a synthetic nighttime hazy image which is generated using Physically Based Rendering Technique (PBRT) [130] as done by the authors in [81] and [83] or by using real-world nighttime hazy image whose corresponding GT images are not available. So, N-HAZE database is the first database which is created for benchmarking nighttime image dehazing methods. N-HAZE database comprises of both real world nighttime hazy images as well as their corresponding GT images. Apart from nighttime hazy images and GT images, dehazed outputs which are obtained by applying popular as well recent image dehazing methods on nighttime hazy images belonging to this database are included in the database to enable the researchers to carry out effective benchmarking of dehazing methods. Some examples of images included in N-HAZE database are given Table 6.2.1.

Image 1_hazy	Image1_he_2011 [38]	Image1_meng_2013 [44]
		
Image1_ancuti_2013 [41]	Image1_zhu_2015 [50]	Image1_li_2015 [59]
		
Image1_he_CO_2016 [129]	Image1_he_CODHWT1_2016 [129]	Image1_he_CODHWT2_2016 [129]
		
Image1_ren_2016 [53]	Image1_cai_2016 [52]	Image1_berman_2016 [49]
		
Image1_santra_2016 [64]	Image1_galdran_2018 [117]	Image1_salazar-Colores_2018 [47]
		
Image1_yu_2019 [91]	Image1_ngo_2020 [118]	Image1_shin_2020 [131]
		
Image1_li_2021 [132]	Image1_ju_2021[133]	Image1_hazefree (GT)
		


Image 2_hazy	Image2_he_2011 [38]	Image2_meng_2013[44]
		
Image2_ancuti_2013[41]	Image2_zhu_2015[50]	Image2_li_2015 [59]
		
Image2_he_CO_2016[129]	Image2_he_CODHWT1_2016 [129]	Image2_he_CODHWT2_2016 [129]
		
Image2_ren_2016 [53]	Image2_cai_2016 [52]	Image2_berman_2016 [49]
		
Image2_santra_2016 [64]	Image2_galdran_2018 [117]	Image2_salazar-Colores_2018 [47]
		
Image2_yu_2019 [91]	Image2_ngo_2020 [118]	Image2_shin_2020 [131]
		
Image2_li_2021 [132]	Image2_ju_2021[133]	Image2_hazefree (GT)
		



Image 3_hazy	Image3_he_2011[38]	Image3_meng_2013 [44]
Image3_ancuti_2013[41]	Image3_zhu_2015 [50]	Image3_li_2015 [59]
Image3_he_CO_2016 [129]	Image3_he_CODHWT1_2016 [129]	Image3_he_CODHWT2_2016[129]
Image3_ren_2016 [53]	Image3_cai_2016 [52]	Image3_berman_2016 [49]
Image3_santra_2016 [64]	Image3_galdran_2018[117]	Image3_salazar-Colores_2018 [47]
Image3_yu_2019 [91]	Image3_ngo_2020 [118]	Image3_shin_2020 [131]

Image3_li_2021 [132]	Image3_ju_2021 [133]	Image3_hazefree (GT)
		

Image 4_hazy	Image4_he_2011[38]	Image4_meng_2013[44]
		
Image4_ancuti_2013[41]	Image4_zhu_2015[50]	Image4_li_2015[59]
		
Image4_he_CO_2016[129]	Image4_he_CODHWT1_2016[129]	Image4_he_CODHWT2_2016[129]
		
Image4_ren_2016[53]	Image4_cai_2016[52]	Image4_berman_2016[49]
		
Image4_santra_2016[64]	Image4_galdran_2018[117]	Image4_salazar-Colores_2018[47]
		
Image4_yu_2019[91]	Image4_ngo_2020[118]	Image4_shin_2020[131]
		

		
Image4_li_2021 [132]	Image4_ju_2021[133]	Image4_hazefree (GT)
		

Image 5_hazy	Image5_he_2011[38]	Image5_meng_2013[44]
		
Image5_ancuti_2013[41]	Image5_zhu_2015[50]	Image5_li_2015[59]
		
Image5_he_CO_2016[129]	Image5_he_CODHWT1_2016[129]	Image5_he_CODHWT2_2016[129]
		
Image5_ren_2016[53]	Image5_cai_2016[52]	Image5_berman_2016[49]
		
Image5_santra_2016[64]	Image5_galdran_2018[117]	Image5_salazar-Colores_2018[47]
		



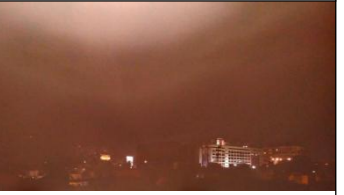



Image5_yu_2019[91]	Image5_ngo_2020[118]	Image5_shin_2020[131]
		
Image5_li_2021 [132]	Image5_ju_2021[133]	Image5_hazefree (GT)
		

Image 6_hazy	Image6_he_2011[38]	Image6_meng_2013[44]
		
Image6_ancuti_2013[41]	Image6_zhu_2015[50]	Image6_li_2015[59]
		
Image6_he_CO_2016[129]	Image6_he_CODHWT1_2016[129]	Image6_he_CODHWT2_2016 [129]
		
Image6_ren_2016 [53]	Image6_cai_2016 [52]	Image6_berman_2016 [49]
		
Image6_santra_2016[64]	Image6_galdran_2018[117]	Image6_salazar-Colores_2018 [47]



6.3. D&N-HAZE Database

This database contains a total of 330 images out of which 24 images are hazy and haze-free (GT) images of four scenes which are captured during both daytime as well as nighttime while others are output images which are obtained by applying several well-known image dehazing methods on hazy images (both synthetically generated hazy images as well as real hazy images) present in this database. For a particular scene, in this database four hazy images and two GT images are included. For e.g. for image 1(which is also referred to as scene 1), four images (image1_daytime_realhaze, image1_daytime_synthetic haze, image1_nighttime_realhaze and image1_nighttime_synthetic haze) and two haze-free images (image1_daytime_hazefree and image1_nighttime_hazefree) are included in this database. The images whose names include “_realhaze term” are captured in the presence of real atmospheric haze and the images whose names include _synthetic haze term are obtained by synthetically introducing haze in images using the “Dehaze” filter option present under the “Effects” section in the “Adobe Lightroom App”. Here the value of “Dehaze” filter is kept fixed at +100 while generating the hazy images. The images whose names include “_ hazefree term” are captured during clear weather conditions. Apart from hazy and haze-free images of each scene, in this database the outputs obtained by applying various well-known image dehazing methods on hazy images are also included to facilitate effective benchmarking of image dehazing methods. Those outputs are named as “imagename_type of haze (realhaze/synthetic haze)_first author surname_publication year”.

This is the first database created for benchmarking image dehazing methods which contain daytime and nighttime synthetic hazy images as well as hazy images captured in the presence of real atmospheric haze of similar scenes. Like S-HAZE database and N-HAZE database, this database too contain GT images and thus facilitates researchers to perform quantitative analyses using both full-reference and no-reference quantitative parameters. Examples of some images included in D&N-HAZE database are given below:



















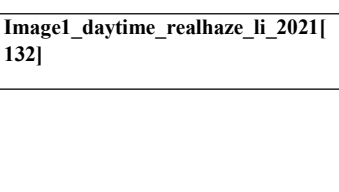
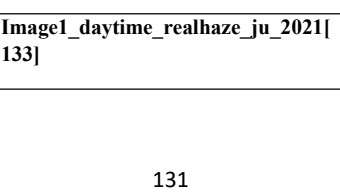
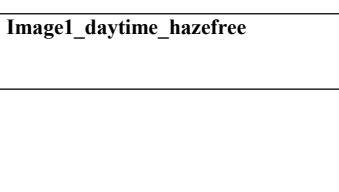
Image1_daytime_realhaze	Image1_daytime_realhaze_he_2011 [38]	Image1_daytime_realhaze_meng_2013[44]
		
Image1_daytime_realhaze_ancuti_2013[41]	Image1_daytime_realhaze_zhu_2015[50]	Image1_daytime_realhaze_li_2015[59]
		
Image1_daytime_realhaze_he_CO_2016[129]	Image1_daytime_realhaze_he_COD HWT1_2016[129]	Image1_daytime_realhaze_he_COD HWT2_2016[129]
		
Image1_daytime_realhaze_ren_2016 [53]	Image1_daytime_realhaze_cai_2016 [52]	Image1_daytime_realhaze_berman_2016 [49]
		
Image1_daytime_realhaze_santra_2016 [64]	Image1_daytime_realhaze_galdran_2018 [117]	Image1_daytime_realhaze_salazar-Colores_2018 [47]
		
Image1_daytime_realhaze_yu_2019 [91]	Image1_daytime_realhaze_ngo_2020[118]	Image1_daytime_realhaze_shin_2020[131]
		
Image1_daytime_realhaze_li_2021[132]	Image1_daytime_realhaze_ju_2021[133]	Image1_daytime_hazefree
		



Image1_daytime_syntheticchaze	Image1_daytime_syntheticchaze_he_2011[38]	Image1_daytime_syntheticchaze_meng_2013[44]
Image1_daytime_syntheticchaze_ancuti_2013[41]	Image1_daytime_syntheticchaze_zhu_2015[50]	Image1_daytime_syntheticchaze_li_2015[59]
Image1_daytime_syntheticchaze_he_CO_2016[129]	Image1_daytime_syntheticchaze_he_CODHWT1_2016[129]	Image1_daytime_syntheticchaze_he_CODHWT2_2016[129]
Image1_daytime_syntheticchaze_ren_2016[53]	Image1_daytime_syntheticchaze_cai_2016[52]	Image1_daytime_syntheticchaze_berman_2016[49]
Image1_daytime_syntheticchaze_santora_2016[64]	Image1_daytime_syntheticchaze_galdran_2018[117]	Image1_daytime_syntheticchaze_salazar-Colores_2018[47]
Image1_daytime_syntheticchaze_yu_2019[91]	Image1_daytime_syntheticchaze_ngon_2020[118]	Image1_daytime_syntheticchaze_shin_2020[131]

Image1_daytime_synthetic_haze_li_2021[132]	Image1_daytime_synthetic_haze_ju_2021[133]	Image1_daytime_hazefree
Image1_nighttime_realhaze	Image1_nighttime_realhaze_he_2011[38]	Image1_nighttime_realhaze_meng_2013[44]
Image1_nighttime_realhaze_ancuti_2013[41]	Image1_nighttime_realhaze_zhu_2015[50]	Image1_nighttime_realhaze_li_2015[59]
Image1_nighttime_realhaze_he_CO_2016[129]	Image1_nighttime_realhaze_he_CO_DHWT1_2016[129]	Image1_nighttime_realhaze_he_CO_DHWT2_2016[129]
Image1_nighttime_realhaze_ren_2016[53]	Image1_nighttime_realhaze_cai_2016[52]	Image1_nighttime_realhaze_berman_2016[49]
Image1_nighttime_realhaze_santra_2016[64]	Image1_nighttime_realhaze_galdra_n_2018[117]	Image1_nighttime_realhaze_salazar-Colores_2018[47]

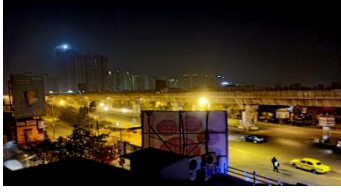
























		
Image1_nighttime_realhaze_yu_2019[91]	Image1_nighttime_realhaze_ngo_2020[118]	Image1_nighttime_realhaze_shin_2020[131]
		
Image1_nighttime_realhaze_li_2021[132]	Image1_nighttime_realhaze_ju_2021[133]	Image1_nighttime_hazefree
		

		
Image1_nighttime_synthetic_haze_4	Image1_nighttime_synthetic_haze_5	Image1_nighttime_synthetic_haze_6
		
Image1_nighttime_synthetic_haze_7	Image1_nighttime_synthetic_haze_8	Image1_nighttime_synthetic_haze_9
		
Image1_nighttime_synthetic_haze_10	Image1_nighttime_synthetic_haze_11	Image1_nighttime_synthetic_haze_12
		
Image1_nighttime_synthetic_haze_13	Image1_nighttime_synthetic_haze_14	Image1_nighttime_synthetic_haze_15

		
Image1_nighttime_synthetic haze_s antra_2016[64]	Image1_nighttime_synthetic haze_ga ldran_2018[117]	Image1_nighttime_synthetic haze_sa lazar-Colores_2018[47]
		
Image1_nighttime_synthetic haze_y u_2019[91]	Image1_nighttime_synthetic haze_ng o_2020[118]	Image1_nighttime_synthetic haze_s hin_2020[131]
		
Image1_nighttime_synthetic haze_li _2021 [132]	Image1_nighttime_synthetic haze_ju _2021[133]	Image1_nighttime_hazefree
		
Image2_daytime_realhaze	Image2_daytime_realhaze_he_2011 [38]	Image2_daytime_realhaze_meng_2 013[44]
		
Image2_daytime_realhaze_ancuti_ 2013[41]	Image2_daytime_realhaze_zhu_201 5[50]	Image2_daytime_realhaze_li_2015[59]
		
Image2_daytime_realhaze_he_CO_ 2016[129]	Image2_daytime_realhaze_he_COD HWT1_2016[129]	Image2 daytime_realhaze_he_COD HWT2_2016[129]










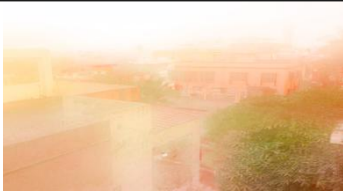







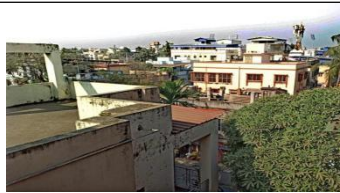








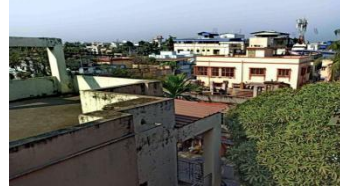









		
Image2_daytime_realhaze_ren_2016[53]	Image2_daytime_realhaze_cai_2016[52]	Image2_daytime_realhaze_berman_2016[49]
		
Image2_daytime_realhaze_santra_2016[64]	Image2_daytime_realhaze_galdran_2018[117]	Image2_daytime_realhaze_salazar-Colores_2018[47]
		
Image2_daytime_realhaze_yu_2019[91]	Image2_daytime_realhaze_ngo_2020[118]	Image2_daytime_realhaze_shin_2020[131]
		
Image2_daytime_realhaze_li_2021[132]	Image2_daytime_realhaze_ju_2021[133]	Image2_daytime_hazefree
		

Image2_daytime_syntheticchaze	Image2_daytime_syntheticchaze_he_2011[38]	Image2_daytime_syntheticchaze_meng_2013[44]
		
Image2_daytime_syntheticchaze_an_cuti_2013[41]	Image2_daytime_syntheticchaze_zhu_2015[50]	Image2_daytime_syntheticchaze_li_2015[59]

		
Image2_daytime_synthetic haze_he_CO_2016[129]	Image2_daytime_synthetic haze_he_CODHWT1_2016[129]	Image2_daytime_synthetic haze_he_CODHWT2_2016[129]
		
Image2_daytime_synthetic haze_re_n_2016[53]	Image2_daytime_synthetic haze_cai_2016[52]	Image2_daytime_synthetic haze_berman_2016[49]
		
Image2_daytime_synthetic haze_sa_ntra_2016[64]	Image2_daytime_synthetic haze_galdran_2018[117]	Image2_daytime_synthetic haze_salazar-Colores_2018[47]
		
Image2_daytime_synthetic haze_yu_2019[91]	Image2_daytime_synthetic haze_ngo_2020[118]	Image2_daytime_synthetic haze_shi_n_2020[131]
		
Image2_daytime_synthetic haze_li_2021[132]	Image2_daytime_synthetic haze_ju_2021[133]	Image2_daytime_hazefree
		
Image2_nighttime_realhaze	Image2_nighttime_realhaze_he_2011[38]	Image2_nighttime_realhaze_meng_2013 [44]



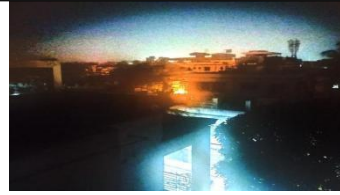















		
Image2_nighttime_realhaze_ancuti_2013[41]	Image2_nighttime_realhaze_zhu_2015[50]	Image2_nighttime_realhaze_li_2015[59]
		
Image2_nighttime_realhaze_he_CO_2016[129]	Image2_nighttime_realhaze_he_CO_DHWT1_2016[129]	Image2_nighttime_realhaze_he_CO_DHWT2_2016[129]
		
Image2_nighttime_realhaze_ren_2016[53]	Image2_nighttime_realhaze_cai_2016[52]	Image2_nighttime_realhaze_berman_2016[49]
		
Image2_nighttime_realhaze_santra_2016[64]	Image2_nighttime_realhaze_galdran_2018[117]	Image2_nighttime_realhaze_salazar-Colores_2018[47]
		
Image2_nighttime_realhaze_yu_2019[91]	Image2_nighttime_realhaze_ngo_2020[118]	Image2_nighttime_realhaze_shin_2020[131]
		
Image2_nighttime_realhaze_li_2021[132]	Image2_nighttime_realhaze_ju_2021[133]	Image2_nighttime_hazefree



Image2_nighttime_syntheticchaze	Image2_nighttime_syntheticchaze_he_2011[38]	Image2_nighttime_syntheticchaze_meng_2013[44]
Image2_nighttime_syntheticchaze_a_ncuti_2013[41]	Image2_nighttime_syntheticchaze_zhu_2015[50]	Image2_nighttime_syntheticchaze_li_2015[59]
Image2_nighttime_syntheticchaze_he_CO_2016[129]	Image2_nighttime_syntheticchaze_he_CODHWT1_2016[129]	Image2_nighttime_syntheticchaze_he_CODHWT2_2016[129]
Image2_nighttime_syntheticchaze_ren_2016[53]	Image2_nighttime_syntheticchaze_cai_2016[52]	Image2_nighttime_syntheticchaze_berman_2016[49]
Image2_nighttime_syntheticchaze_santra_2016[64]	Image2_nighttime_syntheticchaze_galdran_2018[117]	Image2_nighttime_syntheticchaze_salazar-Colores_2018[47]
Image2_nighttime_syntheticchaze_yu_2019 [91]	Image2_nighttime_syntheticchaze_ngo_2020[118]	Image2_nighttime_syntheticchaze_shin_2020[131]








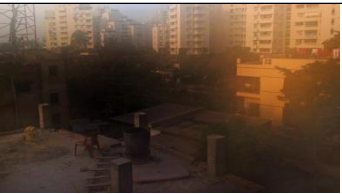
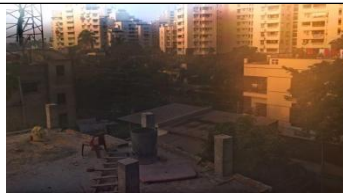






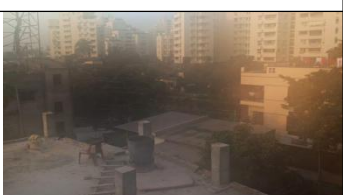
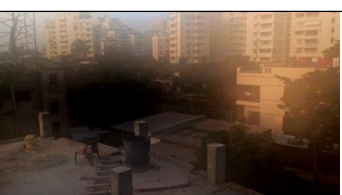




















		
Image2_nighttime_synthetic haze li_2021 [132]	Image2_nighttime_synthetic haze ju_2021 [133]	Image2_nighttime haze free
		
Image3_daytime_real haze	Image3_daytime_real haze he_2011 [38]	Image3_daytime_real haze meng_2013 [44]
		
Image3_daytime_real haze ancuti_2013 [41]	Image3_daytime_real haze zhu_2015 [50]	Image3_daytime_real haze li_2015 [59]
		
Image3_daytime_real haze he_CO_2016 [129]	Image3_daytime_real haze he_COD HWT1_2016 [129]	Image3_daytime_real haze he_CODH WT2_2016 [129]
		
Image3_daytime_real haze ren_2016 [53]	Image3_daytime_real haze cai_2016 [52]	Image3_daytime_real haze berman_2016 [49]
		
Image3_daytime_real haze santra_2016 [64]	Image3_daytime_real haze galdran_2018 [117]	Image3_daytime_real haze salazar-Colores_2018 [47]

		
Image3_daytime_realhaze_yu_2019[91]	Image3_daytime_realhaze_ngo_2020[118]	Image3_daytime_realhaze_shin_2020[131]
		
Image3_daytime_realhaze_li_2021[132]	Image3_daytime_realhaze_ju_2021[133]	Image3_daytime_hazefree
		

Image3_daytime_syntheticchaze	Image3_daytime_syntheticchaze_he_2011[38]	Image3_daytime_syntheticchaze_meng_2013[44]
		
Image3_daytime_syntheticchaze_ancuti_2013[41]	Image3_daytime_syntheticchaze_zhu_2015[50]	Image3_daytime_syntheticchaze_li_2015[59]
		
Image3_daytime_syntheticchaze_he_CO_2016[129]	Image3_daytime_syntheticchaze_he_CODHWT1_2016[129]	Image3_daytime_syntheticchaze_he_CODHWT2_2016[129]
		
Image3_daytime_syntheticchaze_ren_2016[53]	Image3_daytime_syntheticchaze_cai_2016[52]	Image3_daytime_syntheticchaze_berman_2016[49]

		
Image3_daytime_synthetic haze_sastran_2016[64]	Image3_daytime_synthetic haze_galdran_2018[117]	Image3_daytime_synthetic haze_salazar-Colores_2018[47]
		
Image3_daytime_synthetic haze_yu_2019[91]	Image3_daytime_synthetic haze_ngo_2020[118]	Image3_daytime_synthetic haze_shin_2020[131]
		
Image3_daytime_synthetic haze_li_2021 [132]	Image3_daytime_synthetic haze_ju_2021[133]	Image3_daytime_hazefree
		
Image3_nighttime_realhaze	Image3_nighttime_realhaze_he_2011[38]	Image3_nighttime_realhaze_meng_2013[44]
		
Image3_nighttime_realhaze_ancuti_2013[41]	Image3_nighttime_realhaze_zhu_2015[50]	Image3_nighttime_realhaze_li_2015 [59]
		
Image3_nighttime_realhaze_he_CO_2016[129]	Image3_nighttime_realhaze_he_CO DHWT1_2016[129]	Image3_nighttime_realhaze_he_CO DHWT2_2016[129]


































		
Image3_nighttime_realhaze_ren_2016[53]	Image3_nighttime_realhaze_cai_2016[52]	Image3_nighttime_realhaze_berman_2016[49]
		
Image3_nighttime_realhaze_santra_2016[64]	Image3_nighttime_realhaze_galdron_2018[117]	Image3_nighttime_realhaze_salazar-Colores_2018[47]
		
Image3_nighttime_realhaze_yu_2019[91]	Image3_nighttime_realhaze_ngo_2020[118]	Image3_nighttime_realhaze_shin_2020[131]
		
Image3_nighttime_realhaze_li_2020[132]	Image3_nighttime_realhaze_ju_2021[133]	Image3_nighttime_hazefree
		

Image3_nighttime_synthetic haze	Image3_nighttime_synthetic haze_he_2011[38]	Image3_nighttime_synthetic haze_meng_2013[44]
		
Image3_nighttime_synthetic haze_a.ncuti_2013[41]	Image3_nighttime_synthetic haze_zh_u_2015[50]	Image3_nighttime_synthetic haze_li_2015[59]

		
Image3_nighttime_synthetic haze_he_CO_2016[129]	Image3_nighttime_synthetic haze_he_CODHWT1_2016[129]	Image3_nighttime_synthetic haze_he_CODHWT2_2016[129]
		
Image3_nighttime_synthetic haze_ren_2016 [53]	Image3_nighttime_synthetic haze_cai_2016 [52]	Image3_nighttime_synthetic haze_berman_2016 [49]
		
Image3_nighttime_synthetic haze_santra_2016 [64]	Image3_nighttime_synthetic haze_galdran_2018 [117]	Image3_nighttime_synthetic haze_salazar-Colores_2018 [47]
		
Image3_nighttime_synthetic haze_yu_2019[91]	Image3_nighttime_synthetic haze_ngo_2020[118]	Image3_nighttime_synthetic haze_sahin_2020[131]
		
Image3_nighttime_synthetic haze_li_2021 [132]	Image3_nighttime_synthetic haze_ju_2021[133]	Image3_nighttime_hazefree
		
Image4_daytime_realhaze	Image4_daytime_realhaze_he_2011[38]	Image4_daytime_realhaze_meng_2013[44]







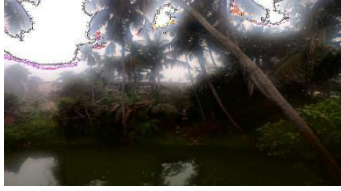


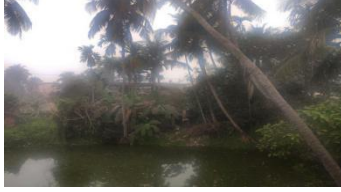




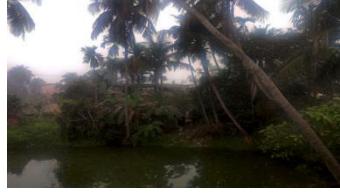



		
Image4_daytime_realhaze_ancuti_2013[41]	Image4_daytime_realhaze_zhu_2015[50]	Image4_daytime_realhaze_li_2015[59]
		
Image4_daytime_realhaze_he_CO_2016[129]	Image4_daytime_realhaze_he_CODHWT1_2016[129]	Image4_daytime_realhaze_he_CODHWT2_2016[129]
		
Image4_daytime_realhaze_ren_2016[53]	Image4_daytime_realhaze_cai_2016[52]	Image4_daytime_realhaze_berman_2016[49]
		
Image4_daytime_realhaze_santra_2016[64]	Image4_daytime_realhaze_galdran_2018[117]	Image4_daytime_realhaze_salazar-Colores_2018[47]
		
Image4_daytime_realhaze_yu_2019[91]	Image4_daytime_realhaze_ngo_2020[118]	Image4_daytime_realhaze_shin_2020[131]
		
Image4_daytime_realhaze_li_2021[132]	Image4_daytime_realhaze_ju_2021[133]	Image4_daytime_hazefree



Image4_daytime_syntheticchaze	Image4_daytime_syntheticchaze_he_2011[38]	Image4_daytime_syntheticchaze_meng_2013[44]
		
Image4_daytime_syntheticchaze_ancuti_2013[41]	Image4_daytime_syntheticchaze_zhu_2015[50]	Image4_daytime_syntheticchaze_li_2015[59]
		
Image4_daytime_syntheticchaze_he_CO_2016[129]	Image4_daytime_syntheticchaze_he_CODHWT1_2016[129]	Image4_daytime_syntheticchaze_he_CODHWT2_2016[129]
		
Image4_daytime_syntheticchaze_ren_2016[53]	Image4_daytime_syntheticchaze_cai_2016[52]	Image4_daytime_syntheticchaze_berman_2016[49]
		
Image4_daytime_syntheticchaze_santana_2016[64]	Image4_daytime_syntheticchaze_galdran_2018[117]	Image4_daytime_syntheticchaze_salazar-Colores_2018[47]
		
Image4_daytime_syntheticchaze_yu_2019[91]	Image4_daytime_syntheticchaze_ngon_2020 [118]	Image4_daytime_syntheticchaze_shin_2020[131]

		
Image4_daytime_synthetic haze_li_2021 [132]	Image4_daytime_synthetic haze_ju_2021[133]	Image4_daytime_hazefree
		
Image4_nighttime_realhaze	Image4_nighttime_realhaze_he_2011[38]	Image4_nighttime_realhaze_meng_2013[44]
		
Image4_nighttime_realhaze_ancuti_2013[41]	Image4_nighttime_realhaze_zhu_2015[50]	Image4_nighttime_realhaze_li_2015[59]
		
Image4_nighttime_realhaze_he_CO_2016[129]	Image4_nighttime_realhaze_he_CO DHWT1_2016[129]	Image4_nighttime_realhaze_he_CO DHWT2_2016[129]
		
Image4_nighttime_realhaze_ren_2016[53]	Image4_nighttime_realhaze_cai_2016[52]	Image4_nighttime_realhaze_berman_2016[49]
		
Image4_nighttime_realhaze_santra_2016[64]	Image4_nighttime_realhaze_galdra_n_2018[117]	Image4_nighttime_realhaze_salazar-Colores_2018[47]

		
Image4_nighttime_realhaze_yu_2019[91]	Image4_nighttime_realhaze_ngo_2020[118]	Image4_nighttime_realhaze_shin_2020[131]
		
Image4_nighttime_realhaze_li_2021 [132]	Image4_nighttime_realhaze_ju_2021[133]	Image4_nighttime_hazefree
		

Image4_nighttime_synthetic haze	Image4_nighttime_synthetic haze_he_2011[38]	Image4_nighttime_synthetic haze_meng_2013[44]
		
Image4_nighttime_synthetic haze_aucuti_2013[41]	Image4_nighttime_synthetic haze_zhu_2015[50]	Image4_nighttime_synthetic haze_li_2015[59]
		
Image4_nighttime_synthetic haze_he_CO_2016[129]	Image4_nighttime_synthetic haze_he_CODHWT1_2016[129]	Image4_nighttime_synthetic haze_he_CODHWT2_2016 [129]
		
Image4_nighttime_synthetic haze_ren_2016 [53]	Image4_nighttime_synthetic haze_cai_2016 [52]	Image4_nighttime_synthetic haze_berman_2016[49]

		
Image4_nighttime_synthetic haze_s antra_2016[64]	Image4_nighttime_synthetic haze_ga ldran_2018[117]	Image4_nighttime_synthetic haze_sa lazar-Colores_2018[47]
		
Image4_nighttime_synthetic haze_y u_2019[91]	Image4_nighttime_synthetic haze_ng o_2020[118]	Image4_nighttime_synthetic haze_s hin_2020[131]
		
Image4_nighttime_synthetic haze_li _2021 [132]	Image4_nighttime_synthetic haze_ju _2021[133]	Image4_nighttime_hazefree
		

All these three databases (S-HAZE, N-HAZE and D&N-HAZE) are designed in this thesis with different objectives like S-HAZE is designed to evaluate the performance efficiencies of daytime image dehazing methods and how accurately the methods can restore the original scene and color characteristics of images (especially in sky regions of images) while N-HAZE is designed focusing exclusively on nighttime image dehazing. On the other hand, D&N-HAZE database comprises of both synthetic hazy and real hazy (captured in the presence of real atmospheric haze) daytime and nighttime images of similar scenes which help researchers to evaluate how the performance efficiencies of image dehazing methods vary according to the variations in nature of haze (real/synthetic) and illumination properties (daytime/nighttime) of images having similar scene contents.

The inclusion of GT images corresponding to each hazy image in these databases helps researchers to perform accurate subjective analyses as well as quantitative analyses using full-reference and no-reference quantitative parameters. The inclusion of dehazed outputs obtained from popular as well as recent image dehazing methods in these databases have also enhanced their acceptability to the researchers as these databases facilitate performing effective benchmarking of image dehazing methods.

In Chapter 7, detailed discussion is carried out regarding a novel Fuzzy Logic based sky segmentation method which is designed to automatically segment the sky regions and non-sky regions of daytime images based on the dissimilarities in their properties. This discussion is included under the 'Image Dehazing' part of the thesis because of the analogy between the two problems.

Chapter 7

The significance of sky regions of images in performing accurate atmospheric light estimation which in turn is required for dehazing images is discussed in details in previous Chapter. In this Chapter, the analogy between the image dehazing problem and the sky segmentation method is studied and a novel Fuzzy Logic based sky segmentation method is designed by exploiting the same logic used in image dehazing problem.

In this work, initially a database is created by collecting more than 150 clear, daytime images from multiple publicly available sources in the internet and then comprehensive experimental analyses are carried out considering multiple parameters from various color spaces like RGB color space, HSV color space and L^*a^*b color space to identify the parameters whose properties vary sharply between sky and non-sky regions of images.

The parameters from particularly these three color spaces are chosen to perform the experimental analyses in the sky-segmentation problem as these parameters have proved their excellence in performing accurate image dehazing in the existing works as well as in the novel image dehazing methods proposed in this thesis.

The logic used for the identification of severely degraded hazy pixels in hazy images can be used for the identification of the sky pixels in daytime images as both the problems aim at the identification of the pixels possessing comparatively higher intensities in images. The pixels possessing high haze content in an image are considered to be high intensity pixels as these pixels are under the maximum influence of airlight as stated in CAP proposed in [50]. Similarly, the pixels belonging to the sky regions of daytime images also possess high intensities because of the presence of sunlight at daytime. This similarity between the image dehazing problem and the sky segmentation problem have served as an inspiration behind the design of the proposed Fuzzy Logic based sky segmentation method utilizing similar logic used for performing image dehazing. RGB color space, HSV color space and L^*a^*b color space are considered to be ideal color spaces to perform identification of most significant parameters for performing sky segmentation as these color spaces have proved their efficiencies in performing image dehazing.

The parameters considered for performing experimental analyses are the three color channels of both the RGB and L^*a^*b color spaces, haze-concentration parameter and channel difference parameter.

The logic behind considering haze-concentration as a parameter for this problem is described in **Section 3.1.1.1 of Chapter 3**.

The channel difference parameter is the magnitude of the difference between the dark channel and the bright channel of images.

The dark channel of an image is computed using DCP [38] which is described in detail in **Section 2.2.1 of Chapter 2**. This channel as described in [38] mostly comprises of very low intensity pixels which belong to the non-sky regions of images. The main reason behind the presence of such low intensity pixels in estimated dark channel of an image is the presence of colourful objects like trees, leaves, etc. and their shadows in non-sky regions of images. The dark channel of an image gives highly accurate estimation of intensity distribution in non-sky regions (regions having comparatively low illumination) of images but often fails to give accurate estimation of intensity distribution in sky regions (regions having high illumination) of images.

To overcome this drawback, the authors in [92] have proposed Bright Channel Prior (BCP) which is simply the reverse of DCP. Unlike DCP, BCP gives accurate estimation of intensity distribution of sky regions of images.

These properties of the estimated dark and bright channels make them ideal to be considered for these type of analyses.

After performing experimental analyses, achromatic channel of L*a*b color space, all the three color channels (R,G and B) of RGB color space, haze-concentration and channel difference parameters are found to be most suitable parameters for performing sky segmentation.

The plots of these selected parameters' values for a sky patch and a non-sky patch are given as examples in Table 7.1-Table 7.2 to show their variations in sky and non-sky regions. The FI rules designed to perform sky-segmentation are given in Table 7.3.

Table 7.1. Selected parameters' values in a sky patch


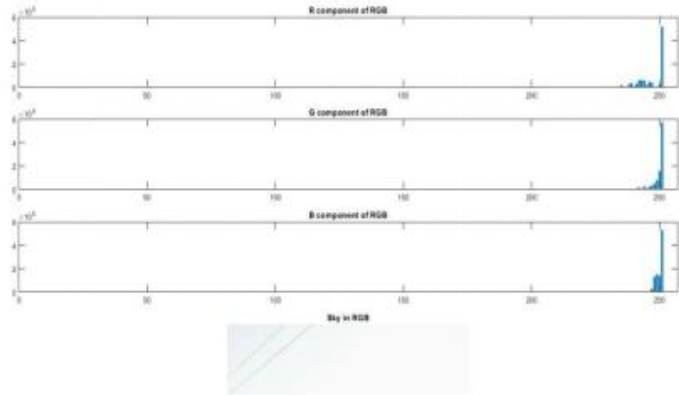
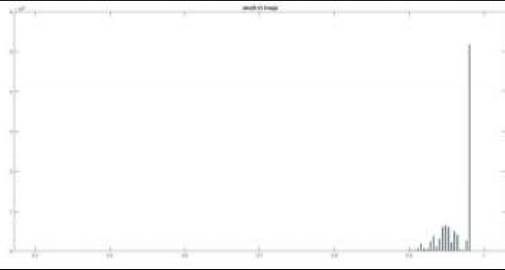
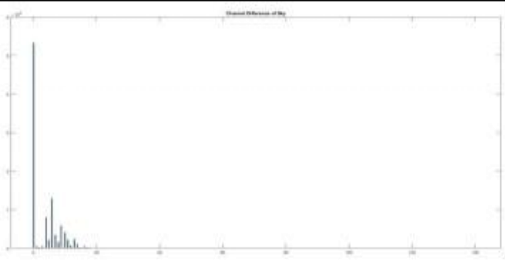
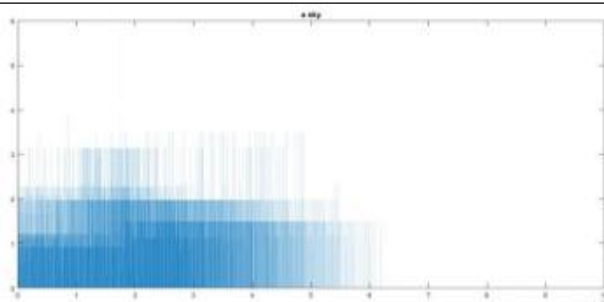
Type of patch	Patch image
Sky	
RGB Colour Space plot	
Magnitude of difference between Saturation and Value channels of HSV Colour Space	
Magnitude of difference between Bright and Dark channel values	
Achromatic channel values	

Table 7.2. Selected parameters' values in a non-sky patch


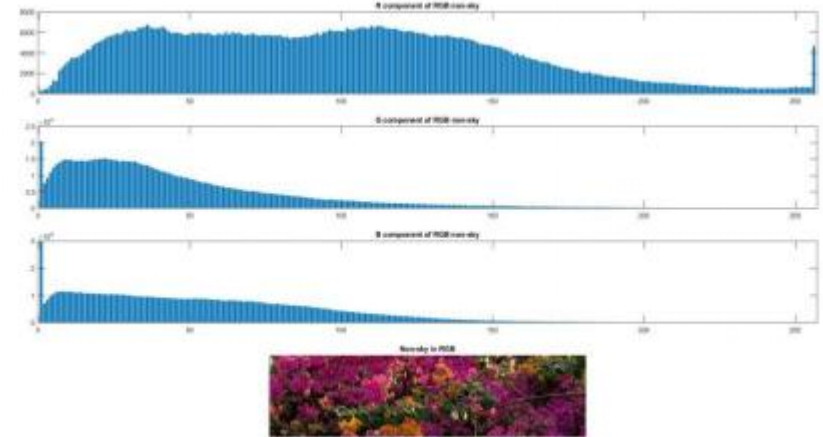
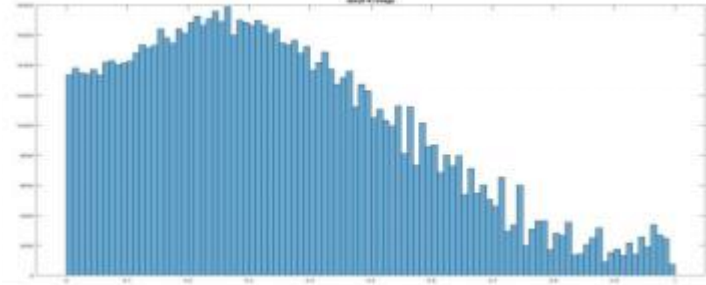
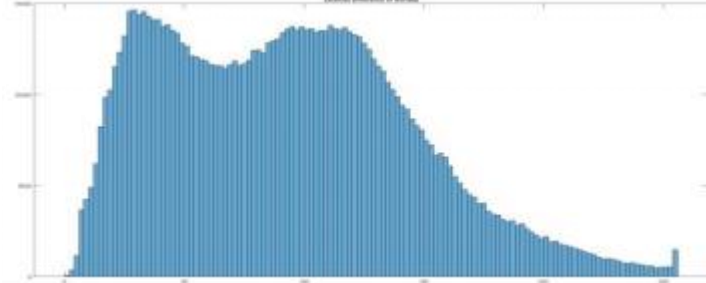
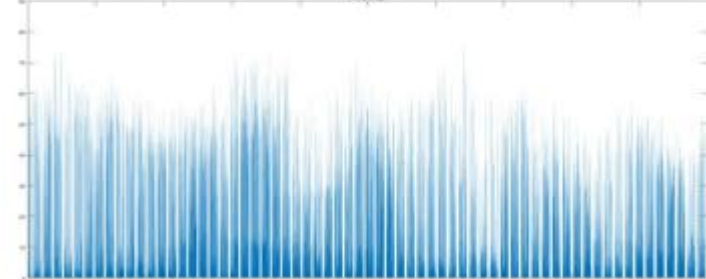
Type of patch	Patch image
Non-sky	
RGB Colour Space plot	
Magnitude of difference between Saturation and Value channels of HSV Colour Space	
Magnitude of difference between Bright and Dark channel values	
Achromatic channel values	

Table 7.3. Novel sets of FI rules designed to perform sky segmentation

If		Then	
achromatic_channel	haze_concentration	channel_difference	Class
Big_Negative	Very_Less_Dense	Very_Low_Difference	Non_Sky
Big_Negative	Very_Less_Dense	Low_Difference	Non_Sky
Big_Negative	Very_Less_Dense	Medium_Difference	Non_Sky
Big_Negative	Very_Less_Dense	High_Difference	Non_Sky
Big_Negative	Very_Less_Dense	Very_High_Difference	Non_Sky
Big_Negative	Less_Dense	Very_Low_Difference	Mostly_Non_Sky
Big_Negative	Less_Dense	Low_Difference	Mostly_Non_Sky
Big_Negative	Less_Dense	Medium_Difference	Non_Sky
Big_Negative	Less_Dense	High_Difference	Non_Sky
Big_Negative	Less_Dense	Very_High_Difference	Non_Sky
Big_Negative	Medium_Dense	Very_Low_Difference	More_or_Less_Sky
Big_Negative	Medium_Dense	Low_Difference	Mostly_Non_Sky
Big_Negative	Medium_Dense	Medium_Difference	Mostly_Non_Sky
Big_Negative	Medium_Dense	High_Difference	Non_Sky
Big_Negative	Medium_Dense	Very_High_Difference	Non_Sky
Big_Negative	Dense	Very_Low_Difference	Almost_Sky
Big_Negative	Dense	Low_Difference	More_or_Less_Sky
Big_Negative	Dense	Medium_Difference	Mostly_Non_Sky
Big_Negative	Dense	High_Difference	Mostly_Non_Sky
Big_Negative	Dense	Very_High_Difference	Non_Sky
Big_Negative	Very_High_Dense	Very_Low_Difference	Sky
Big_Negative	Very_High_Dense	Low_Difference	Sky
Big_Negative	Very_High_Dense	Medium_Difference	Almost_Sky
Big_Negative	Very_High_Dense	High_Difference	Almost_Sky
Big_Negative	Very_High_Dense	Very_High_Difference	More_or_Less_Sky
Negative	Very_Less_Dense	Very_Low_Difference	Non_Sky
Negative	Very_Less_Dense	Low_Difference	Non_Sky
Negative	Very_Less_Dense	Medium_Difference	Non_Sky
Negative	Very_Less_Dense	High_Difference	Non_Sky
Negative	Very_Less_Dense	Very_High_Difference	Non_Sky
Negative	Less_Dense	Very_Low_Difference	Mostly_Non_Sky
Negative	Less_Dense	Low_Difference	Mostly_Non_Sky
Negative	Less_Dense	Medium_Difference	Mostly_Non_Sky
Negative	Less_Dense	High_Difference	Mostly_Non_Sky
Negative	Less_Dense	Very_High_Difference	Mostly_Non_Sky
Negative	Medium_Dense	Very_Low_Difference	More_or_Less_Sky
Negative	Medium_Dense	Low_Difference	More_or_Less_Sky
Negative	Medium_Dense	Medium_Difference	Mostly_Non_Sky
Negative	Medium_Dense	High_Difference	Mostly_Non_Sky
Negative	Medium_Dense	Very_High_Difference	Non_Sky
Negative	Dense	Very_Low_Difference	Almost_Sky
Negative	Dense	Low_Difference	Almost_Sky
Negative	Dense	Medium_Difference	More_or_Less_Sky
Negative	Dense	High_Difference	Mostly_Non_Sky
Negative	Dense	Very_High_Difference	Non_Sky
Negative	Very_High_Dense	Very_Low_Difference	Sky
Negative	Very_High_Dense	Low_Difference	Almost_Sky
Negative	Very_High_Dense	Medium_Difference	More_or_Less_Sky
Negative	Very_High_Dense	High_Difference	Mostly_Non_Sky
Negative	Very_High_Dense	Very_High_Difference	Non_Sky
Zero	Very_Less_Dense	Very_Low_Difference	Mostly_Non_Sky
Zero	Very_Less_Dense	Low_Difference	Mostly_Non_Sky
Zero	Very_Less_Dense	Medium_Difference	Mostly_Non_Sky
Zero	Very_Less_Dense	High_Difference	Non_Sky
Zero	Very_Less_Dense	Very_High_Difference	Non_Sky
Zero	Less_Dense	Very_Low_Difference	Mostly_Non_Sky
Zero	Less_Dense	Low_Difference	Mostly_Non_Sky
Zero	Less_Dense	Medium_Difference	Mostly_Non_Sky
Zero	Less_Dense	High_Difference	Non_Sky
Zero	Less_Dense	Very_High_Difference	Non_Sky
Zero	Medium_Dense	Very_Low_Difference	More_or_Less_Sky
Zero	Medium_Dense	Low_Difference	Mostly_Non_Sky
Zero	Medium_Dense	Medium_Difference	Mostly_Non_Sky
Zero	Medium_Dense	High_Difference	Non_Sky
Zero	Medium_Dense	Very_High_Difference	Non_Sky
Zero	Dense	Very_Low_Difference	Almost_Sky
Zero	Dense	Low_Difference	More_or_Less_Sky
Zero	Dense	Medium_Difference	Mostly_Non_Sky

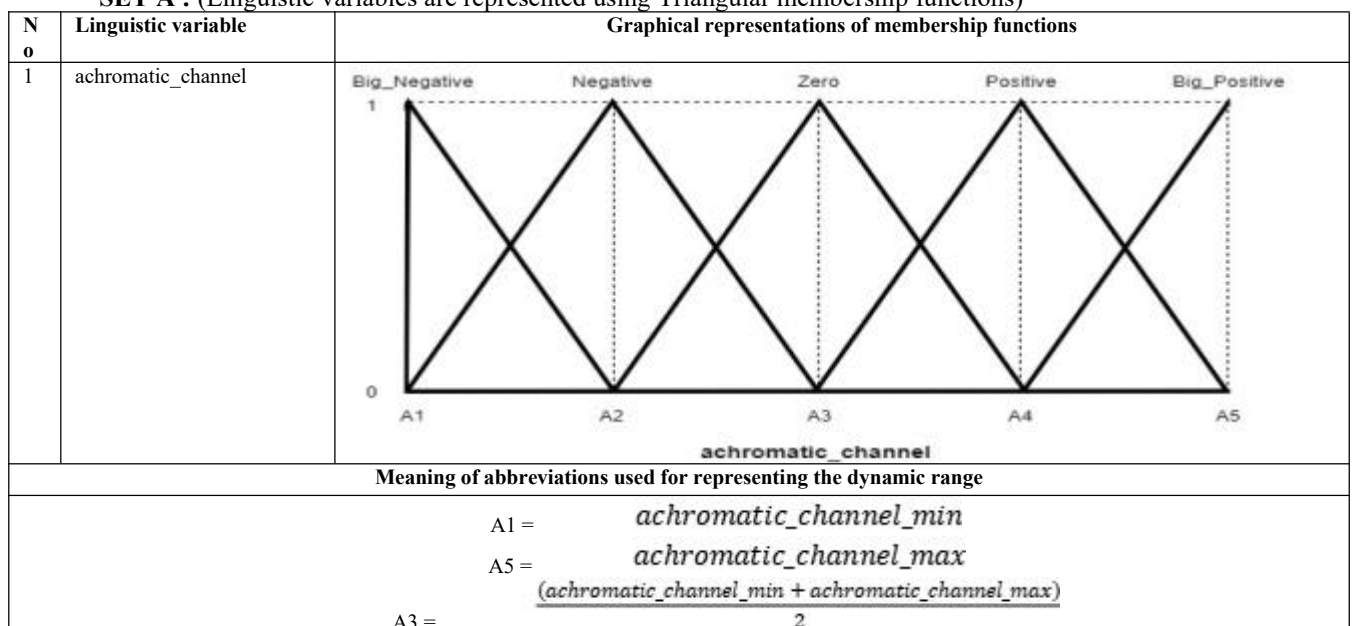
Zero	Dense	High_Difference	Mostly_Non_Sky
Zero	Dense	Very_High_Difference	Non_Sky
Zero	Very_High_Dense	Very_Low_Difference	Almost_Sky
Zero	Very_High_Dense	Low_Difference	Almost_Sky
Zero	Very_High_Dense	Medium_Difference	More_or_Less_Sky
Zero	Very_High_Dense	High_Difference	Mostly_Non_Sky
Zero	Very_High_Dense	Very_High_Difference	Mostly_Non_Sky
Positive	Very_Less_Dense	Very_Low_Difference	Mostly_Non_Sky
Positive	Very_Less_Dense	Low_Difference	Mostly_Non_Sky
Positive	Very_Less_Dense	Medium_Difference	Non_Sky
Positive	Very_Less_Dense	High_Difference	Non_Sky
Positive	Very_Less_Dense	Very_High_Difference	Non_Sky
Positive	Less_Dense	Very_Low_Difference	Mostly_Non_Sky
Positive	Less_Dense	Low_Difference	Mostly_Non_Sky
Positive	Less_Dense	Medium_Difference	Non_Sky
Positive	Less_Dense	High_Difference	Non_Sky
Positive	Less_Dense	Very_High_Difference	Non_Sky
Positive	Medium_Dense	Very_Low_Difference	More_or_Less_Sky
Positive	Medium_Dense	Low_Difference	Mostly_Non_Sky
Positive	Medium_Dense	Medium_Difference	Non_Sky
Positive	Medium_Dense	High_Difference	Non_Sky
Positive	Medium_Dense	Very_High_Difference	Non_Sky
Positive	Dense	Very_Low_Difference	Almost_Sky
Positive	Dense	Low_Difference	More_or_Less_Sky
Positive	Dense	Medium_Difference	Mostly_Non_Sky
Positive	Dense	High_Difference	Non_Sky
Positive	Dense	Very_High_Difference	Non_Sky
Positive	Very_High_Dense	Very_Low_Difference	Almost_Sky
Positive	Very_High_Dense	Low_Difference	Almost_Sky
Positive	Very_High_Dense	Medium_Difference	More_or_Less_Sky
Positive	Very_High_Dense	High_Difference	Mostly_Non_Sky
Positive	Very_High_Dense	Very_High_Difference	Non_Sky
Big_Positive	Very_Less_Dense	Very_Low_Difference	Mostly_Non_Sky
Big_Positive	Very_Less_Dense	Low_Difference	Mostly_Non_Sky
Big_Positive	Very_Less_Dense	Medium_Difference	Non_Sky
Big_Positive	Very_Less_Dense	High_Difference	Non_Sky
Big_Positive	Very_Less_Dense	Very_High_Difference	Non_Sky
Big_Positive	Less_Dense	Very_Low_Difference	Mostly_Non_Sky
Big_Positive	Less_Dense	Low_Difference	Mostly_Non_Sky
Big_Positive	Less_Dense	Medium_Difference	Non_Sky
Big_Positive	Less_Dense	High_Difference	Non_Sky
Big_Positive	Less_Dense	Very_High_Difference	Non_Sky
Big_Positive	Medium_Dense	Very_Low_Difference	More_or_Less_Sky
Big_Positive	Medium_Dense	Low_Difference	Mostly_Non_Sky
Big_Positive	Medium_Dense	Medium_Difference	Non_Sky
Big_Positive	Medium_Dense	High_Difference	Non_Sky
Big_Positive	Medium_Dense	Very_High_Difference	Non_Sky
Big_Positive	Dense	Very_Low_Difference	Almost_Sky
Big_Positive	Dense	Low_Difference	More_or_Less_Sky
Big_Positive	Dense	Medium_Difference	Mostly_Non_Sky
Big_Positive	Dense	High_Difference	Mostly_Non_Sky
Big_Positive	Dense	Very_High_Difference	Non_Sky
Big_Positive	Very_High_Dense	Very_Low_Difference	Almost_Sky
Big_Positive	Very_High_Dense	Low_Difference	Almost_Sky
Big_Positive	Very_High_Dense	Medium_Difference	More_or_Less_Sky
Big_Positive	Very_High_Dense	High_Difference	Mostly_Non_Sky
Big_Positive	Very_High_Dense	Very_High_Difference	Mostly_Non_Sky
red_channel	green_channel	blue_channel	Class
Very_Less_Red	Very_Less_Green	Very_Less_Blue	Non_Sky
Very_Less_Red	Very_Less_Green	Less_Blue	Non_Sky
Very_Less_Red	Very_Less_Green	Medium_Blue	More_or_Less_Sky
Very_Less_Red	Very_Less_Green	High_Blue	Sky
Very_Less_Red	Very_Less_Green	Very_High_Blue	Sky
Very_Less_Red	Less_Green	Very_Less_Blue	Non_Sky
Very_Less_Red	Less_Green	Less_Blue	Non_Sky
Very_Less_Red	Less_Green	Medium_Blue	More_or_Less_Sky
Very_Less_Red	Less_Green	High_Blue	Sky
Very_Less_Red	Less_Green	Very_High_Blue	Sky
Very_Less_Red	Medium_Green	Very_Less_Blue	Non_Sky

Very_Less_Red	Medium_Green	Less_Blue	Non_Sky
Very_Less_Red	Medium_Green	Medium_Blue	Non_Sky
Very_Less_Red	Medium_Green	High_Blue	Mostly_Nonsky
Very_Less_Red	Medium_Green	Very_High_Blue	More_or_Less_Sky
Very_Less_Red	High_Green	Very_Less_Blue	Non_Sky
Very_Less_Red	High_Green	Less_Blue	Non_Sky
Very_Less_Red	High_Green	Medium_Blue	Non_Sky
Very_Less_Red	High_Green	High_Blue	Mostly_Nonsky
Very_Less_Red	High_Green	Very_High_Blue	Mostly_Nonsky
Very_Less_Red	Very_High_Green	Very_Less_Blue	Non_Sky
Very_Less_Red	Very_High_Green	Less_Blue	Non_Sky
Very_Less_Red	Very_High_Green	Medium_Blue	Non_Sky
Very_Less_Red	Very_High_Green	High_Blue	Mostly_Nonsky
Very_Less_Red	Very_High_Green	Very_High_Blue	Mostly_Nonsky
Less_Red	Very_Less_Green	Very_Less_Blue	Non_Sky
Less_Red	Very_Less_Green	Less_Blue	Non_Sky
Less_Red	Very_Less_Green	Medium_Blue	More_or_Less_Sky
Less_Red	Very_Less_Green	High_Blue	Almost_Sky
Less_Red	Very_Less_Green	Very_High_Blue	Sky
Less_Red	Less_Green	Very_Less_Blue	Non_Sky
Less_Red	Less_Green	Less_Blue	Non_Sky
Less_Red	Less_Green	Medium_Blue	More_or_Less_Sky
Less_Red	Less_Green	High_Blue	Almost_Sky
Less_Red	Less_Green	Very_High_Blue	Sky
Less_Red	Medium_Green	Very_Less_Blue	Non_Sky
Less_Red	Medium_Green	Less_Blue	Non_Sky
Less_Red	Medium_Green	Medium_Blue	Mostly_Nonsky
Less_Red	Medium_Green	High_Blue	More_or_Less_Sky
Less_Red	Medium_Green	Very_High_Blue	More_or_Less_Sky
Less_Red	High_Green	Very_Less_Blue	Non_Sky
Less_Red	High_Green	Less_Blue	Non_Sky
Less_Red	High_Green	Medium_Blue	Non_Sky
Less_Red	High_Green	High_Blue	Mostly_Nonsky
Less_Red	High_Green	Very_High_Blue	Mostly_Nonsky
Less_Red	Very_High_Green	Very_Less_Blue	Non_Sky
Less_Red	Very_High_Green	Less_Blue	Non_Sky
Less_Red	Very_High_Green	Medium_Blue	Non_Sky
Less_Red	Very_High_Green	High_Blue	Non_Sky
Less_Red	Very_High_Green	Very_High_Blue	Non_Sky
Medium_Red	Very_Less_Green	Very_Less_Blue	Non_Sky
Medium_Red	Very_Less_Green	Less_Blue	Non_Sky
Medium_Red	Very_Less_Green	Medium_Blue	Mostly_Nonsky
Medium_Red	Very_Less_Green	High_Blue	Mostly_Nonsky
Medium_Red	Very_Less_Green	Very_High_Blue	Almost_Sky
Medium_Red	Less_Green	Very_Less_Blue	Non_Sky
Medium_Red	Less_Green	Less_Blue	Non_Sky
Medium_Red	Less_Green	Medium_Blue	Mostly_Nonsky
Medium_Red	Less_Green	High_Blue	Mostly_Nonsky
Medium_Red	Less_Green	Very_High_Blue	More_or_Less_Sky
Medium_Red	Medium_Green	Very_Less_Blue	Non_Sky
Medium_Red	Medium_Green	Less_Blue	Non_Sky
Medium_Red	Medium_Green	Medium_Blue	Mostly_Nonsky
Medium_Red	Medium_Green	High_Blue	Mostly_Nonsky
Medium_Red	Medium_Green	Very_High_Blue	More_or_Less_Sky
Medium_Red	High_Green	Very_Less_Blue	Non_Sky
Medium_Red	High_Green	Less_Blue	Non_Sky
Medium_Red	High_Green	Medium_Blue	Non_Sky
Medium_Red	High_Green	High_Blue	Mostly_Nonsky
Medium_Red	High_Green	Very_High_Blue	Mostly_Nonsky
Medium_Red	Very_High_Green	Very_Less_Blue	Non_Sky
Medium_Red	Very_High_Green	Less_Blue	Non_Sky
Medium_Red	Very_High_Green	Medium_Blue	Non_Sky
Medium_Red	Very_High_Green	High_Blue	Mostly_Nonsky
Medium_Red	Very_High_Green	Very_High_Blue	Mostly_Nonsky
High_Red	Very_Less_Green	Very_Less_Blue	Nonsky
High_Red	Very_Less_Green	Less_Blue	Nonsky
High_Red	Very_Less_Green	Medium_Blue	Nonsky
High_Red	Very_Less_Green	High_Blue	Mostly_Nonsky
High_Red	Very_Less_Green	Very_High_Blue	Mostly_Nonsky
High_Red	Less_Green	Very_Less_Blue	Nonsky
High_Red	Less_Green	Less_Blue	Nonsky
High_Red	Less_Green	Medium_Blue	Nonsky

High_Red	Less_Green	High_Blue	Mostly Nonsky
High_Red	Less_Green	Very_High_Blue	Mostly Nonsky
High_Red	Medium_Green	Very_Less_Blue	Nonsky
High_Red	Medium_Green	Less_Blue	Nonsky
High_Red	Medium_Green	Medium_Blue	More_or_Less_Sky
High_Red	Medium_Green	High_Blue	Almost_Sky
High_Red	Medium_Green	Very_High_Blue	Almost_Sky
High_Red	High_Green	Very_Less_Blue	Nonsky
High_Red	High_Green	Less_Blue	Mostly Nonsky
High_Red	High_Green	Medium_Blue	More_or_Less_Sky
High_Red	High_Green	High_Blue	Almost_Sky
High_Red	High_Green	Very_High_Blue	Sky
High_Red	Very_High_Green	Very_Less_Blue	Nonsky
High_Red	Very_High_Green	Less_Blue	Mostly Nonsky
High_Red	Very_High_Green	Medium_Blue	More_or_Less_Sky
High_Red	Very_High_Green	High_Blue	Almost_Sky
High_Red	Very_High_Green	Very_High_Blue	Sky
Very_High_Red	Very_Less_Green	Very_Less_Blue	Nonsky
Very_High_Red	Very_Less_Green	Less_Blue	Nonsky
Very_High_Red	Very_Less_Green	Medium_Blue	Nonsky
Very_High_Red	Very_Less_Green	High_Blue	Mostly Nonsky
Very_High_Red	Very_Less_Green	Very_High_Blue	Mostly Nonsky
Very_High_Red	Less_Green	Very_Less_Blue	Nonsky
Very_High_Red	Less_Green	Less_Blue	Nonsky
Very_High_Red	Less_Green	Medium_Blue	Nonsky
Very_High_Red	Less_Green	High_Blue	Mostly Nonsky
Very_High_Red	Less_Green	Very_High_Blue	Mostly Nonsky
Very_High_Red	Medium_Green	Very_Less_Blue	Nonsky
Very_High_Red	Medium_Green	Less_Blue	Mostly Nonsky
Very_High_Red	Medium_Green	Medium_Blue	More_or_Less_Sky
Very_High_Red	Medium_Green	High_Blue	Almost_Sky
Very_High_Red	Medium_Green	Very_High_Blue	Almost_Sky
Very_High_Red	High_Green	Very_Less_Blue	Nonsky
Very_High_Red	High_Green	Less_Blue	Nonsky
Very_High_Red	High_Green	Medium_Blue	More_or_Less_Sky
Very_High_Red	High_Green	High_Blue	Sky
Very_High_Red	High_Green	Very_High_Blue	Sky
Very_High_Red	Very_High_Green	Very_Less_Blue	Nonsky
Very_High_Red	Very_High_Green	Less_Blue	Nonsky
Very_High_Red	Very_High_Green	Medium_Blue	Sky
Very_High_Red	Very_High_Green	High_Blue	Sky
Very_High_Red	Very_High_Green	Very_High_Blue	Sky

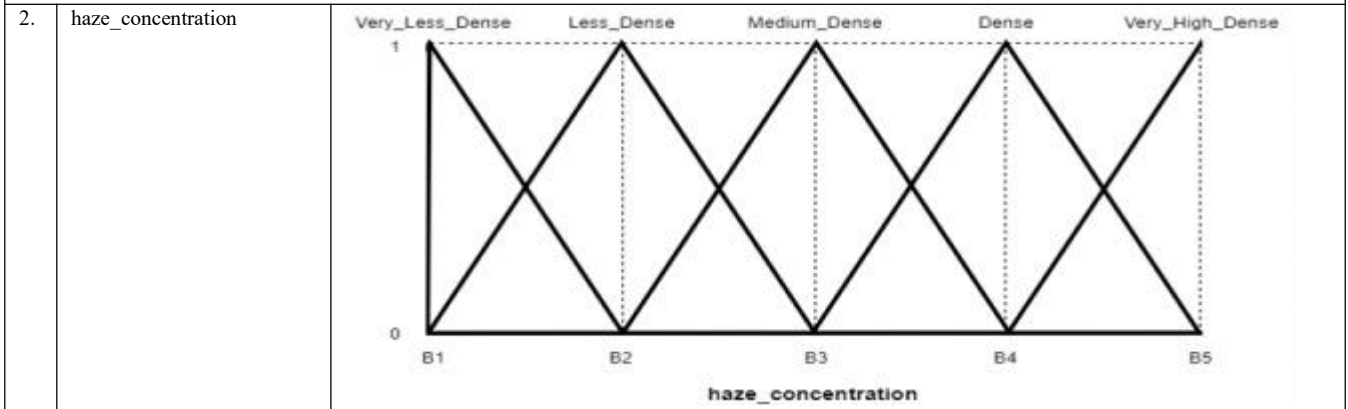
Table 7.4. Graphical representations of membership functions used to represent linguistic variables

SET A : (Linguistic variables are represented using Triangular membership functions)



$$A2 = \left(\frac{\frac{achromatic_channel_min + achromatic_channel_max}{2} + achromatic_channel_min}{2} \right)$$

$$A4 = \left(\frac{\frac{achromatic_channel_min + achromatic_channel_max}{2} + achromatic_channel_max}{2} \right)$$



Meaning of abbreviations used for representing the dynamic range

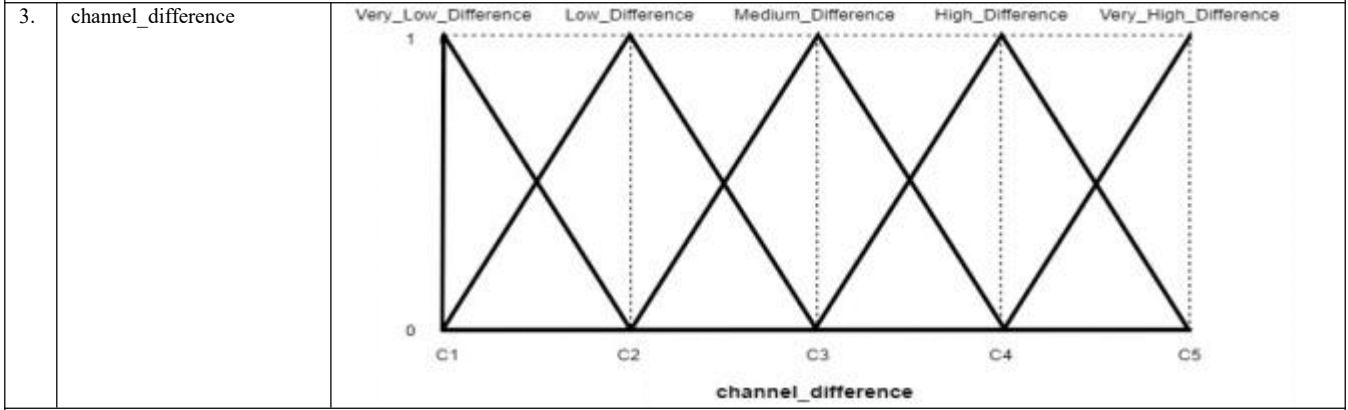
$$B1 = haze_concentration_min$$

$$B5 = haze_concentration_max$$

$$B3 = \frac{haze_concentration_min + haze_concentration_max}{2}$$

$$B2 = \left(\frac{\frac{haze_concentration_min + haze_concentration_max}{2} + haze_concentration_min}{2} \right)$$

$$B4 = \left(\frac{\frac{haze_concentration_min + haze_concentration_max}{2} + haze_concentration_max}{2} \right)$$



Meaning of abbreviations used for representing the dynamic range

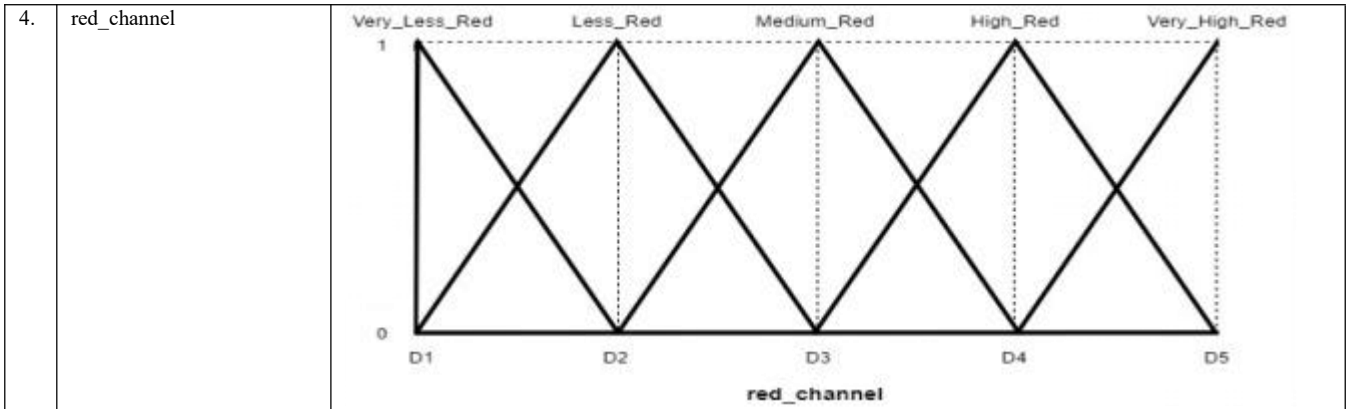
$$C1 = channel_difference_min$$

$$C5 = channel_difference_max$$

$$C3 = \frac{channel_difference_min + channel_difference_max}{2}$$

$$C2 = \left(\frac{\frac{channel_difference_min + channel_difference_max}{2} + channel_difference_min}{2} \right)$$

$$C4 = \left(\frac{\frac{channel_difference_min + channel_difference_max}{2} + channel_difference_max}{2} \right)$$



Meaning of abbreviations used for representing the dynamic range

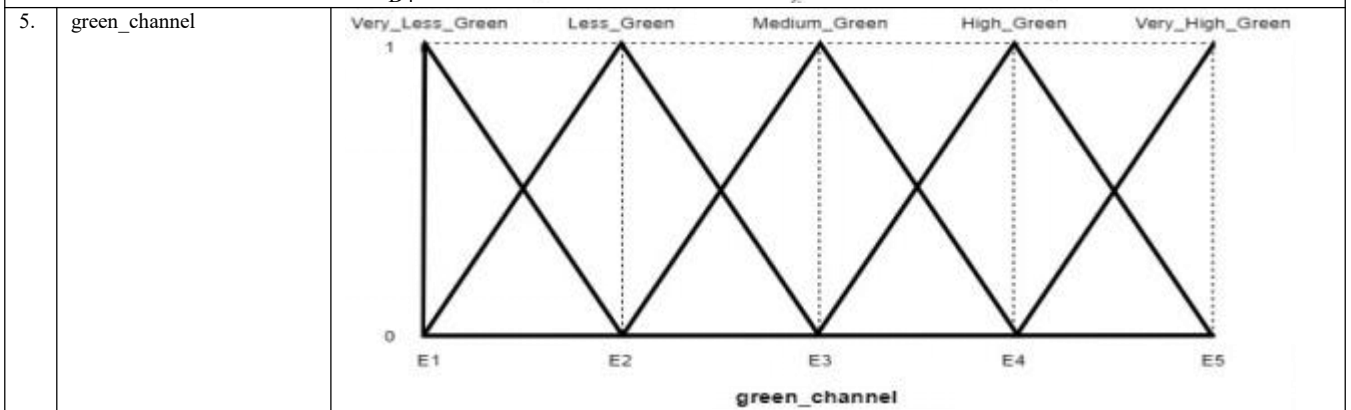
$$D1 = \text{red_channel_min}$$

$$D5 = \text{red_channel_max}$$

$$D3 = \frac{(\text{red_channel_min} + \text{red_channel_max})}{2}$$

$$D2 = \left(\frac{(\frac{(\text{red_channel_min} + \text{red_channel_max})}{2}) + \text{red_channel_min}}{2} \right)$$

$$D4 = \left(\frac{(\frac{(\text{red_channel_min} + \text{red_channel_max})}{2}) + \text{red_channel_max}}{2} \right)$$



Meaning of abbreviations used for representing the dynamic range

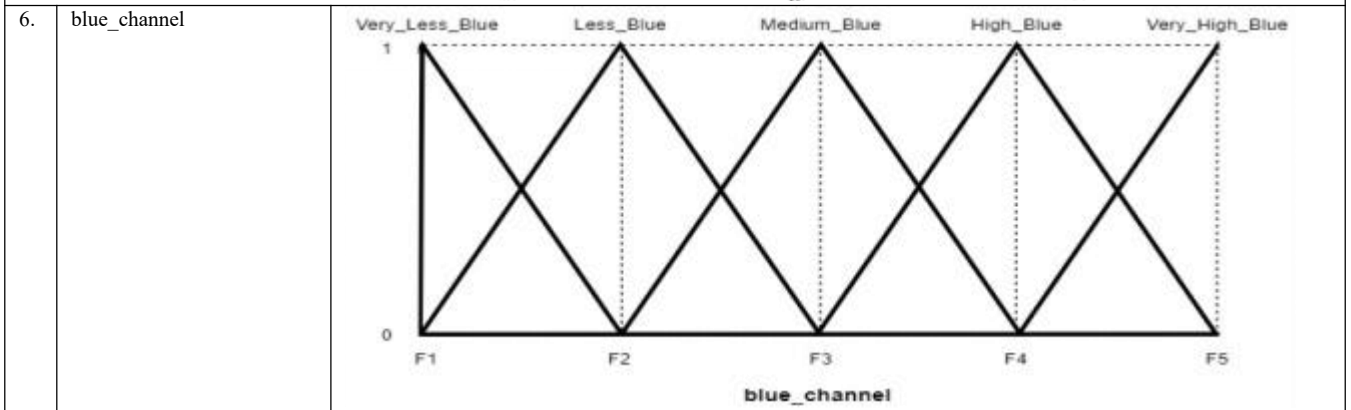
$$E1 = \text{green_channel_min}$$

$$E5 = \text{green_channel_max}$$

$$E3 = \frac{(\text{green_channel_min} + \text{green_channel_max})}{2}$$

$$E2 = \left(\frac{(\frac{(\text{green_channel_min} + \text{green_channel_max})}{2}) + \text{green_channel_min}}{2} \right)$$

$$E4 = \left(\frac{(\frac{(\text{green_channel_min} + \text{green_channel_max})}{2}) + \text{green_channel_max}}{2} \right)$$



Meaning of abbreviations used for representing the dynamic range

		$F1 = \text{blue_channel_min}$ $F5 = \text{blue_channel_max}$ $F3 = \frac{(\text{blue_channel_min} + \text{blue_channel_max}) + \text{blue_channel_min}}{2}$ $F2 = \frac{(\frac{(\text{blue_channel_min} + \text{blue_channel_max})}{2} + \text{blue_channel_min})}{2}$ $F4 = \frac{(\frac{(\text{blue_channel_min} + \text{blue_channel_max})}{2} + \text{blue_channel_max})}{2}$
7	class	
Meaning of abbreviations used for representing the dynamic range		
		$G1 = \text{class_min} = 0$ $G5 = \text{class_max} = 1$ $G3 = \frac{(\text{class_min} + \text{class_max}) + \text{class_min}}{2}$ $G2 = \frac{(\frac{(\text{class_min} + \text{class_max})}{2} + \text{class_min})}{2}$ $G4 = \frac{(\frac{(\text{class_min} + \text{class_max})}{2} + \text{class_max})}{2}$

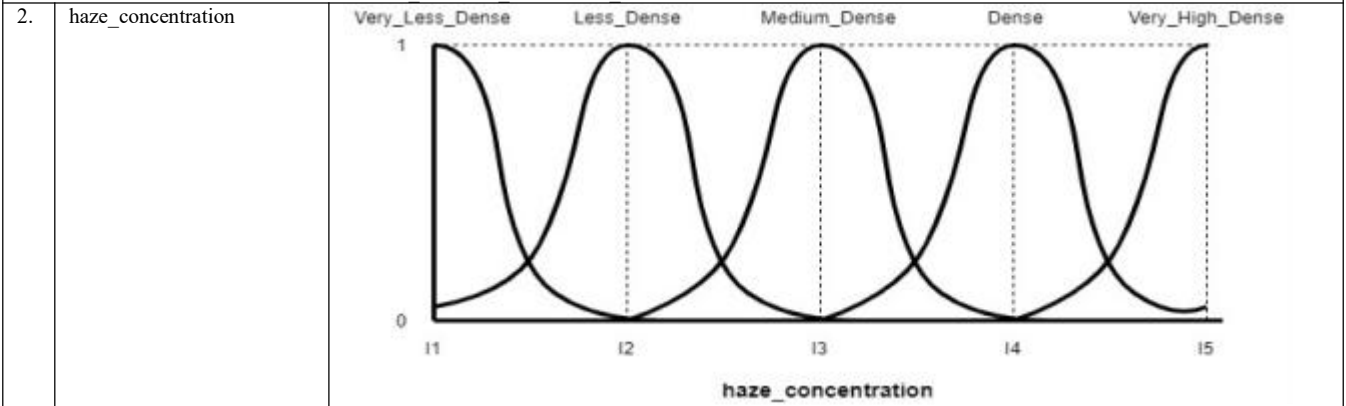
SET B : (Linguistic variables are represented using Gaussian membership functions)

N o	Linguistic variable	Graphical representations of membership functions
1	achromatic_channel	
Meaning of abbreviations used for representing the dynamic range		
		$H1 = \text{achromatic_channel_min}$ $H5 = \text{achromatic_channel_max}$ $H3 = \frac{(\text{achromatic_channel_min} + \text{achromatic_channel_max}) + \text{achromatic_channel_min}}{2}$ $H2 = \frac{(\frac{(\text{achromatic_channel_min} + \text{achromatic_channel_max})}{2} + \text{achromatic_channel_min})}{2}$

$$H4 = \frac{\left(\frac{achromatic_channel_min + achromatic_channel_max}{2}\right) + achromatic_channel_max}{2}$$

$$achromatic_channel_range = \frac{achromatic_channel_max - achromatic_channel_min}{0.5 * achromatic_channel_range}$$

$$standard\ deviation\ achromatic\ channel = \frac{\sqrt{-2 * \log(0.5)} * 4}{}$$



Meaning of abbreviations used for representing the dynamic range

$$I1 = haze_concentration_min$$

$$I5 = haze_concentration_max$$

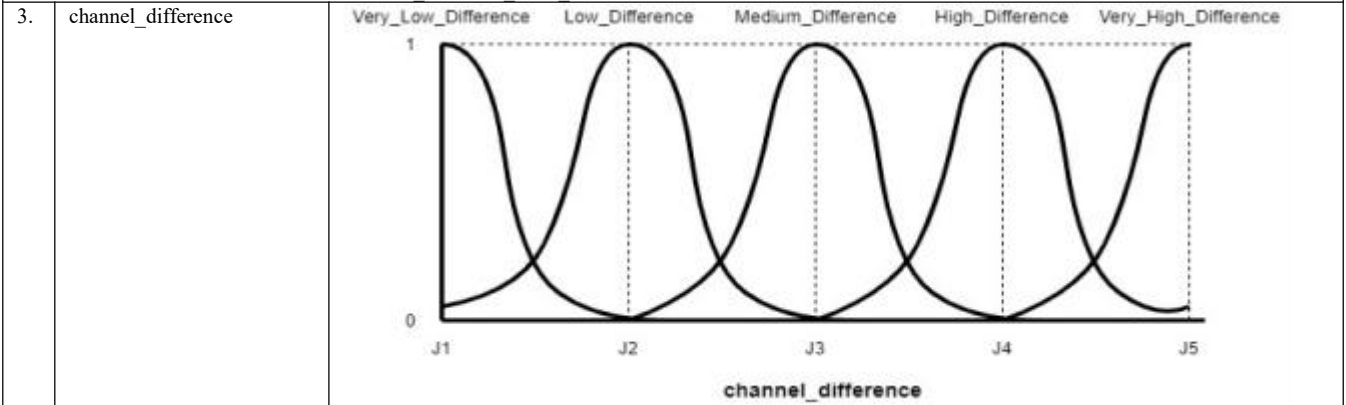
$$I3 = \frac{(haze_concentration_min + haze_concentration_max)}{2} + haze_concentration_min$$

$$I2 = \frac{\left(\frac{haze_concentration_min + haze_concentration_max}{2}\right) + haze_concentration_min}{2}$$

$$I4 = \frac{\left(\frac{haze_concentration_min + haze_concentration_max}{2}\right) + haze_concentration_max}{2}$$

$$haze_concentration_range = \frac{haze_concentration_max - haze_concentration_min}{0.5 * haze_concentration_range}$$

$$standard\ deviation\ haze\ concentration = \frac{\sqrt{-2 * \log(0.5)} * 4}{}$$



Meaning of abbreviations used for representing the dynamic range

$$J1 = channel_difference_min$$

$$J5 = channel_difference_max$$

$$J3 = \frac{(channel_difference_min + channel_difference_max)}{2} + channel_difference_min$$

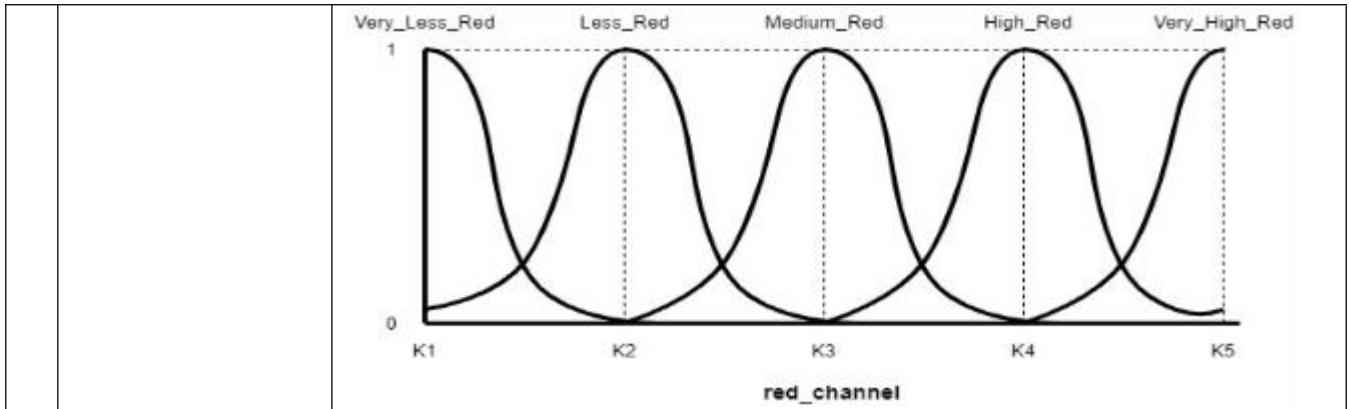
$$J2 = \frac{\left(\frac{channel_difference_min + channel_difference_max}{2}\right) + channel_difference_min}{2}$$

$$J4 = \frac{\left(\frac{channel_difference_min + channel_difference_max}{2}\right) + channel_difference_max}{2}$$

$$channel_difference_range = \frac{channel_difference_max - channel_difference_min}{0.5 * channel_difference_range}$$

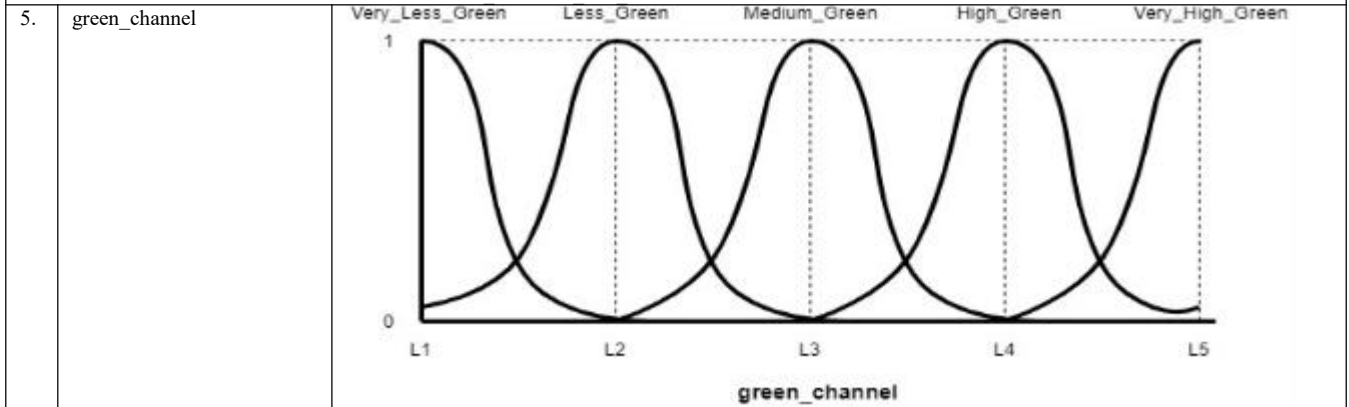
$$standard\ deviation\ channel\ difference = \frac{\sqrt{-2 * \log(0.5)} * 4}{}$$

4.	red channel	
----	-------------	--



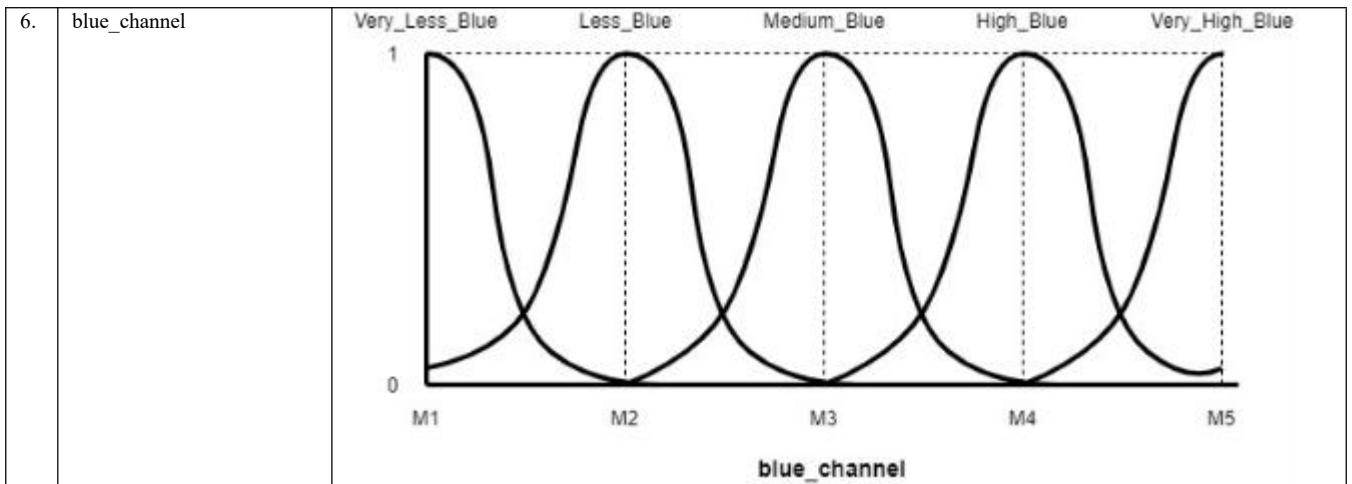
Meaning of abbreviations used for representing the dynamic range

$$\begin{aligned}
 K1 &= \text{red_channel_min} \\
 K5 &= \text{red_channel_max} \\
 K3 &= \frac{\text{red_channel_min} + \text{red_channel_max}}{2} \\
 K2 &= \left(\frac{\frac{\text{red_channel_min} + \text{red_channel_max}}{2} + \text{red_channel_min}}{2} \right) \\
 K4 &= \left(\frac{\frac{\text{red_channel_min} + \text{red_channel_max}}{2} + \text{red_channel_max}}{2} \right) \\
 \text{red_channel_range} &= \text{red_channel_max} - \text{red_channel_min} \\
 \text{standard deviation red channel} &= \frac{0.5 * \text{red_channel_range}}{\sqrt{-2 * \log(0.5)} * 4}
 \end{aligned}$$



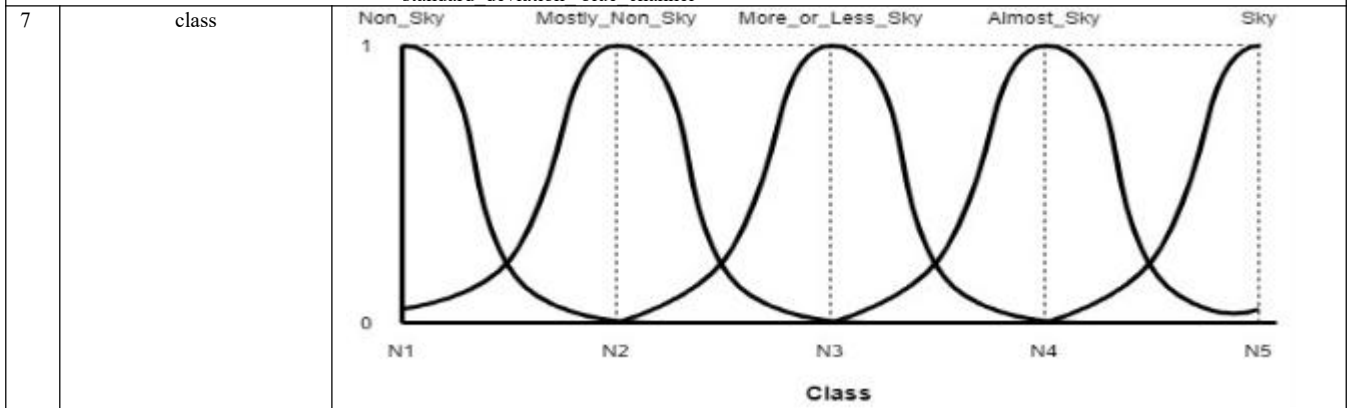
Meaning of abbreviations used for representing the dynamic range

$$\begin{aligned}
 L1 &= \text{green_channel_min} \\
 L5 &= \text{green_channel_max} \\
 L3 &= \frac{\text{green_channel_min} + \text{green_channel_max}}{2} \\
 L2 &= \left(\frac{\frac{\text{green_channel_min} + \text{green_channel_max}}{2} + \text{green_channel_min}}{2} \right) \\
 L4 &= \left(\frac{\frac{\text{green_channel_min} + \text{green_channel_max}}{2} + \text{green_channel_max}}{2} \right) \\
 \text{green_channel_range} &= \text{green_channel_max} - \text{green_channel_min} \\
 \text{standard deviation green channel} &= \frac{0.5 * \text{green_channel_range}}{\sqrt{-2 * \log(0.5)} * 4}
 \end{aligned}$$



Meaning of abbreviations used for representing the dynamic range

$$\begin{aligned}
 M1 &= \text{blue_channel_min} \\
 M5 &= \text{blue_channel_max} \\
 M3 &= \frac{(\text{blue_channel_min} + \text{blue_channel_max}) + \text{blue_channel_min}}{2} \\
 M2 &= \frac{(\frac{(\text{blue_channel_min} + \text{blue_channel_max}) + \text{blue_channel_min}}{2}) + \text{blue_channel_min}}{2} \\
 M4 &= \frac{(\frac{(\text{blue_channel_min} + \text{blue_channel_max}) + \text{blue_channel_max}}{2}) + \text{blue_channel_max}}{2} \\
 \text{blue_channel_range} &= \text{blue_channel_max} - \text{blue_channel_min} \\
 \text{standard deviation blue channel} &= \frac{0.5 * \text{blue_channel_range}}{\sqrt{-2 * \log(0.5)}} * 4
 \end{aligned}$$



Meaning of abbreviations used for representing the dynamic range

$$\begin{aligned}
 N1 &= \text{class_min} = 0 \\
 N5 &= \text{class_max} = 1 \\
 N3 &= \frac{(\text{class_min} + \text{class_max}) + \text{class_min}}{2} \\
 N2 &= \frac{(\frac{(\text{class_min} + \text{class_max}) + \text{class_min}}{2}) + \text{class_min}}{2} \\
 N4 &= \frac{(\frac{(\text{class_min} + \text{class_max}) + \text{class_max}}{2}) + \text{class_max}}{2} \\
 \text{class_range} &= \text{class_max} - \text{class_min} \\
 \text{standard deviation class} &= \frac{0.5 * \text{class_range}}{\sqrt{-2 * \log(0.5)}} * 4
 \end{aligned}$$

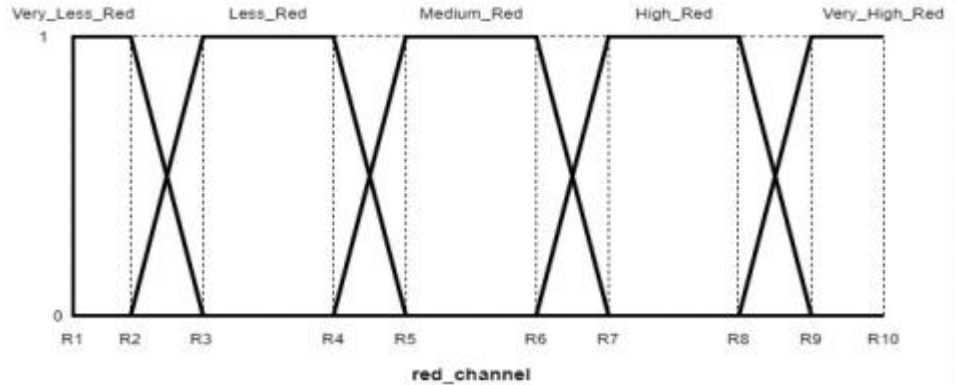
SET C : (Linguistic variables are represented using Trapezoidal membership functions)

No	Linguistic variable	Graphical representations of membership functions
1	achromatic_channel	
Meaning of abbreviations used for representing the dynamic range		
<p style="text-align: center;">O1 = achromatic_channel_min O10 = achromatic_channel_max</p> <p>achromatic_channel_range = achromatic_channel_max - achromatic_channel_min</p> <p>O2 = achromatic_channel_min+(achromatic_channel_range*0.1) O3 = achromatic_channel_min+(achromatic_channel_range*0.225) O4 = achromatic_channel_min+(achromatic_channel_range*0.325) O5 = achromatic_channel_min+(achromatic_channel_range*0.45) O6 = achromatic_channel_min+(achromatic_channel_range*0.55) O7 = achromatic_channel_min+(achromatic_channel_range*0.675) O8 = achromatic_channel_min+(achromatic_channel_range*0.775) O9 = achromatic_channel_min+(achromatic_channel_range*0.9)</p>		
2.	haze_concentration	
Meaning of abbreviations used for representing the dynamic range		
<p style="text-align: center;">P1 = haze_concentration_min P10 = haze_concentration_max</p> <p>haze_concentration_range = haze_concentration_max - haze_concentration_min</p> <p>P2 = haze_concentration_min+(haze_concentration_range*0.1) P3 = haze_concentration_min+(haze_concentration_range*0.225) P4 = haze_concentration_min+(haze_concentration_range*0.325) P5 = haze_concentration_min+(haze_concentration_range*0.45) P6 = haze_concentration_min+(haze_concentration_range*0.55) P7 = haze_concentration_min+(haze_concentration_range*0.675) P8 = haze_concentration_min+(haze_concentration_range*0.775) P9 = haze_concentration_min+(haze_concentration_range*0.9)</p>		
3.	channel_difference	

Meaning of abbreviations used for representing the dynamic range

$Q1 = \text{channel_difference_min}$
 $Q10 = \text{channel_difference_max}$
 $\text{channel_difference_range} = \text{channel_difference_max} - \text{channel_difference_min}$
 $Q2 = \text{channel_difference_min} + (\text{channel_difference_range} * 0.1)$
 $Q3 = \text{channel_difference_min} + (\text{channel_difference_range} * 0.225)$
 $Q4 = \text{channel_difference_min} + (\text{channel_difference_range} * 0.325)$
 $Q5 = \text{channel_difference_min} + (\text{channel_difference_range} * 0.45)$
 $Q6 = \text{channel_difference_min} + (\text{channel_difference_range} * 0.55)$
 $Q7 = \text{channel_difference_min} + (\text{channel_difference_range} * 0.675)$
 $Q8 = \text{channel_difference_min} + (\text{channel_difference_range} * 0.775)$
 $Q9 = \text{channel_difference_min} + (\text{channel_difference_range} * 0.9)$

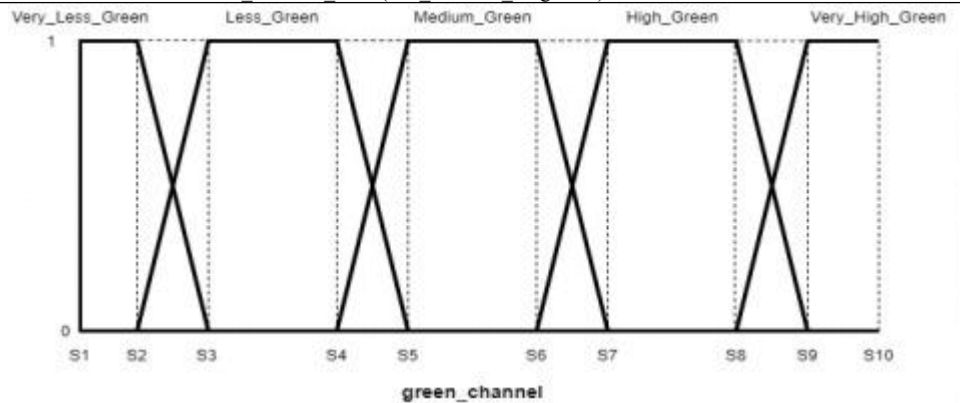
4. red_channel



Meaning of abbreviations used for representing the dynamic range

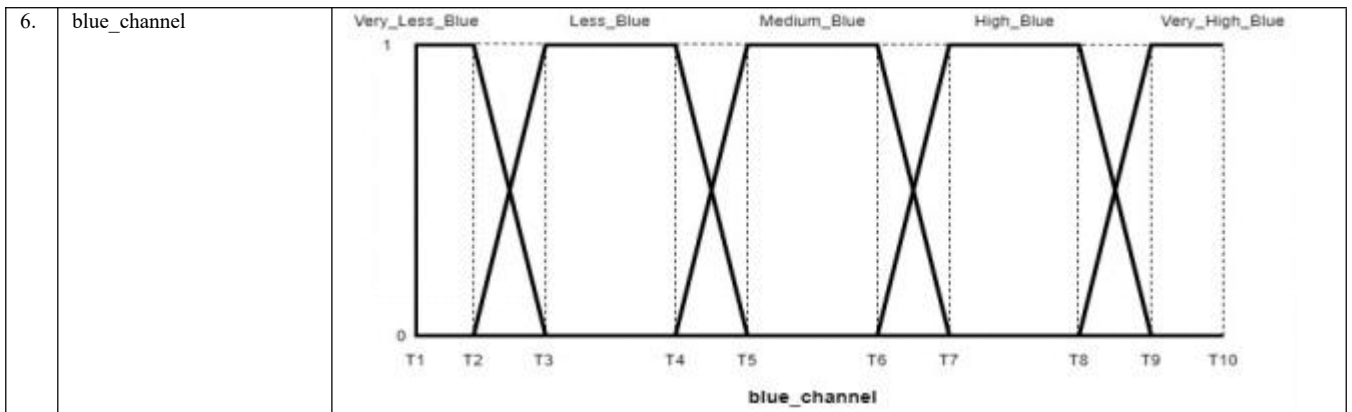
$R1 = \text{red_channel_min}$
 $R10 = \text{red_channel_max}$
 $\text{red_channel_range} = \text{red_channel_max} - \text{red_channel_min}$
 $R2 = \text{red_channel_min} + (\text{red_channel_range} * 0.1)$
 $R3 = \text{red_channel_min} + (\text{red_channel_range} * 0.225)$
 $R4 = \text{red_channel_min} + (\text{red_channel_range} * 0.325)$
 $R5 = \text{red_channel_min} + (\text{red_channel_range} * 0.45)$
 $R6 = \text{red_channel_min} + (\text{red_channel_range} * 0.55)$
 $R7 = \text{red_channel_min} + (\text{red_channel_range} * 0.675)$
 $R8 = \text{red_channel_min} + (\text{red_channel_range} * 0.775)$
 $R9 = \text{red_channel_min} + (\text{red_channel_range} * 0.9)$

5. green_channel



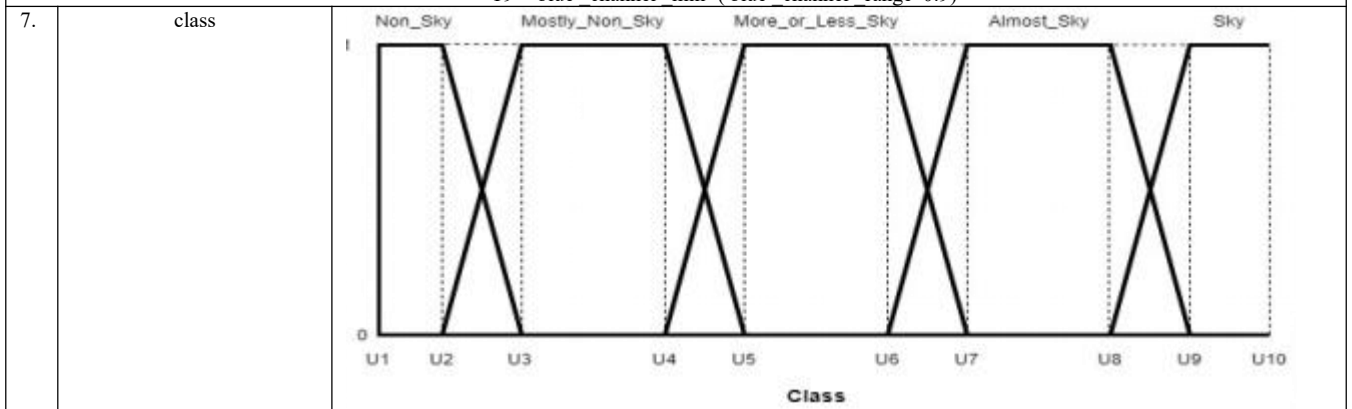
Meaning of abbreviations used for representing the dynamic range

$S1 = \text{green_channel_min}$
 $S10 = \text{green_channel_max}$
 $\text{green_channel_range} = \text{green_channel_max} - \text{green_channel_min}$
 $S2 = \text{green_channel_min} + (\text{green_channel_range} * 0.1)$
 $S3 = \text{green_channel_min} + (\text{green_channel_range} * 0.225)$
 $S4 = \text{green_channel_min} + (\text{green_channel_range} * 0.325)$
 $S5 = \text{green_channel_min} + (\text{green_channel_range} * 0.45)$
 $S6 = \text{green_channel_min} + (\text{green_channel_range} * 0.55)$
 $S7 = \text{green_channel_min} + (\text{green_channel_range} * 0.675)$
 $S8 = \text{green_channel_min} + (\text{green_channel_range} * 0.775)$
 $S9 = \text{green_channel_min} + (\text{green_channel_range} * 0.9)$



Meaning of abbreviations used for representing the dynamic range

$T1 = \text{blue_channel_min}$
 $T10 = \text{blue_channel_max}$
 $\text{blue_channel_range} = \text{blue_channel_max} - \text{blue_channel_min}$
 $T2 = \text{blue_channel_min} + (\text{blue_channel_range} * 0.1)$
 $T3 = \text{blue_channel_min} + (\text{blue_channel_range} * 0.225)$
 $T4 = \text{blue_channel_min} + (\text{blue_channel_range} * 0.325)$
 $T5 = \text{blue_channel_min} + (\text{blue_channel_range} * 0.45)$
 $T6 = \text{blue_channel_min} + (\text{blue_channel_range} * 0.55)$
 $T7 = \text{blue_channel_min} + (\text{blue_channel_range} * 0.675)$
 $T8 = \text{blue_channel_min} + (\text{blue_channel_range} * 0.775)$
 $T9 = \text{blue_channel_min} + (\text{blue_channel_range} * 0.9)$



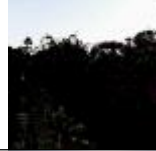

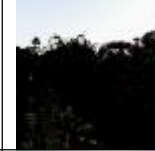

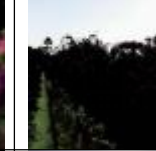


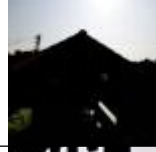

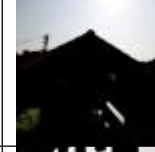

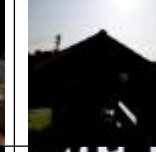


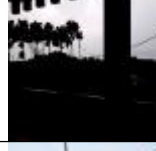

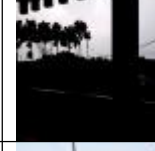

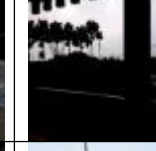


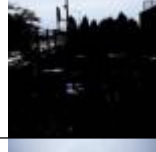




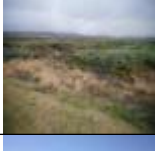

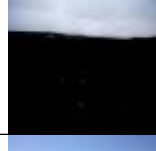
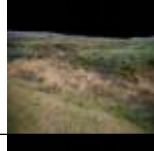
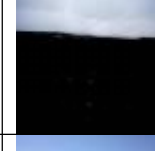

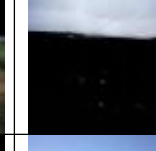


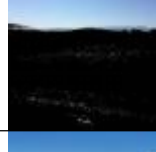



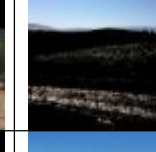


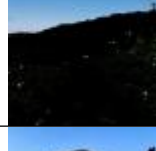








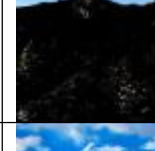

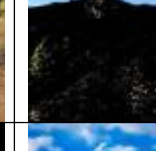

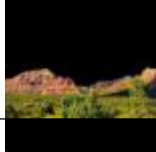
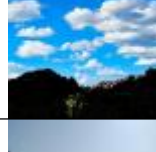
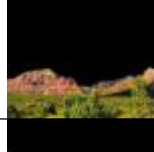


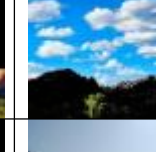






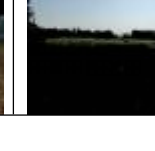


Meaning of abbreviations used for representing the dynamic range

$U1 = \text{class_min} = 0$
 $U10 = \text{class_max} = 1$
 $\text{class_range} = \text{class_max} - \text{class_min}$
 $U2 = \text{class_min} + (\text{class_range} * 0.1)$
 $U3 = \text{class_min} + (\text{class_range} * 0.225)$
 $U4 = \text{class_min} + (\text{class_range} * 0.325)$
 $U5 = \text{class_min} + (\text{class_range} * 0.45)$
 $U6 = \text{class_min} + (\text{class_range} * 0.55)$
 $U7 = \text{class_min} + (\text{class_range} * 0.675)$
 $U8 = \text{class_min} + (\text{class_range} * 0.775)$
 $U9 = \text{class_min} + (\text{class_range} * 0.9)$

Here the same experiment is repeated thrice using different membership functions to represent linguistic variables. Although Trapezoidal membership function is not an appropriate choice for previous methods as it gives almost constant outputs over a wide range but here it is also considered to note the results. Graphical representations of membership functions are given in tabular form in this Chapter for better understanding as these figures are represented using several variables which needs to be explained along with the figures.

Table 7.5. Results obtained using the proposed Fuzzy controller

No	Original image	Segmented image (using Triangular membership function)	Difference image obtained by subtracting Original and Segmented images (using Triangular membership function)	Segmented image (using Gaussian membership function)	Difference image obtained by subtracting Original and Segmented images (using Gaussian membership function)	Segmented image (using Trapezoidal membership function)	Difference image obtained by subtracting Original and Segmented images (using Trapezoidal membership function)
1.							
2.							
3.							
4.							
5.							
6.							
7.							
8.							
9.							
10.							



The results given in Table 7.5 proves the efficiency of the proposed Fuzzy controller (which is designed for performing sky segmentation).

**DETECTION OF
STRUCTURALLY VARIANT
ERYTHROCYTES**

Chapter 8

A brief introduction on blood components and their functionalities are given in **Section 1.2 of Chapter 1**. Poikilocytes (erythrocytes whose shapes differ from the bi-concave disc shape like structure of normal erythrocytes) are introduced in that Section and the practical significance of timely detection of the presence of poikilocytes in human blood stream to prevent the severity of various life-threatening diseases are also discussed in the same Section.

Few decades ago, poikilocytes detection were done manually by counting the number of poikilocytes present in blood sample by medical professionals. This type of detection procedure is not only tedious but is also not reliable as it is subjected to human error. To overcome these drawbacks, researchers have proposed several automated poikilocytes detection methods to make the process reliable as well as less tedious .

8.1. Literature Survey

Inspired by the real-life significance of this problem, various researchers have proposed several automated poikilocytes detection methods. Brief discussion on the methodology of each of the existing methods is given in this Section and their shortcomings are also highlighted.

8.1.1. Counting based automated poikilocytes detection methods

In this Section, the methodologies of existing counting based automated poikilocytes detection methods are discussed in details and their shortcomings are highlighted.

In [134], the authors performed poikilocytes detection using Circular Hough Transform (CHT) method [135] based on the radius range of erythrocytes present in blood smear images. The radius range is computed by considering the difference between the maximum radii and minimum radii of erythrocytes after cropping them manually.

Limitations:

- This method is subjected to human error as it involves manual segmentation of erythrocytes from blood smear images.
- The use of CHT technique to perform poikilocytes detection in this method also enhances the chances of erroneous detection as CHT technique detects all erythrocytes having circular shapes like spherocytes and stomatocytes (whose morphology are given in Fig.1.2.2.3 of **Chapter 1**) as normal erythrocytes because of the similarities in their morphology with that of normal erythrocytes.

The authors in [136] performed poikilocytes detection using the CHT method similarly as done in [134] but instead of manually segmenting erythrocytes from other components present in blood smear images, the authors performed segmentation of erythrocytes using a histogram based threshold technique in this method.

Limitation:

- Although the histogram based threshold technique reduces the error associated with the manual erythrocytes segmentation method used in [134] but the chances of erroneous detection of poikilocytes having circular morphology as normal erythrocytes due to the use of CHT technique also exists in [136].

In [137], the authors performed segmentation of erythrocytes from rest of the blood components present in blood smear images using similar intensity based threshold technique as done in [136] but instead of using CHT method for performing poikilocytes detection, they detected poikilocytes using an Iterative Structured Circle detection method which they have designed by exploiting the circle detection strategies adapted in randomized circle detection techniques.

Limitations:

- Like the authors of the methods proposed in [134] and [136], the authors of this method too performed poikilocytes detection using circle detection algorithms. Hence, this method also cannot accurately detect poikilocytes having circular morphology similar to that of the normal erythrocytes.
- This method fails to effectively separate the overlapped erythrocytes present in blood smear images and thus enhances chances of erroneous erythrocytes detection.

The authors in [138] tried to detect the overlapped erythrocytes using the following mathematical equation:

$$RBC_o = \frac{\text{Detected Total Pixel Area of RBC}}{\text{Approx Pixel Area of Single RBC}} \quad (8.1.1.1)$$

If the RBC_o value for any erythrocyte is much higher than one, then that erythrocyte is considered as overlapped erythrocyte and eliminated from counting process. The authors in [139] tried to solve this issue by segmenting overlapped cells using a clump splitting process.

All the methods discussed in this Section are counting based methods which are designed to detect and segment erythrocytes from other components present in blood smear images and then counting them. None of these methods can classify erythrocytes into different classes based on the variations in their morphology.

To solve these limitations, several authors have designed handcrafted features (features extracted by human experts) based automated erythrocytes classification methods. The methodologies and limitations of these methods are discussed in the following Section.

8.1.2. Handcrafted features based automated erythrocytes classification methods

Brief descriptions of the methods designed for classifying various types of poikilocytes by several authors using handcrafted features and traditional classifiers are given in this Section. The limitations of these methods are also highlighted here.

In [140], the authors performed classification of normal and abnormal erythrocytes using compactness and 7 invariant moment features [141] extracted from erythrocytes which are segmented from blood smear images using the Otsu's threshold technique [142]. In this method, classification is done using Artificial Neural Network.

The authors in [143] classified erythrocytes using a set of twenty seven handcrafted features which includes four geometrical features (area, compactness, perimeter and form factor), 7 invariant moment features and sixteen first order and second statistical features like mean, skewness, kurtosis, energy, etc. The classifier used in this method is Back Propagation Neural Network [144].

Six types of poikilocytes namely macrocytic normochromic, macrocytic hypochromic, normocytic normochromic, normocytic hypochromic, microcytic normochromic and microcytic hypochromic are classified in [145] using a set of thirty five features which includes geometrical features (area, compactness, major axis length, minor axis length, area, etc.), textural features (Harlick features) as well as entropy based features. Bayesian classifier is used in this method to perform classification.

In [146], the authors classified four types of poikilocytes namely, elliptocytes, dacrocytes, macrocytes and echinocytes using a set of geometrical features (form factor, major axis, minor axis, perimeter, compactness, eccentricity, etc) using Artificial Neural Network and Decision Tree classifier respectively and compared their performance efficiencies.

Maity *et al.* [147] classified eight types of poikilocytes (namely, microcytic, elliptocyte, macrocytic, dacrocyte, sickle, kerocytes, acanthocytes, echinocytes) and normal erythrocytes using geometrical features (area, perimeter, compactness, solidity, etc.). C4.5 decision tree classifier [148] is used here for performing classification.

The authors in [149] classified normal and abnormal erythrocytes using a set of geometrical features (area, perimeter, compactness, etc) and textural features (correlation, entropy, energy, etc) using Support Vector Machine and Extreme machine learning [150] classifiers respectively and compared their accuracy.

In [151],the authors classified three different types of poikilocytes namely, dacrocytes, elliptocytes and schistocytes using geometrical features (area, perimeter, major axis, minor axis, solidity, eccentricity,etc.). The authors performed classification by applying majority voting theory on the results obtained from three classifiers SVM, k-Nearest Neighbor (KNN) algorithm [152] and Neural Network.

The authors in [153] performed classification of normal and abnormal erythrocytes using 7 invariant moment features, geometrical features (area, major axis, minor axis,perimeter), textural features, kurtosis, skewness, etc using Back Propagation Neural Network and LeNet 5 network [154] respectively and compared their accuracy.

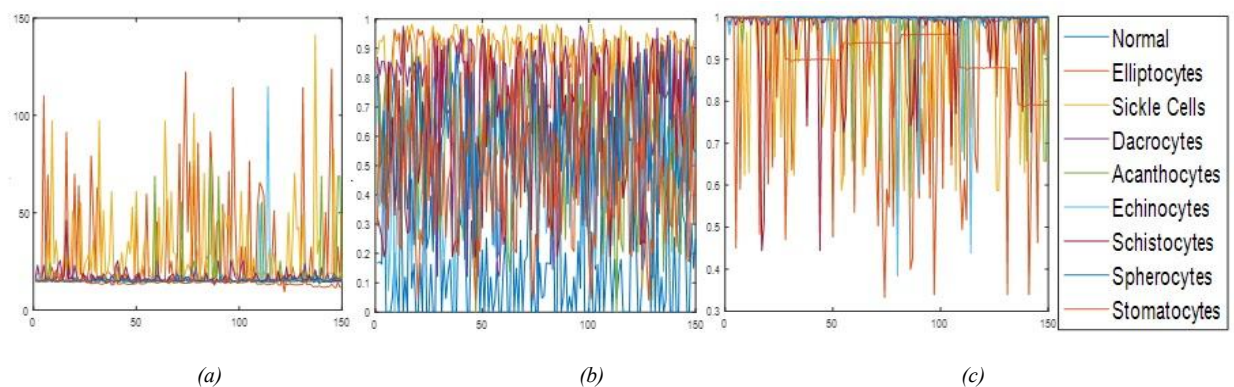
In [155], the authors designed a hybrid neural network classifier to classify normal and abnormal erythrocytes using geometrical features (circularity, medical axis ratio,eccentricity), textural features and entropy based features.

The authors in [156], proposed a digital in-line holographic microscopy (DIHM) and machine learning based erythrocytes classification method. They extracted erythrocytes by segmenting the area around the centre portions of their reconstructed hologram images obtained using local-peak-searching algorithm. Following segmentation, they extracted 12 features (information on intensity distributions, morphological and optical focusing characteristics) from segmented erythrocytes and classified them into three classes (discocytes, echinocytes, and spherocytes) using decision tree model with 97% accuracy.

Some of the common handcrafted features used in these works are listed as follows:

- Compactness
- Eccentricity
- Solidity
- Form factor
- Medical Axis Ratio
- 7 Invariant Moments

Initially, these features are extracted from randomly selected 150 cells of each type of erythrocytes whose morphological details are given in Fig. 1.2.2.3 of **Chapter 1**. The publicly available sources from which these cell images are taken are mentioned in Chapter 9. The plots showing the overlapping characteristics of handcrafted features extracted from the cells are given in Fig. 8.1.2.1.



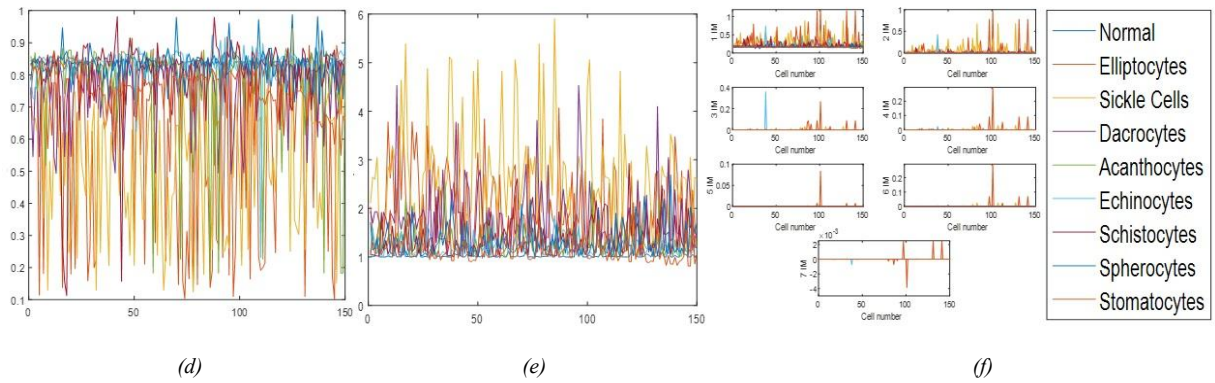


Fig.8.1.2.1. *Overlapping characteristics of handcrafted features (a) Compactness, (b) Eccentricity, (c) Solidity,(d)Form Factor, (e) Medical Axis Ratio, (f) 7 Invariant Moments,*

From the above plots, it is seen that for all the features, the values obtained for each type of erythrocyte often cannot be distinguished from each other which often leads to the production of erroneous results as neural networks classify objects into different classes based on the variations between numerical values of features extracted from objects belonging to different classes.

This limitation of existing handcrafted features based erythrocytes detection methods have urged authors to propose various CNN based erythrocytes classification methods as CNN has inherent feature extraction capability and thus these type of networks solve the problems which arise while performing classification using erroneous features.

8.1.3. CNN based automated erythrocytes classification methods

Brief descriptions of the existing CNN based automated erythrocytes detection methods are given in this Section and their shortcomings are also highlighted.

The authors in [157] designed a novel CNN architecture comprising of three convolutional layers, three pooling layers, two dropout layers and one fully-connected layer to classify eight types of poikilocytes (namely, discocytes, echinocytes, elongated, granular, oval, reticulocytes, sickle and stomatocytes).

In [158], the authors performed identification of malaria infected cells using LeNet5 [154], AlexNet [159] and GoogLeNet [160] respectively and stated that each CNN gives better classification accuracy compared to SVM trained with seven colour, entropy and shape features to perform classification of similar data.

Ten different types of erythrocytes (namely normal,echinocyte, dacrocyte, schistocyte, elliptocyte, acanthocyte, stomatocyte ,spherocyte, overlap and target cell) are classified in [161] by a 150 layer CNN having several dense connections. The architecture of this CNN highly resembles the architecture of DenseNet which is proposed in [162].

In [163], the authors performed segmentation and classification of erythrocytes collected from patients suffering from SCD using deformable U-Net which is designed by replacing the convolutions layers of the U-Net [164] with deformable convolution layers to make it more robust towards local features.

The authors in [165] classified erythrocytes into two classes namely, crossiant and slippers using a CNN whose architecture comprises of alternate convolution, pooling and Rectified Linear Unit (ReLU) layers but unlike previous approaches, it can take on a range of floating point values as it replaces the classification layer with a regression layer with linear transfer functions.

The CNN features used in these methods are subjected to inherent information loss as all of them have used ReLU as an activation function.

Mathematically, ReLU is explained as follows:

$$f(x) = \max(x, 0) \quad (8.1.3.1)$$

The information loss occur due to suppression of negative values.

Some authors have proposed few variants of ReLU to solve this issue. The proposed variants are mathematically represented as follows:

i. Leaky ReLU (LReLU) [166]:

$$f(x) = 0.01 * x \quad \text{for } x < 0 \quad (8.1.3.2)$$

$$f(x) = x \quad \text{for } x \geq 0 \quad (8.1.3.3)$$

Although LReLU slightly solves the problem by partially suppressing the negative values, but the results thus produced are not consistent.

ii. Parametric ReLU (PReLU) [167]:

$$f(x) = \alpha * x \quad \text{for } x < 0 \quad (8.1.3.4)$$

$$f(x) = x \quad \text{for } x \geq 0 \quad (8.1.3.5)$$

Here, the authors replaced 0.01 in LReLU by ‘ α ’ to improve its performance. But PReLU saturates for large negative values.

iii. Exponential ReLU(EReLU) [168]:

$$f(x) = \alpha(\exp(x) - 1) \quad \text{for } x < 0 \quad (8.1.3.6)$$

$$f(x) = x \quad \text{for } x \geq 0 \quad (8.1.3.7)$$

Although EReLU gives better results compared to LReLU and PReLU, but it also saturates for very large negative values.

iv. Concatenated ReLU (CReLU) [169] concatenates both the positive and negative parts of the activation, thus preserving both the positive and negative values.

Here the authors replaced 0.01 in LReLU by ‘ α ’ to improve its performance. But PReLU saturates for large negative values.

In [170], the authors tried to solve this information loss issue and after performing extensive analyses. They stated that the performances of CNN improve after removing the last ReLU layer from their architectures.

Inspired by the observation of the authors in [170] and the methods proposed in [171]-[172] which stated that performing classification using feature ensemble formed by incorporating features of multiple CNN is beneficial compared to performing similar operation using features of single CNN, as shallow CNN provide more general features whereas deep CNN provide specific features, so combination of these features help in better image representation and also significantly improves the methods’ performances, a novel two-stage feature selection algorithm is proposed in this thesis.

This is probably the first algorithm proposed which performs selective suppression of positive or negative values or none of each feature, depending upon its significance in classification. The proposed algorithm facilitates the selection of only significant features, that too in their most informative form (Fully Connected (FC)/ReLU/ InverseReLU), in contrary to any of the above-mentioned variants which suppresses negative values of all features identically ignoring their contribution in performing classification.

Detailed description of each step of the proposed two-stage feature selection algorithm is carried out in **Chapter 9** and improvement in the results obtained using the proposed two-stage feature selection method while performing detection of nine types of erythrocytes over existing methods are also shown in **Chapter 9**.

Chapter 9

Detailed discussion on each step of the proposed two-stage feature selection algorithm designed in this thesis is carried out in this Chapter. Comprehensive comparative analyses of the performance efficiencies of the proposed method as well as several existing methods are also conducted in this Chapter to prove the excellence of the proposed method over the state-of-the-art methods.

The proposed method aims at performing automatic erythrocytes detection using ensemble of selected features obtained from multiple CNN (namely, AlexNet [159], Vgg16 and Vgg19 [171]) to overcome the shortcomings of existing works arising due to the highly overlapping characteristics of handcrafted features. The main merit of this work lies in the proposed two-stage feature selection algorithm, which reduces information loss occurring in traditional CNN due to the suppression of negative values of features by ReLU activation function and also largely reduces the feature space's dimension. Moreover, it is the first algorithm proposed which is capable of selectively suppressing the positive or negative values or none of each feature depending upon its respective significance in performing classification, in contrary to the previously proposed variants of ReLU. The first stage of the feature selection algorithm deals with construction of a feature space for each CNN by performing inter-selection among its Fully Connected (FC), ReLU and InverseReLU features and selecting features possessing minimum Fuzzy Entropy (FE) and maximum newly formulated Total Contribution Score (TCS) values simultaneously. In the second stage of the algorithm, intra-selection among features within each selected feature space of each CNN is performed to eliminate less significant features which simultaneously satisfy the stated redundancy and non-relevancy criteria. Finally, feature ensemble is created by concatenating the selected features obtained from each CNN and detection of nine types of erythrocytes is performed using the created feature ensemble. The proposed method registers mean average precision (mAP) of 98.6%, thus proving its excellence over existing works.

The architecture of the deep neural network designed in this method is inspired by the architecture of the Regions with CNN (R-CNN) [172] except the fact that in R-CNN, the features extracted from one CNN is used to perform object detection while in the designed network selected features from multiple CNN are used to perform erythrocytes detection.

9.1. Data collection

The data (blood smear images) containing the nine types of erythrocytes (whose morphology are given in Fig.1.2.2.3 of **Chapter 1**) are obtained from several publicly available sources e.g. websites containing royalty free stock photos of RBC [173]-[175] and image banks containing blood smear images of different types of erythrocytes [176]-[179].

9.2. Fine-tuning

All three CNN considered for performing feature extraction in this work are trained with images from ImageNet database [180]. The characteristics of images in that database differ widely from that of blood smear images obtained from [173]-[179] as ImageNet database mostly comprises of generic images, whereas blood smear images are microscopic images. So fine-tuning of these CNN are essential for performing proper feature extraction. For this reason, initially a 1000-way classification layer of each of these networks is replaced with a 10-way classification layer (where 9 represents the number of erythrocytes' classes and 1 for background). Then the parameters of CNN are trained using stochastic gradient descent algorithm [181] and learning rate (0.001). All region proposals with ≥ 0.6 Intersection over Union (IoU) overlap with ground-truth box are treated as positive instances for that box class only while the rest are treated as negative instances for that class.

9.3. Construction of feature spaces

Initially, region proposals are extracted from each image similarly as done in [172] and the generated regions proposals are wrapped according to the respective input dimension specifications of CNN (e.g. 227x227 (AlexNet) and 224x224 (Vgg16 and Vgg19)) and then the wrapped region proposals are fed to each CNN for feature extraction.

The features are extracted from the FC layer located before the last ReLU layer of each CNN as it is proved in [170] that this layer provides the most informative features.

After initial feature extraction, a feature space is obtained for each CNN which comprises of features consisting of both positive and negative values (FC features). Following which, ReLU and InverseReLU activation functions are applied on features in FC feature space of each CNN respectively.

InverseReLU activation function is mathematically expressed as:

$$f(x) = \min(0, x) \quad (9.1.3.1)$$

ReLU and InverseReLU features consist of positive and negative values of corresponding FC features respectively.

So, at the completion of this stage, three (FC, ReLU and InverseReLU) features sets are obtained for each CNN.

9.4. Proposed two-stage feature selection method

The proposed two-stage feature selection method performs feature selection based on two parameter's values namely, Fuzzy Entropy (FE) and Total Contribution score (TCS) parameters. Details of these parameters are given in the following Sections.

9.4.1. Fuzzy Entropy [31]

This parameter measures the fuzziness (uncertainties) of features. So higher FE suggests lesser information content and vice versa.

9.4.2. Total Contribution Score

TCS is a novel parameter which is formulated in this work by exploring the basic concepts of neural networks to measure the total contribution of each feature in performing classification.

In order to perform TCS evaluation for each feature, a fully connected neural network consisting of an input layer, a hidden layer and an output layer is designed whose architecture is given in Fig.9.4.2.1. The number of neurons in the input and output layers are equal to total number of input features and total number of output classes respectively. The number of neurons in hidden layer are set to according to the three thumb rules stated in [182].

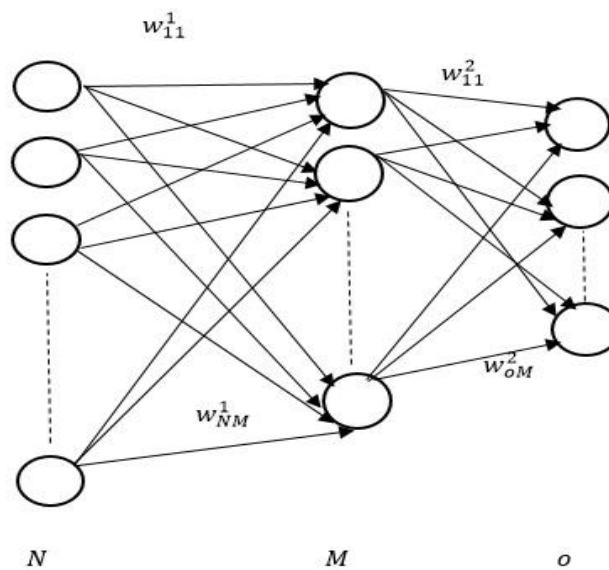


Fig.9.4.2.1. Neural network designed to perform TCS evaluation

The neural network is trained using mini-batch gradient descent optimization algorithm and back-propagation as a gradient computing technique. The weight values are initialized as stated in [183].

$$\mathbf{W}^{[l]} = N \left(0, \sqrt{\frac{2}{n^{[l]} + n^{[l-1]}}} \right) \quad (9.4.2.1)$$

This method of weight initialization is called Xavier/He initialization. In (9.4.2.1), ' $\mathbf{W}^{[l]}$ ' and ' $n^{[l]}$ ' represent the weights and the number of neurons in the ' l^{th} ' layer of the neural network respectively. Here as the neural network uses softmax as an activation function in the output layer so its loss function is Cross Entropy Loss function which is mathematically represented as:

$$L(y, \hat{y}) = - [(1 - y) \log (1 - \hat{y}) + y \log \hat{y}] \quad (9.4.2.2)$$

In (9.4.2.2), ' y ' and ' \hat{y} ' represent actual output and predicted output respectively.

Since in this work, mini-batch gradient descent algorithm is used so the cost function is:

$$J = \frac{1}{B} \sum_{i=1}^B L^{(i)} \quad (9.4.2.3)$$

In (9.4.2.3), ' $L^{(i)}$ ' and ' B ' represent loss function for i^{th} sample and mini-batch size respectively. The value of ' B ' is set to 64. As L2 regularization is used here to avoid over-fitting, so one more term is added to the above equation and the final cost function becomes:

$$J_{new} = J + \frac{\lambda}{2} \|\mathbf{W}\|^2 \quad (9.4.2.4)$$

$$J_{new} = J + \frac{\lambda}{2} \mathbf{W}^T \mathbf{W} \quad (9.4.2.5)$$

The weights are updated using the following weight update rule:

$$\mathbf{W}_{t+1} = \mathbf{W}_t - \alpha \frac{\partial J}{\partial \mathbf{W}_t} - \alpha \frac{\lambda}{2} \frac{\partial \mathbf{W}_t^T \mathbf{W}_t}{\partial \mathbf{W}_t} \quad (9.4.2.6)$$

In (9.4.2.6) ' \mathbf{W}_{t+1} ' and ' \mathbf{W}_t ' represent weight values after and before upgradation respectively, ' α ' represents learning rate. The value of ' α ' and ' λ ' are set 0.01 and 0.001 respectively.

In each session, only one feature space is given as input to the neural network at a time to obtain the updated weight values.

So at the end of each session, two sets of updated weights are obtained. One set of weights connects the input layer's neurons with hidden layer's neurons while another set of weights connects hidden layer's neurons with the output neurons. The magnitude of weight connecting two neurons signifies the strength of connections between them or in other words the weight represents the influence of one neuron have on another neuron.

Here the magnitudes of updated weights connecting the input layer neurons with the hidden layer neurons are used to evaluate the TCS values, as these weight values are directly related to the input features.

Higher the magnitude of the weight connecting an input neuron to a hidden layer neuron, more will be its influence on classification due to the following reasons:

The magnitude of the weight connecting an input neuron with a neuron in the hidden layer gives the measure of the influence that input feature has on the output of that hidden neuron.

The output of that hidden neuron in turn will be fed as inputs to all output neurons in the output layer, so the predicated classes, which are obtained as final outputs also become indirectly influenced by the input feature.

From these reasons, it can be concluded that the total contribution of a feature in performing classification can be evaluated by performing the summation of the magnitudes of weights connecting it to the hidden layer neurons, so this parameter is termed as Total Contribution Score (TCS).

The entire process is explained as follows:

At the end of each session, two sets of updated weight matrices are obtained, one containing the updated weights connecting the input neurons with the hidden layer neurons while the other containing the updated weights connecting the hidden layer neurons with the output neurons. Here focus is on the updated weights connecting the input neurons with the hidden layer neurons. The dimension of that updated weight matrix is $M \times N$, where ' M ' and ' N ' represent the number of hidden neurons and input features respectively.

Let ' W_{ji} ' be the weight connecting the ' i^{th} ' input neuron to the ' j^{th} ' hidden neuron. TCS value of the ' i^{th} ' feature is calculated using the following equation:

$$TCS_i = \sum_{j=1}^M |W_{ji}| \quad (9.4.2.7)$$

The above equation is formulated from the concept that the weights connecting the ' i^{th} ' feature with all ' M ' hidden neurons, give the measure of the influence of the ' i^{th} ' feature on the outputs of ' M ' hidden neurons which in turn influence the final outputs or in other words it gives one the measure of the total influence of the ' i^{th} ' feature on classification. Hence, the summation of the magnitudes of these weights give the desired TCS value of the ' i^{th} ' feature.

TCS values are computed for all FC features giving the FC feature space of each CNN as inputs to the neural network each time.

For computing TCS values of ReLU features, a particular feature in the FC feature space of each CNN is replaced with its corresponding ReLU counterpart each time, keeping all other features unchanged and feeding the entire feature space as inputs to the designed neural network to evaluate the TCS value of that particular ReLU feature. Similar operation is repeated while computing the TCS values of InverseReLU features also.

Here, only one feature in the FC feature space of each CNN is replaced either with its corresponding ReLU feature or with its corresponding InverseReLU feature each time keeping all other features and parameters of neural networks unaltered while evaluating their respective TCS values in order to measure how the contribution of that particular feature varies by suppressing its positive values or negative values respectively in an identical environment.

Finally, after computing both FE and TCS values of FC, ReLU and InverseReLU features of each CNN, one set of FE values and one set of TCS values are obtained for each type of features of each CNN having dimension $1 \times N$. So, altogether nine FE sets and nine TCS sets are obtained.

9.4.3. First stage of the proposed feature selection algorithm

The first stage of the algorithm deals with the construction of a new feature space for each CNN by performing inter-selection among its FC, ReLU and InverseReLU features and selecting features which possesses minimum FE and maximum TCS values simultaneously, thus eliminating chances of information loss. The pseudo-code of this stage of the algorithm is given in Fig.9.4.3.1 below.

A. Input: $FE_{FC}[1 \times N]$, $FE_{IR}[1 \times N]$, $FE_R[1 \times N]$, $TCS_{FC}[1 \times N]$, $TCS_{IR}[1 \times N]$, $TCS_R [1 \times N]$

Output: Indices[$1 \times N$]-contain indices of features possessing minimum FE and maximum TCS values simultaneously

1. Initialization: $p=0$, $k=0$, $indices=zeros[1 \times N]$;

2. **for** $j=1$ to N **do**

3. $a1=FE_{FC}(1,j)$; $b1=FE_{IR}(1,j)$; $c1=FE_R(1,j)$; $d1=TCS_{FC}(1,j)$; $e1=TCS_{IR}(1,j)$; $f1=TCS_R(1,j)$;

5. $FE_{min}=\min(a1,\min(b1,c1))$;

6. $TCS_{max}=\max(d1,\max(e1,f1))$;

```

7.   if (FEmin==a1) then      8.   p=1;
9.   if (FEmin==b1) then      10.  p=2;
11.  if (FEmin==c1) then      12.  p=3;
13.  if (TCSmax==d1) then     14.  k=1;
15.  if (TCSmax==e1) then     16.  k=2;
17.  if (TCSmax==f1) then     18.  k=3;
19.  if (p==k) then          20.  Indices(1,j)=p;
21.  end for

22. return Indices[1xN]

B. Input: Indices[1xN]

Output: 3 arrays (the column indices of different features) and the respective sizes of these arrays

1. [r1,c1]=find(Indices==1); [r2,c2]=find(Indices==2);
   [r3, c3]=find(Indices==3);
2.[mc1,nc1]=size(c1);[mc2,nc2]=size(c2);
   [mc3,nc3]=size(c3);
3. return c1,c2,c3,nc1,nc2,nc3

C. Input: FeaturesFC[PxN], FeaturesIR[PxN], FeaturesR [PxN],c1[1xnc1],
c2[1xnc2],c3[1xnc3],nc1,nc2,nc3

Output: Selected features [Px(nc1+nc2+nc3)]

1.Initialization:FC[Pxnc1]=zeros[Pxnc1], IR[Pxnc2]=zeros[Pxnc2], R[Pxnc3]=zeros[Pxnc3]
2. for i1=1 to nc1 do 3. a2=c1(1,i1);
4. FC(:,i1)=FeaturesFC(:, a2); 5. end for
6. for i2=1 to nc2 do 7. a3=c2(1,i2);
8. IR(:,i2)=FeaturesIR(:, a3); 9. end for
10. for i3=1 to nc3 do 11. a4=c3(1,i3);
12. R(:,i3)=FeaturesR(:, a4); 13. end for
14. Features selected after the 1st stage=horzcat (FC,IR,R);
15. return Selected features

```

Fig. 9.4.3.1. Psuedo code of the 1st stage of the feature selection of each CNN model

9.4.4. Second stage of the proposed feature selection algorithm

In the second stage of the algorithm, intra-selection of features within each selected feature spaces obtained for each CNN is performed by eliminating features which simultaneously satisfies the redundancy and non-relevancy criterion stated below, thus ensuring the selection of most significant features.

a) Redundancy criteria states that if FE values of two or more features become equal, then those features are termed as redundant features.

b) Non-relevancy criteria states that if the TCS value of a feature is below the empirically chosen threshold, 'th', then it satisfies the non-relevancy condition.

Mathematically the threshold, 'th' is defined as:

$$th = 0.20 * \frac{1}{L} \sum_{i=1}^L TCS_i \quad (9.4.2.8)$$

In (9.4.2.8), 'L' represent the total number of features in the feature space of each CNN obtained after the first selection stage. The value of 'L' may vary for each CNN.

At this stage, TCS values of features belonging to the selected feature space of each CNN obtained after the first stage are computed by giving the selected features as inputs to the neural network after modifying the number of input neurons and hidden neurons as mentioned earlier.

At the end of this stage, a feature space is obtained for each CNN which comprises of final selected features. These selected features obtained from each CNN after two-stage feature selection are concatenated to form the feature ensemble using which erythrocytes detection is performed here.

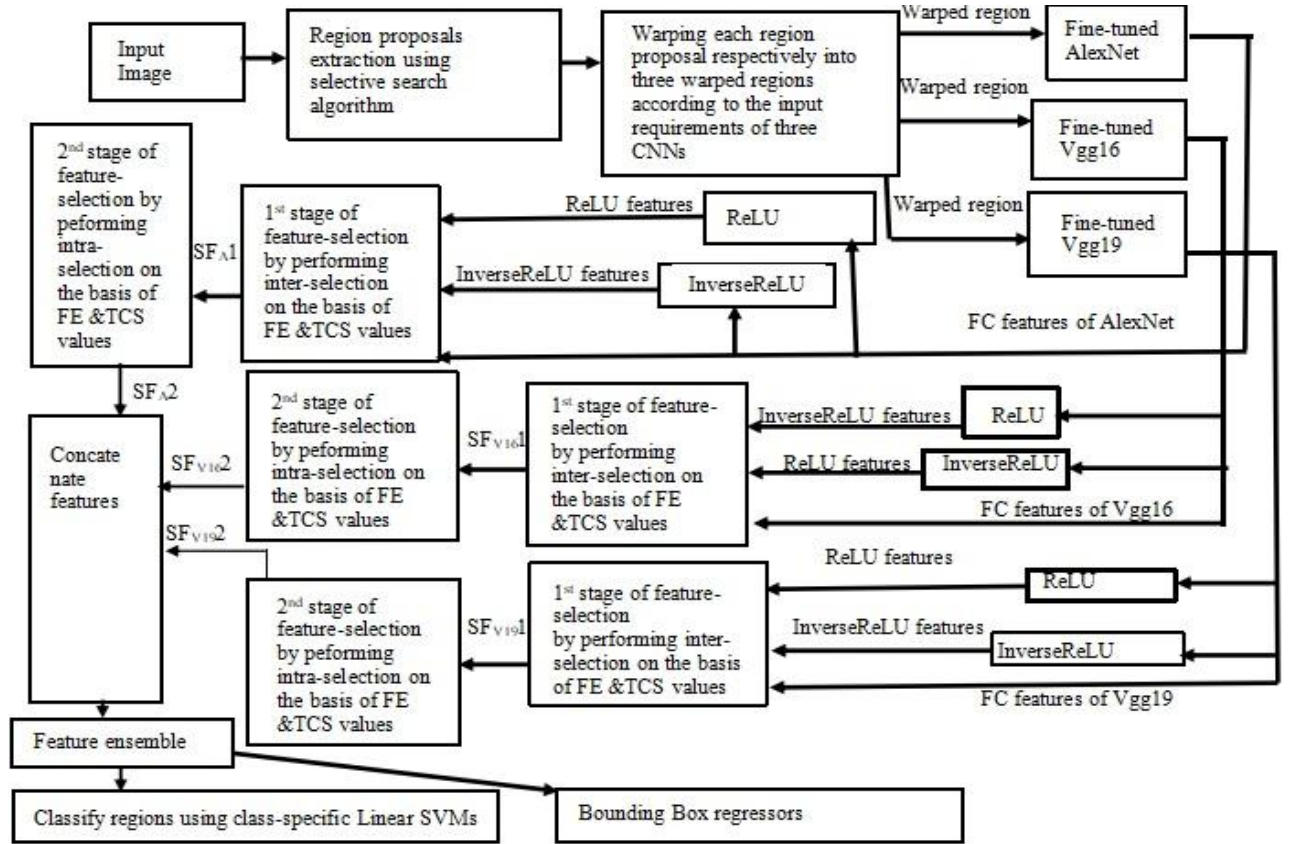


Fig.9.4.1. Block diagram of the proposed two-stage feature selection method

9.5. Experimental Results

Results obtained at each stage of the proposed method as well as the comparative analyses' results are given in this Section. FE and TCS values computed for FC,ReLU and InverseReLU features at each stage of the proposed two-stage feature selection algorithm are given in graphical form in Fig.9.5.1 and Fig.9.5.2 respectively. Stage-wise reduction of feature space dimension is shown in Table 9.5.1. Stage-wise improvement achieved in the performance of the designed method with the introduction of the two-stage feature selection algorithm is shown in Table 9.5.2. Comparative analyses of the mAP values obtained from the proposed method as well as from several existing methods using only ALL-IDB database [176] are given in Table 9.5.3. Comparative analyses of the mAP values obtained from the proposed method as well as CNN based erythrocytes detection methods and handcrafted features based erythrocytes detection methods on the entire database considered in this work are given in Table 9.5.4 and Table 9.5.5 respectively. Some examples of output results are given in Fig.9.5.3.

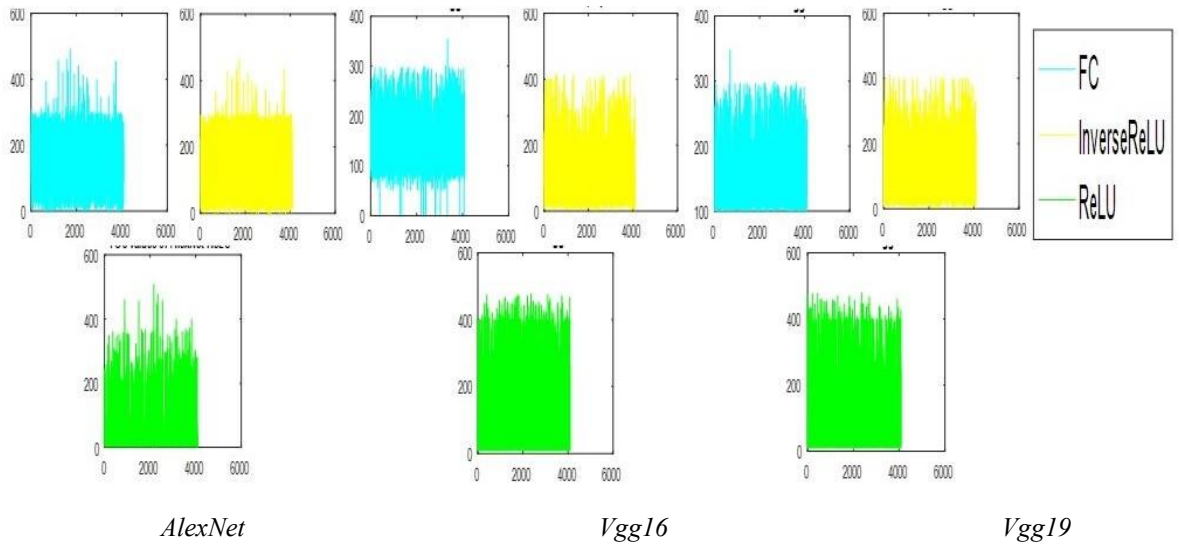
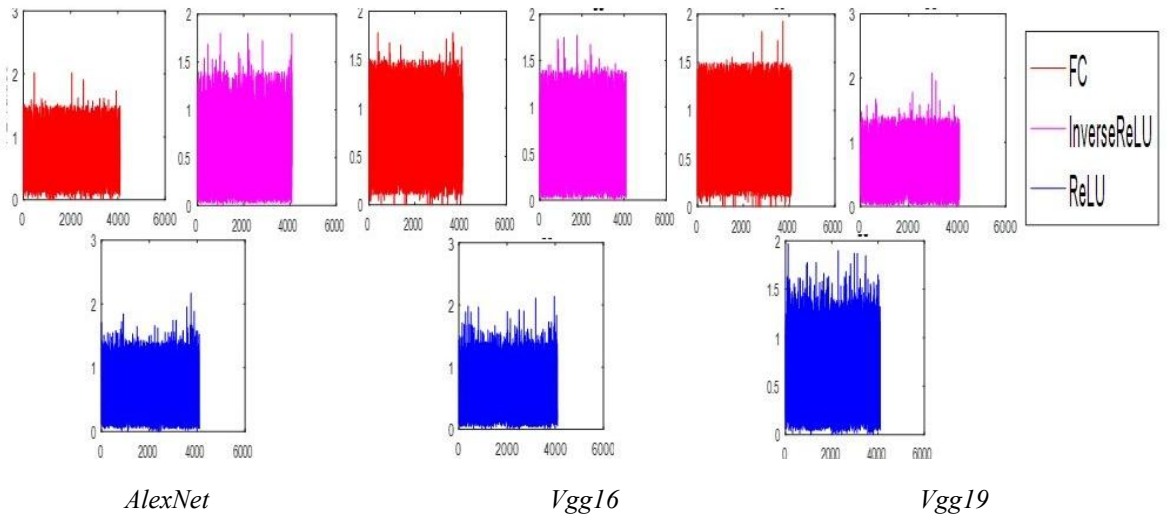
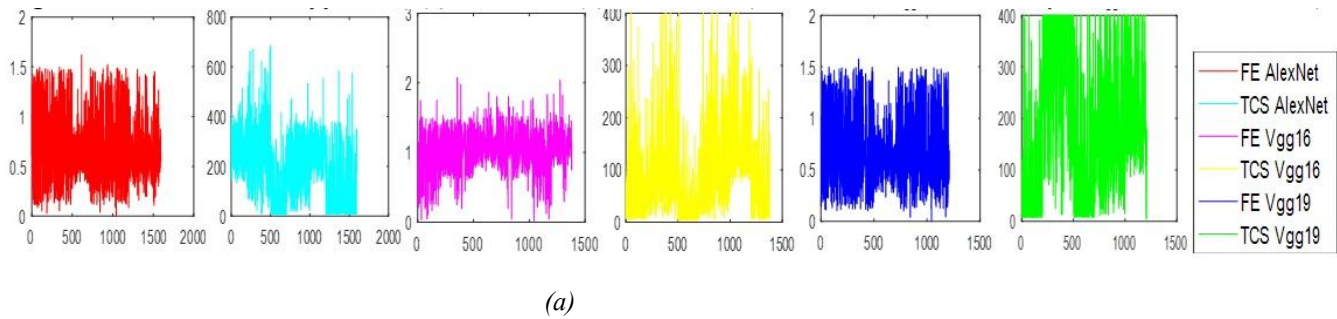


Fig. 9.5.1. FE and TCS values of features **(a)** FE values, **(b)** TCS values (x-axis: Feature dimension, y-axis: Parameter values)



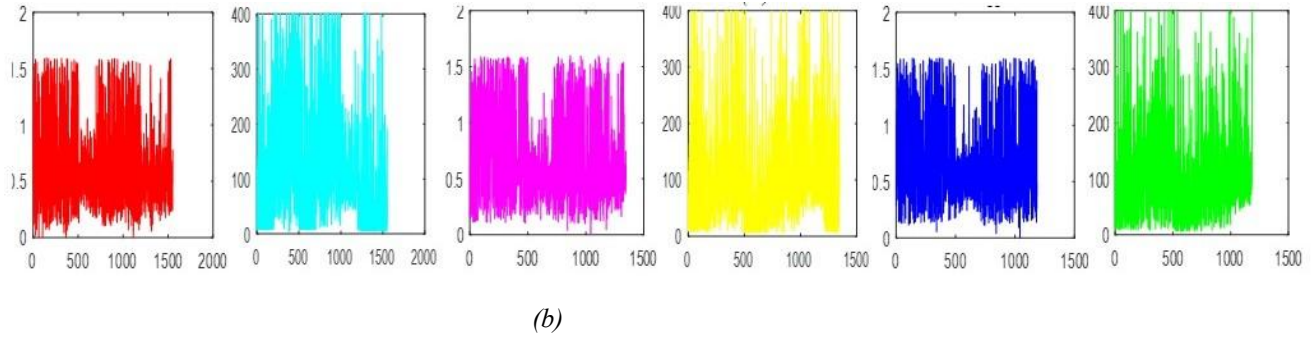


Fig.9.5.2. FE and TCS values of features selected at different stage **(a)** First stage, **(b)** Second stage(x-axis: Feature dimension, y-axis: Parameter values)(Please zoom to see the axis ranges)

Table 9.5.1. Stage-wise reduction in dimension of each CNN’s feature space

Type of CNN model	Initial Feature dimensionality	Feature dimensionality after the first stage of feature selection	Feature dimensionality at the end of second stage of feature selection
AlexNet	4096	1594	1552
Vgg16	4096	1380	1345
Vgg19	4096	1210	1185

Table 9.5.2. Improvement in results obtained due to the introduction of the proposed two-stage feature selection algorithm

	Normal	Elliptocytes	Sickle cells	Dacrocytes	Acanthocytes	Echinoocytes	Stomatocytes	Spherocytes	Schistocytes	mAP
FC _{AlexNet} (no fine-tuning)	60.7	62.9	66.5	38.9	69.2	48.7	56.4	70.6	76.5	61.2
FC _{AlexNet} (fine-tuning)	72.2	65.8	68.9	44.6	72.0	55.8	64.9	76.5	71.9	65.8
FC _{Vgg16} (no fine-tuning)	78.4	73.1	66.8	52.6	78.2	61.6	67.4	83.7	84.5	71.8
FC _{Vgg16} (fine-tuning)	84.5	76.8	70.5	60.4	73.2	68.9	69.2	78.6	83.7	74.0
FC _{Vgg19} (no fine-tuning)	82.6	78.4	72.9	68.1	79.9	75.1	73.7	82.3	87.5	77.8
FC _{Vgg19} (fine-tuning)	85.9	82.8	78.6	72.5	81.7	88.5	82.6	89.7	92.7	83.9
Feature ensemble (FC _{AlexNet} &FC _{Vgg16} &FC _{Vgg19}) (no feature-selection & no fine-tuning)	95.7	88.5	85.1	87.5	93.7	91.9	86.3	90.4	96.1	90.6
Feature ensemble (FC _{AlexNet} &FC _{Vgg16} &FC _{Vgg19}) (no feature-selection but with fine-tuning)	96.6	92.5	87.6	89.6	95.9	92.7	88.6	91.9	97.3	92.5
Proposed Methodology	99.8	98.5	97.2	98.1	99.3	98.7	97.9	98.6	99.2	98.6

Table 9.5.3. Comparative analysis of proposed method with existing works using ALL –IDB database [176]

Ref	Features used	Abnormalities detected	Performance
Tomari. <i>et.al</i> [140] (#)	Morphological features: Compactness and Moment Invariants	Normal and Abnormal RBCs (But types of abnormalities are not detected)	82%
Razzak <i>et al.</i> [184] (⊖)	Morphological features: Area, Circularity, Parameter, Centroid, Medial axis ratio, Cell deform ratio, Roughness, Regularity, Uniformity and Coarseness	Normal, Sickle cells, Elliptocytes & Ovalocytes, Acanthocytes, Burr cells and Helmet cells	83.14%
Shirazi <i>et al.</i> [149] (⊖)	Morphological features: Area, Perimeter, Compactness, Circularity, Rectangularity Texture features: Entropy, Energy, Mean, Min, Max, Contrast, Correlation and Homogeneity	Normal and Abnormal RBCs (But types of abnormalities are not detected)	95.3% (ELM) 90% (SVM)
Proposed method	Feature ensemble (obtained by grouping selected features of multiple CNNs)	Normal, Elliptocytes, Sickle cells, Dacrococytes, Echinocytes, Acanthocytes, Schistocytes, Spherocytes, Stomatocytes	96.9%

Table 9.5.4. Comparative analysis of proposed method with existing CNN based works

Ref	CNN architecture	Information loss	Abnormalities detected	Performance
Xu <i>et al.</i> [157] (⊖)	10 layer CNN including 3 convolution, 3 pooling, 2 Dropout layers and a Fully connected layer with ReLU activation.	Present	Discocytes, Oval, Echinocytes, Enlogated, Sickle, Granular, Reticulocytes, Stomatocytes	88.6%
Dong <i>et al.</i> [158] (⊖)	LeNet , AlexNet, GoogleLeNet (These CNNs use ReLU as activation function)	Present	Malaria infected and non-infected RBCs	95.1%(LeNet) 92.7%(Alexnet) 98% (GoogLeNet)
Kihm <i>et al.</i> [165] (⊖)	12 layer CNN consisting of 1 Input, 3convolution, 3 max-pooling, 3 ReLU, 1output and 1 regression layers.	Present	Slipper, Croissant and Others	86.3%
Proposed method	Fine-tuned AlexNet, Vgg16 and Vgg19. Features extracted from these CNNs are used together for RBC detection after proper selection.	Absent	Normal, Elliptocytes, Sickle cells, Dacrococytes, Echinocytes, Acanthocytes, Schistocytes, Spherocytes, Stomatocytes	98.6%

Table 9.5.5. Comparative analysis of the proposed method with existing handcrafted features based methods

Ref	Features used	Abnormalities detected	Performance
Veluchamy <i>et al.</i> [143] (#)	Morphological features: Area, Perimeter, Compactness, Form Factor, First order statistical features: Mean, Dispersion, Variance, Average Energy, Skewness, Kurtosis, Median, Mode Second Order Statistical features: Energy, Entropy, Correlation, Inertia, Homogeneity Algebraic Moment Invariants: 7 Moment Invariants	Normal and Abnormal cells. (Abnormal cells are collected from patients infected with Sickle cell anaemia).	73.3%
Bhowmick <i>et al.</i> [145] (⊖)	First order statistical features: Energy, Skewness, Kurtosis, Means intensity, variance, Entropy based features: Shannon entropy, Renyi entropy, Kapurs entropy, Yager entropy	Macrocytic Normochromic, Macrocytic Hypochromic, Normocytic Normochromic, Normocytic Hypochromic,	88.99%

	Morphological features: Area, Perimeter, Eccentricity, Compactness,	Microcytic Normochromic, Microcytic Hypochromic,	
Dalvi <i>et al.</i> [146] (ω)	Morphological features: Area, Major Axis, Minor Axis, Perimeter, Form Factor, Diameter, Shape Geometry, Compactness, Eccentricity, Solidity, Bounding Box Ratio, Equidiameter, Extent, Circularity	Normal, Elliptocytes, Echinocytes, Tear drop cells (Tear drop cells), Macrocytes	76.54%(ANN) 73.87%(DT)
Maity <i>et al.</i> [147] (ω)	Morphological features: Perimeter, Area, Min-radius, Max-radius, Feret, Angle of the Feret, Breadth, Convex Hull, Convex Hull area, Radius of Minimal Bounding Circle, Count correct, Aspect Ratio, Roundness, Area Equivalent Diameter, Perimeter Equivalent Diameter, Equivalent Ellipse Area, Compactness, Solidity, Concavity, Convexity, Shape, RFactor, Modification Ratio, Sphericity, Area of Bounding Box, Rectangularity, Centroid Distance Mean, Centroid Distance Variance, Eccentricity, Orientation, Centre of Mass Algebraic Moment Invariants: 7 Moment Invariants	Normal, Microcytic, Macrocytic, Dacrocytes Sickle cells, Echinocytes, Elliptocytes, Acanthocytes, Keratocytes	98%
Lofli <i>et al.</i> [151] (#)	Morphological features: Area, Perimeter, Circularity, Equivalent Diameter, Major Axis, Minor Axis, Eccentricity, Convex Area, Extent Factor, Solidity Factor, Deviation Factor, Elongation, Difference of Major and Minor Axis, Difference of area and the area of an ellipse with equivalent major and minor axis, Focus, Euler Number	Dacrocytes, Schistocytes, Elliptocytes	97.5% 94.55% (SVM)
Lee <i>et al.</i> [155] (ω)	Morphological features: Cell circularity, Medial axes ratio, Deformation ratio, Eccentricity, Hausdorff distance Texture features: Mean, Variance, Smoothness of the cell, Entropy	Burr cells, Sickle cells, Horn cells, Elliptocytes	95.5%
Go <i>et al.</i> [156] (ω)	Information on intensity distributions: Intensity distributions, Full width at half-maximum, Peak-to-peak distance, Standard deviations of light intensities for focal plane image Morphological features: Area, Perimeter Optical focusing characteristics: Real and virtual foci, Real and virtual focal lengths, Maximum Intensity	Discocytes, Echinocytes, Spherocytes	96.04%(DT)
Proposed method	Feature ensemble containing selected features of three fine-tuned CNNs namely AlexNet, Vgg16 and Vgg19	Normal, Elliptocytes, Sickle cells, Dacrocytes, Echinocytes, Acanthocytes, Schistocytes, Spherocytes, Stomatocytes	98.6%

ANN: Artificial Neural Network, DT: Decision Tree classifier

**The methods which are re-implemented using the codes of handcrafted features available in the internet are marked by (#) and the methods whose mAP are calculated using the information derived from confusion matrices provided by the authors themselves (in their respective publications) are marked by (ω)*

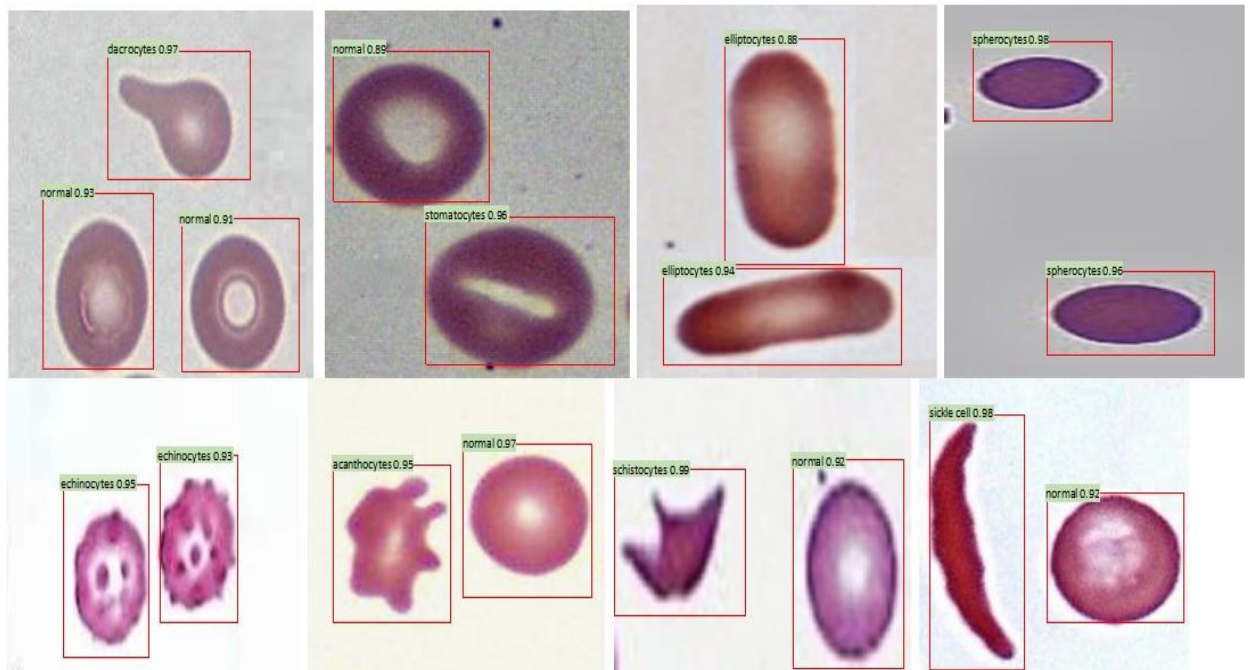


Fig.9.5.3.. Examples showing output results (bounding box with maximum score are displayed only) (Please zoom to see class labels)

The proposed method successfully overcomes the limitations of the existing works arising due to the highly overlapping characteristics of handcrafted features. The proposed two-stage feature selection algorithm efficiently deals with the inherent information loss occurring in traditional CNN due to the suppression of negative values of features. The two-stage feature selection algorithm proposed in this work depends on two parameters namely, a) FE and b) TCS, which measures the information content and total contribution in classification of each feature respectively. The first stage of the algorithm constructs feature space of each CNN by conducting inter-selection among its FC, ReLU and InverseReLU features, and selecting features possessing minimum FE and maximum TCS values simultaneously, thereby eliminating chances of information loss. The second stage of the algorithm deals with the elimination of redundant and less relevant features from the previously selected feature space of each CNN by implementing the redundancy and non-relevancy criteria, thus selecting only most significant features. This algorithm also largely reduces the dimension of feature space obtained from each CNN at each stage of the proposed method. Comparative analyses' results prove that the proposed algorithm outperforms existing works. Although some variants of ReLU are previously proposed to solve the inherent information loss problem occurring in traditional CNN but none of those networks perform selective suppression of positive or negative values or none of each feature in a feature space according to its significance in performing classification and preserve it (if selected) in its most informative form. This is the first algorithm designed to perform selective suppression of features based on their respective information content and contribution in performing classification .

**LAND COVER CLASSIFICATION
USING FULL-POLARIMETRIC
IMAGE DATA**

Chapter 10

Despeckling raw image data obtained from SAR and performing automated classification of despeckled data are one of the most important applications of remote sensing field. The raw data obtained from SAR are usually corrupted with multiplicative speckle noise which needs to be filtered from images before further processing to avoid chances of producing erroneous results. General introduction on SAR and its imaging technique, SAR image, POLSAR image and speckle noise are given in **Section 1.3** of **Chapter 1**. The methods proposed in this thesis are mainly focused on despeckling raw POLSAR image data obtained from SAR and classification of despeckled data. Hence, literature survey on this research area is also focused on these two topics.

10.1. Existing despeckling filters

Inspired by the practical significance of satellite image despeckling, many researchers have proposed various despeckling filters. The methodologies of these filters and their limitations are discussed in details in that Section.

10.1.1 Lee filter and its' proposed variants

Lee filter [185]: This filter is capable of suppressing both multiplicative as well as additive noises. It performs noise removal from each center pixel, $I(i, j)$ in accordance to the mean and variance values derived from its local statistics (characteristics of the neighboring pixels lying in the local window of pre-defined size centered at $I(i, j)$). Unlike most of the filters which process pixels in sequential order, Lee filter supports parallel processing of pixels. It performs noise suppression using the fundamentals of minimum mean square estimation (MMSE) criterion in its simplest form. Mathematically, the filtering operation of the Lee filter is explained as follows:

$$\widehat{R}(i, j) = \bar{I}(i, j) + W(i, j)(\bar{I}(i, j) - I(i, j)) \quad (10.1.1.1)$$

Where,

$$W(i, j) = 1 - \frac{C_v}{C_l}$$

\widehat{R} =Estimated filtered image, $C_v = \frac{\sigma_v}{\bar{v}}$ =Variance coefficient of noisy image with standard deviation, σ_v , $C_l = \frac{\sigma_l}{\bar{l}}$ = Variance coefficient of noise-free image with standard deviation, σ_l .

The mean value, $\bar{I}(i, j)$ needed for filtering the center pixel, $I(i, j)$ located at pixel position (i, j) of a local window of size $(2m + 1) \times (2n + 1)$ is computed as follows:

$$\bar{I}(i, j) = \frac{1}{(2n+1)(2m+1)} \sum_{k=i-n}^{n+i} \sum_{l=j-m}^{m+j} I(k, l) \quad (10.1.1.2)$$

Strength:

- This filter supports parallel processing of pixels which facilitates its direct implementation in various real-time applications.

Limitation:

- As this filter performs noise suppression in the homogeneous as well as in the heterogeneous regions of images identically, it often leads to undesired blurring of edges, target points and man-made structures.

Lee Sigma filter [186]: Like Lee filter, this filter too performs noise removal from images using MMSE technique as stated in (10.1.1.1) but in a slightly different way. It derives the mean and variance values needed for filtering each center pixel considering the local mean and variance values of neighboring pixels which lie within its two-sigma neighborhood $[I(i, j) - 2\sigma_v, I(i, j) + 2\sigma_v]$ only instead of considering the mean and variance of all the neighboring pixels as proposed in [185]. The authors have designed this filter assuming the distribution of noise in images is Gaussian in nature. They have stated that in the presence of Gaussian noise, 95.5% of the neighboring pixels which possess almost similar properties with that of the center pixel fall within the two-sigma range while other pixels

belonging to different population fall outside the range. Mathematically, $\bar{I}(i, j)$ is calculated in [186] using the following equation:

$$\bar{I}(i, j) = \frac{\sum_{k=i-n}^{n+i} \sum_{l=j-m}^{m+j} I(k, l) \delta(k, l)}{\sum_{k=i-n}^{n+i} \sum_{l=j-m}^{m+j} \delta(k, l)} \quad (10.1.1.3)$$

Where,

$$\begin{aligned} \delta(k, l) &= 1, \text{ if } I(i, j) - 2\sigma_v \leq I(k, l) \leq I(i, j) + 2\sigma_v \\ &= 0, \text{ otherwise} \end{aligned}$$

Although the use of the two-sigma neighborhood concept has improved the edge-preservation capacity of this filter compared to the Lee filter but the use of this concept disables this filter to suppress sharp spot noise present in images effectively as this type of noise mostly corrupts single pixel or two adjoining pixels.

To mitigate this drawback, the authors in [186] have introduced an additional step in their proposed filtering algorithm to determine the total number of pixels (M) lying within the two-sigma neighborhood range.

$\bar{I}(i, j)$ = two-sigma average, if $M > K$

$\bar{I}(i, j)$ = immediate neighbor average, if $M \leq K$

Although the introduction of this additional step has enabled this filter to suppress sharp spot noise to some extent but selecting an optimum value of the threshold (K) is a very challenging task as it plays a significant role in determining the filter's performance efficiency.

The authors in [186] have manually set the value of K lesser than 4 for a local window of dimension 7×7 and lesser than 3 for a local window of dimension 5×5 .

Strength:

- This filter is designed to preserve edges of images much more effectively compared to the traditional Lee filter [185] as it performs filtering of pixels considering only the mean and variance values of the selected neighboring pixels which possess almost similar properties.

Limitations:

- This filter performs noise removal assuming that the noise values are always Gaussian-distributed but in reality it is not so as the speckle noise mostly possesses Rayleigh distribution which lead to the introduction of a bias in the filtered data.
- This filter performs over-smoothing of target points.
- Manual selection of the threshold (K) often leads to undesired blurring of edges and erroneous noise suppression.
- Isolated dark pixels are not filtered effectively.

Enhanced Lee filter [187]: Ideally, the pixels belonging to the homogeneous and heterogeneous regions of images should be filtered using different strategies in order to achieve better noise suppression and edge-preservation as the properties of the pixels belonging to homogeneous and heterogeneous regions of images vastly differ from one another.

Neither the Lee filter [185] nor the Lee Sigma filter [186] were designed considering this logic which is the main cause behind undesired blurring of edges occurring in outputs obtained from these filters.

Enhanced Lee filter [187] is designed incorporating this logic which has enabled it to filter pixels belonging to the homogeneous and heterogeneous regions differently.

Mathematically, the noise suppression technique adapted by this filter is described as follows:

$$\hat{R}(i, j) = \begin{cases} \bar{I}(i, j) & \text{if } C_I \leq C_V \text{ (homogenous region)} \\ \bar{I}(i, j) + W(i, j)(\bar{I}(i, j) - I(i, j)) & \text{if } C_V < C_I < C_{max} \text{ (heterogenous region)} \\ I(i, j) & \text{if } C_I \leq C_{max} \text{ (Isolated points or Edges)} \end{cases} \quad (10.1.1.4)$$

Where,

$$W(i, j) = \exp\left(\frac{-K_{damp}(C_I - C_V)}{C_{max} - C_I}\right)$$

K_{damp} and C_{max} represent damping coefficient and the maximum value of the variance coefficient of noise-affected image respectively.

Strengths:

- This filter suppresses noise present in homogeneous and heterogeneous regions of images efficiently and preserves image details present in heterogeneous regions of images much more effectively compared to the Lee filter and Lee Sigma filter.
- It also efficiently preserves isolated points present in images.

Limitation:

- This filter often cannot satisfactorily suppress spot noise present in images as it often erroneously classifies spot noise as target points.

Improved Lee Sigma filter [188]: The filter performs noise removal similarly as proposed in the Lee Sigma filter [186] but using redefined sigma ranges. It addresses most of the shortcomings of the Lee Sigma filter and provides solutions for each of them as discussed below:

- Lee Sigma filter performs noise removal considering the noise values are always Gaussian-distributed and the mean is always equal to the value of the center pixel in a local window which often results in the introduction of bias in the filtered data as the actual noise values are mostly Rayleigh distributed. Moreover, the means of the noise probability density functions are not equal to the center pixels' values. Improved Lee Sigma filter solves this shortcoming by computing the refined sigma ranges $[I_1 \hat{\mu} \ I_2 \hat{\mu}]$ where $\hat{\mu}$ is the unique mean which is computed independently for each local window using a simple integration based approach.
- The authors in [188] have experimentally proved that filtering dark spot noise similarly as proposed in the Lee Sigma filter but using the computed mean, $\hat{\mu}$ instead of $\bar{I}(i, j)$ have also enabled their proposed filter to suppress the dark spot noise much more effectively compared to the Lee Sigma filter.
- In [188], the authors have also designed a novel target preservation approach to preserve the target points present in SAR images based on their observation that the back-scattered signals obtained from the target points are mostly dominated by a small number of strong scatters. Hence, the target points are usually represented by clusters of large number of bright pixels. The authors assumed that if in any local window there are more than T_k (threshold) number of pixels whose intensities are greater than 98th percentile (Z_{98}) of the intensities of all pixels belonging to that image, then those pixels are considered to be the target points and are preserved without any processing.

The pseudo-code of the filtering operation of the Improved Lee Sigma filter is given in Fig.10.1.1.1.

- Step 1: Compute 98th percentile (Z_{98}) of the intensities of all the pixels belonging to an image.
 Step 2: If the number of pixels possessing intensities higher than Z_{98} are greater than T_k within any local window of size 3×3 , then preserve those pixels as target points and go to Step 5. Else go to Step 3.
 Step 3: Compute MMSE for a 3×3 local window and compute its refined sigma range $[I_1 \hat{\mu} \ I_2 \hat{\mu}]$.
 Step 4: Performing filtering of pixels similarly as proposed in the Lee Sigma Filter but using refined sigma ranges.
 Step 5: Stop.

Fig.10.1.1.1. Pseudo-code of the filtering strategy adapted by the Improved Lee Sigma Filter

In [188], the authors have empirically chosen the range of T_k to be [5 7].

Strengths:

- This filter preserves target points and suppresses dark spot noise present in images much more effectively compared to the Lee Sigma filter.
- This filter facilitates the evaluation of unique sigma range for each local window in accordance to the probability density function of the noise distribution within that window. This eliminates any chances of the introduction of bias in the filtered data.

Limitation:

- This filter cannot preserve target points comprising of pixels lesser than T_k efficiently.

Fast Non-Local Lee filter [189] and Non-Local Lee filter [190]: The authors in [189] and [190] have designed hybrid filters by combining the structure similarity property of the Non-Local Means filter [191] and the homogeneous similarity property of the Lee filter.

Non-Local Means filter efficiently suppresses additive noise present in the structurally similar regions of images using a weighted averaging operator.

In traditional Non-Local Means filter [191], structural similarity among different image patches are detected by measuring the Euclidean distances among them. In [190], the authors have improved the structural similarity detection procedure by replacing the Euclidean distance metric by a new distance metric which can be approximated as a Chi square distribution with $n \times n$ degrees of freedom. The use of this new distance metric facilitates the formation of a collection, S_i of similar patches whose size is generally much larger compared to the size of patches.

In order to preserve target points and the homogeneous regions of images accurately, the authors in [189] have proposed a new filtering approach by substituting the weighted average filtering operation of the Non-Local Means filter [191] with the MMSE based filtering operation of the Lee filter. Mathematically, the filtering operation of the Fast Non-Local Lee filter is explained as follows:

$$\hat{R}_i = \bar{S}_i + \frac{(D(S_i) - \bar{S}_i^2 \sigma_s^2)}{(1 + \sigma_s^2)} (I_i - \bar{S}_i) \quad (10.1.1.5)$$

Where, \bar{S}_i is the average of the set, S_i and $D(S_i)$ is the variance of the set, S_i .

The authors in [189] have also proposed a direction projection method for performing point target preservation.

The main difference between the filters designed in [189] and [190] is that unlike [189], the authors in [190] have used Lee filter in distributive manner to perform filtering of pixels.

Strengths:

- The use of new distance metric and direction projection method in [189] and [190] has decreased their computational complexities by ten folds compared to the traditional Non-Local Means filter [191].
- The use of both the structural similarity property of the Non-Local Means filter and the homogeneity property of the Lee filter in the proposed Non-Local Lee filters has enabled these filters to perform both speckle noise suppression and edge preservation efficiently.

Limitation:

- These filters do not comprise of any additional measure to filter the dark spot noise present in images.

Refined Lee filter [192]: This filter performs speckle noise removal similarly as done in traditional Lee filter [185] but instead of considering the mean and variance values of all pixels located within a local window, the authors in [192] have performed filtering of pixels considering the mean and variance values of only the 'k' number of pixels located within a $K \times K$ neighborhood. They have used

K-Nearest Neighborhood (KNN) algorithm [152] for selecting these ' k ' pixels. KNN performs classification based on the assumption that similar data points should be located in close proximity and it selects ' k ' nearest neighbors of any sample data point using the Euclidean distance metric. In [192], KNN is used to select ' k ' nearest neighbors of each center pixel using the Euclidean distance metric assuming the properties of the pixels possess similar properties as that of $I(i, j)$.

Strength:

- The authors have experimentally proved that this filter can perform noise suppression and edge-preservation much more efficiently compared to the Lee filter when filtering is performed considering the mean and variance values of 6 nearest neighbors out of 9 neighbors located within a 3×3 neighborhood.

Limitations:

- KNN algorithm being a supervised classification algorithm requires a proper set of labeled data for training and validating the neural network designed for performing the selection of ' k ' number of nearest neighbors. Preparing such a set of labeled data in this case is very challenging.

Optimum selection of the local window size and the value of ' k ' depending upon the distribution of noise in images in order to achieve desired performance of the filter are also very challenging.

10.1.2. Median filter and its' proposed variants

Median Filter [193]: This filter is designed by the authors in [193] to primarily suppress the salt and pepper noise present in images. It performs filtering of the center pixel of any local window of size $(2n + 1) \times (2n + 1)$ with the median value of the intensities of all the pixels lying within that window.

Strength:

- Being a non-linear filter, this filter performs edge-preservation much more efficiently compared to the linear filters like mean (average) filter.

Limitation:

- This filter does not include any additional measures either to preserve target points or to suppress sharp spot noise present in images effectively.

Local Adaptive Median Filter [194]: This filter [194] is designed following the noise suppression strategies proposed in the Lee Sigma Filter [186] and the Local Sigma Filter [195].

Local Sigma Filter [195] performs noise suppression similarly as proposed in [186] but instead of using fixed sigma range, this filter performs noise suppression using flexible sigma range by introducing a user-defined variable, M whose value is either 1 or 2. The introduction of this user-defined variable facilitates the variation of the sigma range of each local window depending upon its' noise distribution. The flexible sigma range is defined as $[I(i, j) - M\sigma_V, I(i, j) + M\sigma_V]$.

Local Adaptive Median Filter performs noise suppression similarly as proposed in [195] but unlike [186] and [195], where the authors have filtered all the pixels, this filter performs noise suppression of the center pixel of any local window only when it is classified as a noisy pixel. For this purpose, prior to performing noise filtering , the authors in [194] classified the pixels into two classes namely, a. valid pixel and b. noisy pixel, using the following logic:

- If the center pixel of any local window lies within the computed sigma range, then it is classified as a valid pixel and is preserved without despleckling.

- b. If the center pixel of any local window lies outside the computed sigma range, then it is classified as a noisy pixel and is substituted with the median value of the pixels which lie within the sigma range, $[I(i, j) - M\sigma_V, I(i, j) + M\sigma_V]$.

Strength:

- The use of adaptive sigma range and the selective pixel filtering strategy has enabled this filter to perform noise suppression much more efficiently compared to the traditional Median filter [193].

Limitation:

- The performance efficiency of this filter is largely dependent on the optimum selection of M which is very challenging.
- This filter like Lee Sigma Filter often introduces bias in the filtered data as it performs despeckling of SAR and POLSAR images considering the noise values in these images are Gaussian distributed but in reality it is not so as it is stated in [188], that noise present in SAR and POLSAR images mostly follows Rayleigh distribution.

3D Median Filter with PSA [196] and 3D Median Filter with RPSA [197]:

In both [196] and [197], the authors introduced 3-D Median filters which suppresses speckle noise present in images having $N \times N \times N$ dimension using their proposed partial sort algorithm (PSA) and reverse partial sort algorithm (RPSA) respectively. The use of PSA and RPSA to sort the pixel intensities present in any local window in order to compute its median value has reduced the computational complexities of these filters largely compared to the computational complexity of a similar filter which performs sorting of pixels using the bubble sort algorithm. PSA and RPSA sort elements in a 2D array as stated in Fig.10.1.2.1 and Fig.10.1.2.2 respectively.

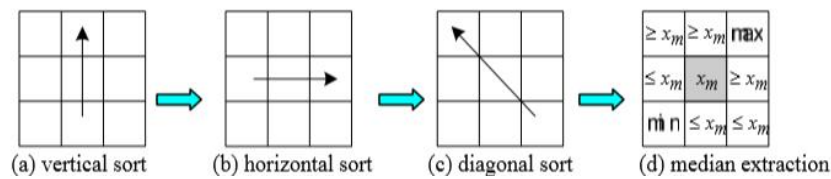


Fig.10.1.2.1. Sequence following which sorting is performed by PSA algorithm [197]

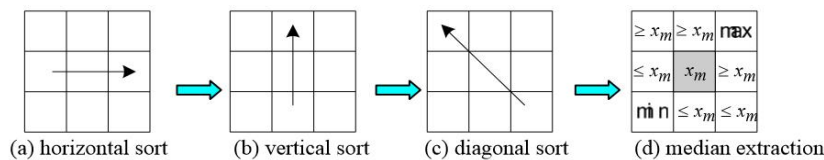


Fig.10.1.2.2. Sequence following which sorting is performed by RPSA algorithm [197]

In the above figures, x_m represents the median value of the pixels lying in the local window of size 3×3 whereas min and max signify the minimum and maximum intensities of the pixels located within that window.

In [196] and [197], the authors have performed despeckling of images using 3D sliding window which moves first along the x direction then along the y direction followed by the z direction as shown in the figure below:

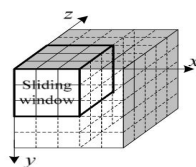


Fig.10.1.2.3. 3D sliding window [197]

When the sliding window moves by one position, N^2 samples of the old plane gets substituted by N^2 samples of new plane. The median value of a $3 \times 3 \times 3$ window is computed in [196] and [197] as depicted in the figure below:

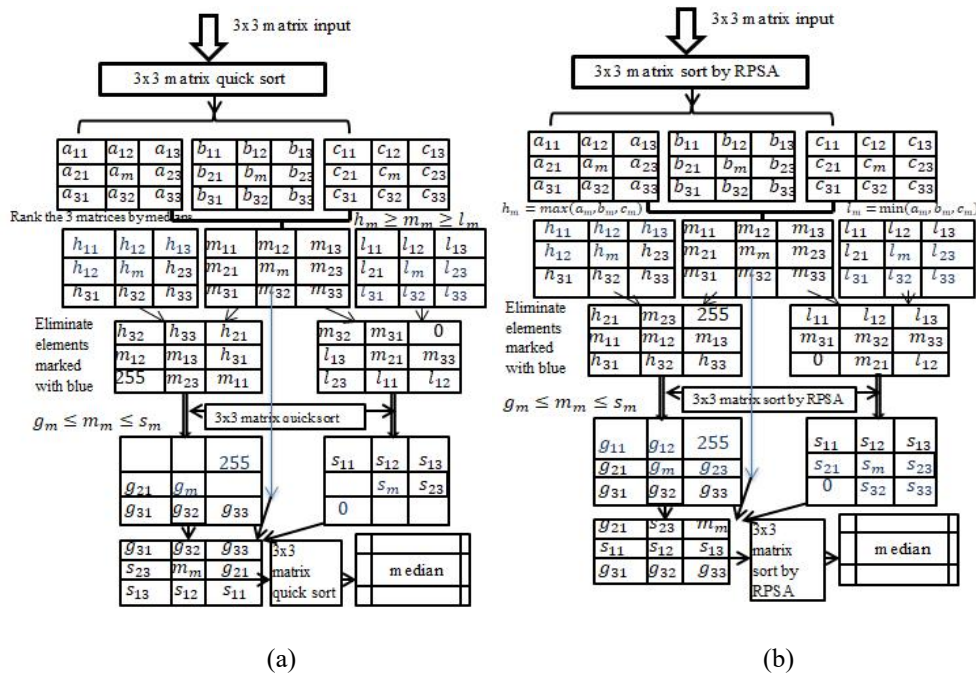


Fig.10.1.2.4. (a) and (b) Median value computation techniques proposed in [196] and [197] respectively.

Where (a, b, c) represent three 3×3 planes of an image and (h, l, m) represent median values of these planes where $h_m \geq m_m \geq l_m$.

Strength:

The use of PSA and RPSA for sorting elements has largely reduced the computational complexities of these filters as bubble sort algorithm requires 36 ‘compare and exchange’ operations to perform sorting of 9 elements in a 3×3 matrix whereas PSA and RPSA requires only 24 and 17 ‘compare and exchange’ operations to perform the same operation.

Limitation:

These filters have not adapted any additional measures to suppress sharp spot noise or preserve target points and edges present in images.

An Improved Median Filtering Algorithm Combined with Average Filtering [198] and An Improved Median Filtering Algorithm for Image Noise Reduction [199]: This filter [198] is designed by combining the properties of Mean (Average) Filter and Median Filter. It filters only the noisy pixels present in images using a sliding filter window which changes its size in accordance to the local distribution of noise in images considering PSNR [103] as an objective function which enhances this filter’s noise suppression capability by many folds compared to that of traditional Median filter [193].

The authors in [198] and [199] have designed these filters based on the assumption that noisy pixels differ largely from their neighboring pixels while all other pixels including the edge pixels possess almost similar properties as that of neighboring pixel. If the intensity of the center pixel of any local window is greater than the mean of all the pixels lying within that window, then it is considered as a

noisy pixel and is despeckled. Else it is considered as a valid pixel and is preserved without despeckling.

In [198], the authors have proposed a novel adaptive window concept which facilitates the adjustment of the size of the local window in accordance to the local distribution of noise in images having $M \times N$ dimension as discussed below:

Step 1. If the center pixel of any local window of size $(2n + 1) \times (2n + 1)$ is detected as a noisy pixel, then it is filtered similarly as proposed in [193].

Step 2. Calculate $PSNR_n$ value as described below:

$$Mean\ square\ error_n\ (MSE_n) = \frac{1}{M \times N} \sum_{i=1}^M \sum_{j=1}^N (I_{out}(i,j) - I(i,j))^2 \quad (10.1.2.1)$$

$$PSNR_n = 10 \log \left(\frac{a_{max}^2}{MSE} \right) \text{ dB} \quad (10.1.2.2)$$

Where, I_{out} : Filtered image, $a_{max} = 2^k - 1$, k : the number of bits used for pixel representation in an image.

Step 3. Then set $n = n + 2$, and filter I_{out} using the despeckling algorithm proposed in [193]. The output thus obtained is termed as I'_{out} .

Step 4. Calculate $PSNR_{n+1}$ value similarly as stated in (10.1.2.1) and (10.1.2.2) but replacing I with I_{out} and I_{out} with I'_{out} .

Step 5. If $PSNR_n > PSNR_{n+1}$, then consider the size of the window to be $(2n + 1) \times (2n + 1)$. Else go to Step 3 and further enlarge the size of the window.

The authors in [197] have proposed a similar despeckling algorithm but included a different adaptive window size selection procedure in their proposed filter which is defined as follows:

Step 1. Initialize: $n = 1$

Step 2. Compute: $A1 = med - min$ and $A2 = med - max$. (med, max and min represent the median, maximum and minimum values of the intensities of the pixels lying within any local window).

Step 3. If $A1 > 0$ and $A2 < 0$, then go to Step 4. Else ,enlarge the size of the window by setting $n = n + 1$ and go to Step 2.

Step 4. Substitute $I(i,j) = med$

Step 5. Stop.

In [197], the authors have also proposed a statistical histogram based median value computation procedure which decreases this filter's computational complexity from $O(N^2)$ to $O(N)$ where $O(N^2)$ denotes the computational complexity of the traditional Median filter [193].

Strength:

- The use of adaptive window concept has increased the noise suppression capabilities of these filters compared to the Median filter [193].

Limitation:

- These filters perform noise suppression in both homogeneous and heterogeneous regions identically which often lead to undesired edge-blurring.

10.1.3. Hybrid Median Filter and its proposed variants

This filter [198] performs noise suppression in each 5×5 local window of an image using the following steps:

Step 1: Pixels belonging to each 5×5 window are grouped into two groups where the first group comprises of the neighboring pixels located 90° angles to the center pixel. This group is termed as "+" sub-neighborhood while the second group comprises of the neighboring pixels located 45° angles to the center pixel. This group is termed as "x" sub-neighborhood. The median value of the first group is termed as m_+ and the median value of the second group is termed as m_x .

Step 2: The center pixel is substituted by the median of m_+ , m_x and $I(i, j)$.

The filtering operation proposed in [198] is pictorially represented as follows:

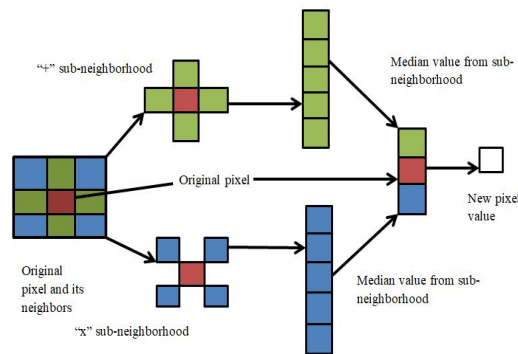


Fig.10.1.3.1. Pictorial representation of the filtering algorithm proposed in [198]

Modified Hybrid Median Filter [199]: This filter [199] performs noise suppression using similar despeckling algorithm proposed in [198] but modified the second step as follows:

Instead of replacing the value of the center pixel with the median of m_+ , m_x and $I(i, j)$, the authors have replaced it with the median of max_x , m_+ and $I(i, j)$ where max_x is the maximum value of the pixels lying in the window centered at pixel location, (i, j) .

Strength of the filters proposed in [198] and [199]:

Computational complexities of these filters are much lesser compared to that of the traditional Median filter [193].

Limitation of the filters proposed in [198] and [199]:

The use of sliding window of identical size to despeckle both homogeneous and heterogeneous regions of images by these filters often lead to the production of blurred images.

Adaptive window Hybrid Median Filter [200]: The authors in [200] have performed noise suppression similarly as proposed in [198] but with the introduction of a new adaptive window concept. They proposed that if the center pixel of any local window is detected as edge pixel, then it is filtered using the same algorithm proposed in [198] but using an local window of size 3×3 . Else, it is filtered using a local window of size 5×5 . The authors have used Sobel operator with the threshold value (100) to perform edge detection in their work.

Strength:

The use of sliding windows of different sizes by this filter to suppress noise in both the homogeneous and heterogeneous regions of images has somewhat improved its edge-preservation capability compared to the filters proposed in [198] and [199].

Limitation:

The use of Sobel operator by this filter for performing edge detection that too using a fixed threshold often leads to the production of erroneous results as Sobel operator cannot efficiently detect edges in noisy images (as discussed in **Section 3.1.2.6** of **Chapter 3**). This significantly influences the performance efficiencies of this filter.

10.1.4. Frost filter and its' proposed variants

Frost filter [201]: This filter is an exponentially weighted adaptive filter which performs speckle noise suppression in images using MMSE method. The noise suppression technique adapted by the Frost filter to suppress the speckle noise present at the center pixel located at pixel position, (i, j) of a processing window of size $(2n+1) \times (2n+1)$ is mathematically explained as follows:

$$\hat{R}(i, j) = \sum_{k=i-n}^{i+n} \sum_{l=j-n}^{j+n} h(k, l) I(k, l) \quad (10.1.4.1)$$

Where, $h(k, l) = K_1 e^{-K_{damp} C_l^2 (k, l) d_{kl}}$, K_1 : normalization coefficient, K_{damp} : damping (or tuning) factor, $C_l = \frac{\sigma_l}{\bar{I}}$: Local variation coefficient which is defined as the ratio of the standard deviation, σ_l and the mean, \bar{I} of the pixels which lie within the processing window centred at (i, j) .

$$d_{kl} = \sqrt{(i-k)^2 + (j-l)^2} \quad (10.1.4.2)$$

d_{kl} : Euclidean distance between the centre pixel located at pixel position, (i, j) and neighbouring pixel located at pixel position say (k, l) within the processing window centred at (i, j) . From (10.1.4.2), it becomes quite evident that the weight factor, $h(k, l)$ largely depends on the Euclidean distance, d_{kl} . It decreases when d_{kl} increases and vice versa.

Strength:

- Frost filter performs noise suppression using an exponentially damped weight factor whose value changes in accordance to the change in local statistics. This property of the Frost filter has somewhat improved its' performance efficiency compared to that of the linear denoising filters like average filter.

Limitations:

- The performance efficiency of the Frost filter is largely dependent on the optimum selection of the damping (or tuning) factor, K_{damp} as smaller K_{damp} facilitates better noise suppression but poor edge-preservation while larger K_{damp} facilitates better edge-preservation but poor noise suppression. Hence, performing noise suppression in an image using constant K_{damp} often leads to the production of undesired outputs. Ideally, K_{damp} should vary in accordance to variations of the statistics of pixels lying within the processing window to achieve better results but Frost filter does not support such adaptive variation of K_{damp} .
- The traditional Frost filter [201] is not a directional filter i.e. it an isotropic diffusion filter. This property of Frost filter is explained by considering a 2D sub-window comprising of 90° neighbors of $I(i, j)$ as follows:

$I(i-1, j-1)$	$I(i, j-1)$	$I(i+1, j-1)$
$I(i-1, j)$	$I(i, j)$	$I(i+1, j)$
$I(i-1, j+1)$	$I(i, j+1)$	$I(i+1, j+1)$

Fig.10.1.4.1. 2D sub-window comprising of 90° neighbors

$$W_{sub90^\circ} = \{(i-1, j), (i+1, j), (i, j-1), (i, j+1)\} \quad (10.1.4.3)$$

The distance between each 90° neighbor lying within W_{sub90° and the center pixel, $I(i, j)$ is evaluated as:

$$d_1 = \sqrt{(i-i+1)^2 + (j-j)^2} = 1 \quad (10.1.4.4)$$

$$d_2 = \sqrt{(i-i-1)^2 + (j-j)^2} = 1 \quad (10.1.4.5)$$

$$d_3 = \sqrt{(i-i)^2 + (j-j+1)^2} = 1 \quad (10.1.4.6)$$

$$d_4 = \sqrt{(i-i)^2 + (j-j-1)^2} = 1 \quad (10.1.4.7)$$

Since, $d_1 = d_2 = d_3 = d_4$, it proves that the Frost filter is not a directional filter. Hence, it cannot preserve edges within a processing window efficiently.

- The use of fix sized processing window to perform noise suppression often leads to the production of undesired results as processing window of large size can efficiently suppress noise in homogeneous regions of images but cannot preserve edges in heterogeneous regions effectively. Similarly, processing window of small size cannot perform noise suppression in homogeneous regions satisfactorily.

Enhanced Frost filter [202]: In order to achieve better edge-preservation and noise suppression, ideally the pixels belonging to the homogeneous regions and heterogeneous regions of images should be filtered using different techniques as the properties of these regions significantly differ from one another and the size of the processing window and damping (or tuning) factor should change in accordance to variations in distribution of speckle noise and local characteristics of the pixels lying within each processing window.

Frost filter proposed in [201] performs noise suppression of pixels belonging to different regions of images identically using a fixed size processing window and a constant K_{damp} .

Moreover, it also does not include any additional measure to preserve edges and target points present in images which often lead to the production of undesired blurred outputs. To mitigate these drawbacks, the authors in [202] have designed the Enhanced Frost filter which performs noise removal from pixels belonging to different regions using varied techniques and modified weight factor $h'(k, l)$ but like Frost filter [201], this filter too performs noise suppression using a processing window of fixed size and a constant K_{damp} as defined below:

$$\hat{R}(i, j) = \begin{cases} \bar{I} & \text{if } C_I \leq C_V (HmR) \\ I(i, j) * h'(i, j) & \text{if } C_V < C_I < C_{\max} (HtR) \\ I(i, j) & \text{if } C_V \leq C_{\max} (E \& IP) \end{cases} \quad (10.1.4.8)$$

Where HmR: Homogeneous Regions, HtR: Heterogeneous Regions and E&IP: Edges & Isolated points.

$$h'(k, l) = e^{-\frac{K_{damp}(C_I(k,l) - C_V(k,l))}{C_{\max} - C_I(k,l)} d_{kl}} \quad (10.1.4.9)$$

$C_V = \frac{\sigma_V}{V}$: Coefficient of variation of an image corrupted with noise having standard deviation, σ_V

and mean \bar{V} . C_{\max} : Maximum value of the local coefficient variation of a noise-affected image.

Strength:

- This filter performs noise suppression much more efficiently compared to the traditional Frost filter proposed in [201] as it suppresses speckle noise present in pixels belonging to heterogeneous regions and homogeneous regions of images using different procedures and also preserves pixels which are identified as target points or edges without any despeckling which prevents undesired blurring of those structures.

Limitation:

- This filter like Frost filter [201] too performs noise suppression using constant K_{damp} and fixed size processing window which often introduces unwanted artifacts in the filtered outputs.

Anisotropic Diffusion Frost filter(ADFF) [203]: The authors in [203] have designed ADFF by introducing the edge-sensitivity property in the traditional Frost filter model [201] following the concepts of Anisotropic Diffusion proposed in [204]. They have defined eight possible edge-directions (shown in Fig.10.1.4.2) within each 5×5 processing window and performed edge-detection along those directions using ratio detector [205].

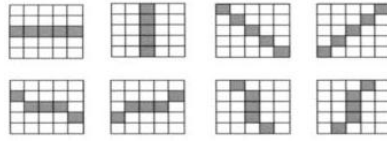


Fig. 10.1.4.2. Eight possible edge directions in each 5×5 processing window (Shaded areas represent possible edge directions) [203]

The ratio detector values (r_d) are defined as the ratios of the arithmetic averages of the pixels lying in the non-overlapping neighborhoods on the opposite sides of the possible edge points (which are marked as shaded areas).

$$r_{max} = \max(r_d), \text{ for } d = 1, 2, \dots, 8 \quad (10.1.4.10)$$

$$r_{min} = \min(r_d), \text{ for } d = 1, 2, \dots, 8 \quad (10.1.4.11)$$

The authors have modified the filtering mechanism proposed in [201] as stated in (10.1.4.12) in such a way that if edge exists in any of the possible edge-directions (shown in Fig.10.1.4.2), then the coefficients of the weight factor in that direction will be more significant compared to other areas.

$$\hat{R}(i, j) = \sum_{k=i-n}^{i+n} \sum_{l=j-n}^{j+n} \text{div}(h(k, l)).I(k, l) \quad (10.1.4.12)$$

Where $\text{div}()$: Divergence operator.

In [203], the authors have evaluated the coefficients of the weight factor depending upon the values of five variables namely, $r_d, r_{max}, r_{min}, T_l$ and T_h (where T_l and T_h are pre-determined thresholds ($0 < T_l < T_h < 1$)) as follows:

1. If $r_{min} < T_l$, it suggests the presence of a strong edge in the processing window and the direction of edge is $d_{sub} \in d$. Then the coefficients of weight factor are evaluated as follows:

$$\text{div}(h(k, l)) = \begin{cases} 0, & \text{if } (k, l) \notin W_{d_{sub}} \\ 1, & \text{if } (k, l) \in W_{d_{sub}} \end{cases}$$

So, the output of ADFF is the average value of the pixels lying along the edge direction.

2. In homogeneous areas, $r_{min} > T_h$. Then $\text{div}(h(k, l)) = 1, \text{ for all } (k, l)$. In this case, ADFF acts a average filter.

3. If $T_l < r_{min} < T_h$,

$$div(h(k, l)) = \left(1 - \frac{r_d - r_{min}}{r_{max} - r_{min}}\right) h_0(k, l), (k, l) \in W_{d_{sub}}$$

This suggests that edge directions having smaller r_d , will possess more significant weight coefficients.

Strength:

- The edge sensitivity property of this filter has enabled it to perform edge-preservation and enhancement efficiently.

Limitations:

- Like Frost filter [201] and Enhanced Frost filter [202], ADFE too performs noise suppression using a constant K_{damp} and a processing window of fixed size which often leads to the production of undesired outputs.
- The performance efficiency of this filter is largely dependent on the optimum selection of the two thresholds (T_l and T_h) values which is very challenging.

Modified Frost filter based on adaptive tuning factor [206]: This filter facilitates the variation of K_{damp} for each processing window of dimension $(2n + 1) \times (2n + 1)$ in accordance to the extent of noise corruption of the centre pixel, $I(i, j)$. The authors have determined the noise content of the centre pixel by estimating its' significance using the t-statistics method. The adaptive tuning (or damping) factor (K'_{damp}) is computed in [206] as discussed below:

$$K'_{damp}(s, h) = T(i, j).Q(s, h) \quad (10.1.4.13)$$

Where,

$$T(i, j) = \frac{|I(i, j) - \bar{I}(i, j)|}{\sigma(i, j)}$$

$$Q(s, h) = \frac{|I(s, h) - I(i, j)|}{\frac{1}{\{(2n + 1)(2n + 1)\} - 1} \sum_{k=i-m}^{i+m} \sum_{l=j-n}^{j+n} |I(k, l) - I(i, j)|}$$

$K'_{damp}(s, h)$: adaptive tuning factor, $T(i, j)$: t statistics value, $Q(s, h)$: gray characteristics of neighboring pixels, $\sigma(i, j)$: standard deviation of the window centered at (i, j) . This filter performs noise suppression similarly as Frost filter [203] but using K'_{damp} instead of K_{damp} .

Strength:

- This filter allows the tuning (or damping) factor to vary in accordance to the noise corruption of the centre pixel which reduces the chances of producing blurred outputs.

Limitations:

- This filter is also an isotropic diffusion filter like Frost filter [201] and Enhanced Frost filter [203]. Hence, it often cannot preserve edges satisfactorily.
- The drawbacks arising in the Frost filter models discussed above due to the use of processing window of fixed size to perform noise suppression too exists in this filter.

Double Adaptive Frost filter(DAFF) [207]: The authors in [207] have designed DAFF to perform speckle noise suppression using adaptive window and adaptive tuning (or damping) factor which improve the edge-preservation and noise suppression capabilities of this filter compared to the previously proposed variants of the Frost filter.

DAFF performs adaptive variation of K'_{damp} using the same method proposed in [206] and it varies the size of the processing window in accordance to the variations in properties of the boundary pixels of the current processing window instead of taking into account the properties of all neighbouring pixels to reduce the computational load as discussed below:

$$W(i, j) = \begin{cases} \min [W(i, j) + 2, W_{max}], & \text{if } C'_l(i, j) \leq T'(i, j) \\ \max [W(i, j) - 2, W_{min}], & \text{if } C'_l(i, j) > T'(i, j) \end{cases} \quad (10.1.4.14)$$

$$C'_l(i, j) = \frac{\sigma'_l}{\bar{I}'} \quad (\text{modified local variation coefficient}) \quad (10.1.4.15)$$

Where, σ'_l and \bar{I}' are mean and standard deviation of the boundary pixels of the processing window centred at (i, j) .

$$T'(i, j) = \eta \left(1 + \sqrt{\frac{1+2\sigma_F^2}{8(W(i, j)-1)}} \right) \cdot \sigma_F \quad (\text{threshold}) \quad (10.1.4.16)$$

Where,

σ_F : speckle variation coefficient and $W(i, j)$: processing window centered at (i, j) .

W_{min} and W_{max} are minimum and maximum allowable window sizes. DAFF performs noise suppression considering the initial size of the processing window to be equal to W_{min} and then vary its' size as stated in (10.1.4.16). DAFF filters pixels belonging to homogeneous regions ($C'_l(i, j) < \sigma_F$) and heterogeneous regions ($C'_l(i, j) \geq \sigma_F$) using Box filter and the filtering method proposed in [206] respectively.

Strength:

- This filter performs noise suppression using adaptive tuning factor and processing window of adaptive size which enable it to minimize the chances of producing blurred outputs.

Limitation:

- This filter is also an isotropic diffusion filter.

Most of the filters whose noise suppression strategies are discussed in **Section 10.1** perform despeckling of images using manually selected thresholds which often lead to the production of undesired outputs. To mitigate these drawbacks of existing filters, a novel Degree of Purity & Scattering diversity based Advanced Lee filter is designed in this thesis taking into account the variations in the scattering properties of the center pixel and its' neighbouring pixels located within a pre-defined neighborhood. Detailed discussion on the noise suppression strategy adapted in the designed filter is carried out in **Chapter 11**. The comparative qualitative and quantitative analyses' results obtained by applying the designed filter as well as several existing filters on similar set of data are also given in **Chapter 11** to the prove the efficiency of the designed filter over existing filters.

10.2. Existing CNN based POLSAR image classification methods

In the era of deep learning, various researchers have designed several deep neural networks to classify satellite images but those networks are mostly focused on performing SAR image classification. Very few deep neural networks are designed to classify POLSAR images till date. As this thesis mainly

focuses on POLSAR image classification, so the literature survey is focused on the architectures of deep neural networks designed to perform POLSAR image classification.

Lenet 5 [208] is considered to be one of the most suitable choice to perform POLSAR image classification by the authors in existing works. This network comprises of two convolutional layers, two fully connected layers, two pooling layers and one softmax layer. The authors in [209] have performed POLSAR image classification using Lenet 5 network in their work. The architectures of the deep neural networks designed in [210] and [211] to perform POLSAR image classification also highly resembles the architecture of Lenet 5 network.

The authors in [212] have performed POLSAR image classification using ResNet architecture [213] based on information derived from several decomposition methods' based features which are extracted using PolSARpro software [214]. After performing extraction of these features, the authors have performed feature selection using a parameter namely, S which is designed to measure the separability among features. The authors in [212] have also divided the pixels in an POLSAR image into several super-pixels using SLIC technique [86] prior to performing feature extraction and performed classification based on the information derived from super-pixels which improves the method's efficiency.

The authors in [215] have designed their network, fixed feature size CNN (namely, FFS-CNN) to perform multi-pixel classification at a time i.e. to classify all pixels located within a patch simultaneously by taking into account the similarities in the properties of the pixels located within an image patch. The feature extraction part of this network comprises of four convolutional layers, two fully connected layers and one reshape layer. The architecture of this part of the network is designed after taking inspiration from the architecture of Lenet 5 network because of its' popularity in this field. Unlike Lenet 5 architecture, the architecture of FFS-CNN is devoid of any pooling layers as in FFS-CNN, image patches of dimension $(15 \times 15 \times 9)$ is given as inputs to the network, so the introduction of pooling layers after convolutional layers in this case often lead to information loss. FFS-CNN performs multi-pixel classification simultaneously unlike Lenet 5 network. This part of the architecture is designed after taking inspiration from Fully Connected Network (FCN) [216]. FFS-CNN uses softmax classifier to perform classification.

In this thesis, a novel deep-neural network namely, Crop-Net is designed to perform classification of croplands growing multiple types of crops using a selected set of features. Crop-Net is designed to perform simultaneous classification of multiple pixels located within a patch after taking inspiration from FFS-CNN architecture. Detailed discussion on the architecture of Crop-Net is carried out in **Chapter 12** and the improvement in performance achieved by applying Crop-Net to perform classification of croplands compared to Lenet 5 and FFS-CNN in terms Precision, Recall, F1score and Test Accuracy is shown in the same Chapter. A novel feature selection method namely, Ranking method is designed in this thesis to perform selection of the most significant features from a set of extracted features to obtain a trade-off between computational burden and test accuracy. The proposed feature selection method is also explained in **Chapter 12**.

Chapter 11

Detailed discussion on the filtering strategy of the proposed Degree of purity & Scattering diversity based Advanced Lee filter is carried out in this Chapter and qualitative as well as quantitative results obtained from the comparative analyses performed to prove the excellence of the proposed filter over existing works are also given in this Chapter.

11.1. Filtering mechanism of the proposed filter

Being a modified version of the Lee filter [185], the proposed filter performs despeckling of images similarly as done in Lee filter but using weighted local statistics values i.e. weighted values of mean and standard deviation of pixels lying within a patch. These weights are computed in accordance to the similarities in scattering properties of neighbouring pixels and centre pixel of a patch which is to be filtered. The scattering properties are estimated using degree of purity and scattering diversity values. The use of these weights in performing despeckling of images helps in giving the neighboring pixels possessing similar scattering properties with that of the centre pixel more importance while performing despeckling compared to other less similar neighboring pixels. The weight assigned to each pixel is computed based on its similarity with the centre pixel and thus helps in reducing the chances of producing undesired outputs.

The details of each step of the proposed Degree of purity & Scattering Diversity based Advanced Lee filter is given below:

Step 1: In this step, each channel of POLSAR image is divided into 5×5 overlapping patches.

Pictorial representation of an arbitrary patch for e.g., ' i^{th} ' patch of any channel obtained after division is given below:

$(x-2, y-2)$	$(x-1, y-2)$	$(x, y-2)$	$(x+1, y-2)$	$(x+2, y-2)$
$(x-2, y-1)$	$(x-1, y-1)$	$(x, y-1)$	$(x+1, y-1)$	$(x+2, y-1)$
$(x-2, y)$	$(x-1, y)$	(x, y)	$(x+1, y)$	$(x+2, y)$
$(x-2, y+1)$	$(x-1, y+1)$	$(x, y+1)$	$(x+1, y+1)$	$(x+2, y+1)$
$(x-2, y+2)$	$(x-1, y+2)$	$(x, y+2)$	$(x+1, y+2)$	$(x+2, y+2)$

Fig.11.1.1. 5×5 image patch

Step 2: In this step, degree purity and scattering diversity values for a POLSAR image is generated using PolSARpro software [214]. These parameters give estimation of the scattering properties of the target area from which electromagnetic waves are reflected back to SAR. Hence, these parameters help in the identification of the type of land cover present in the target area as reflection property of target areas vary in accordance to the type of land cover present in those areas. The generated degree purity and scattering diversity files are two-dimensional in nature whose height and width are equal to the height and width of POLSAR image for which these files are generated. After the generation of these files, they are also divided into 5×5 overlapping patches.

At the end of this step, a 5×5 patch containing the degree purity values and a 5×5 patch containing the scattering diversity values are obtained for each $5 \times 5 \times 3$ patch obtained from a POLSAR image i.e. degree purity and scattering diversity values of a particular pixel position is considered to be constant for all three channels of a POLSAR image.

So the ' i^{th} ' patch generated from 'degree purity' file and the ' i^{th} ' patch generated from 'scattering diversity' file will correspond to the ' i^{th} ' patch generated from all the three color channels of a POLSAR image.

Step 3: In this step, the extent of similarity between the degree purity and scattering diversity values of the centre pixel of any patch w.r.t each of its' neighbouring pixels is estimated. Depending upon the extent of similarity between the center pixel w.r.t each of its' neighboring pixel, a weight value is computed for each pixel. . These weights are computed using a set of 25 novel FI rules which are given in Table 11.1.1. Graphical representations of fuzzy membership functions (Triangular and Gaussian) used to represent the linguistic variables of FI rules are given in Fig.11.1.2 and Fig.11.1.3 respectively.

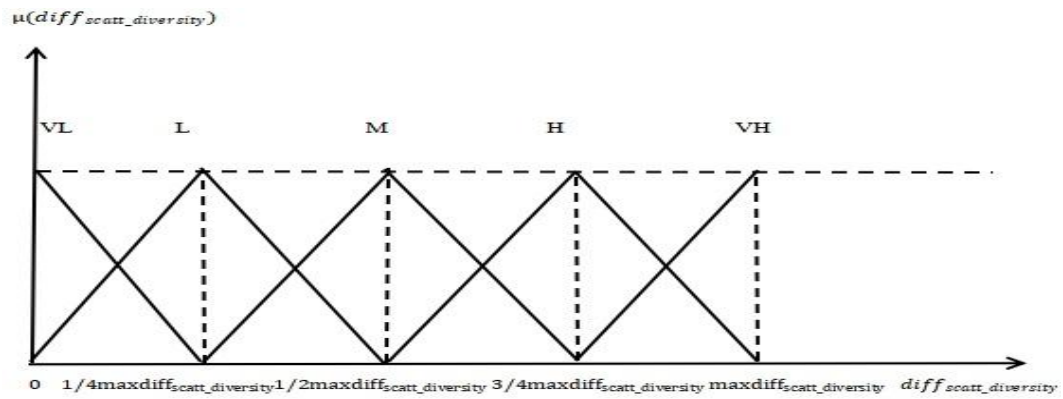
Input linguistic variables of the designed FI rules: $diff_{scatt_diversity}$, $diff_{degree_purity}$

Output linguistic variable of the designed FI rules: $w(weight)$

Table 11.1.1. Novel FI rules designed to perform weight estimation

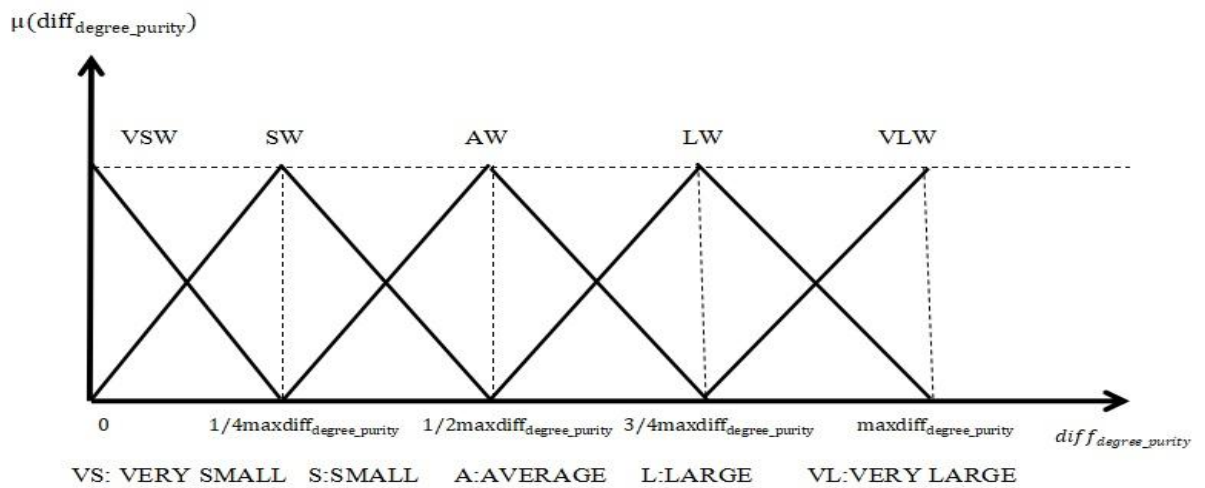
Code	IF		THEN
Rule	$diff_{scatt_diversity}$	$diff_{degree_purity}$	w
1	VERY LOW	VERY SMALL	VERY LARGE WEIGHT
2	VERY LOW	SMALL	VERY LARGE WEIGHT
3	VERY LOW	AVERAGE	LARGE WEIGHT
4	VERY LOW	LARGE	AVERAGE WEIGHT
5	VERY LOW	VERY LARGE	SMALL WEIGHT
6	LOW	VERY SMALL	VERY LARGE WEIGHT
7	LOW	SMALL	LARGE WEIGHT
8	LOW	AVERAGE	AVERAGE WEIGHT
9	LOW	LARGE	AVERAGE WEIGHT
10	LOW	VERY LARGE	SMALL WEIGHT
11	MEDIUM	VERY SMALL	LARGE WEIGHT
12	MEDIUM	SMALL	LARGE WEIGHT
13	MEDIUM	AVERAGE	AVERAGE WEIGHT
14	MEDIUM	LARGE	SMALL WEIGHT
15	MEDIUM	VERY LARGE	SMALL WEIGHT
16	HIGH	VERY SMALL	AVERAGE WEIGHT
17	HIGH	SMALL	AVERAGE WEIGHT
18	HIGH	AVERAGE	SMALL WEIGHT
19	HIGH	LARGE	SMALL WEIGHT
20	HIGH	VERY LARGE	VERY SMALL WEIGHT
21	VERY HIGH	VERY SMALL	AVERAGE WEIGHT
22	VERY HIGH	SMALL	AVERAGE WIGHT
23	VERY HIGH	AVERAGE	SMALL WEIGHT
24	VERY HIGH	LARGE	VERY SMALL WEIGHT
25	VERY HIGH	VERY LARGE	VERY SMALL WEIGHT

Case 1:



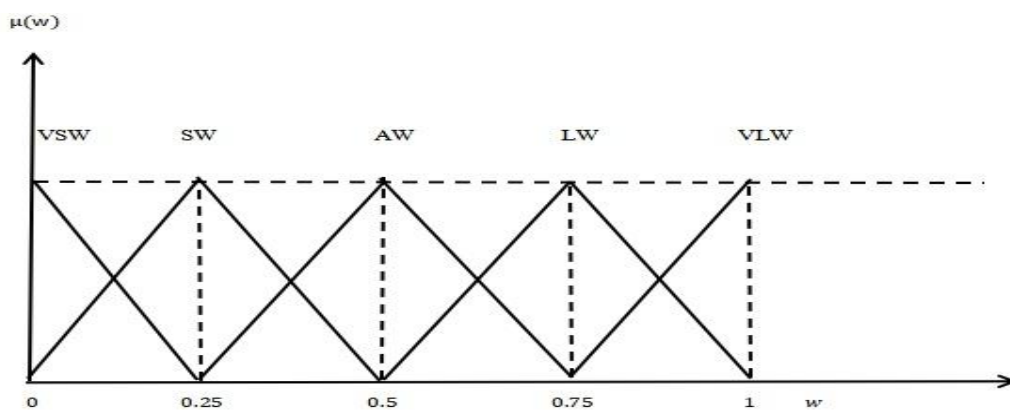
VL: VERY LOW
 L: LOW
 M: MEDIUM
 H: HIGH
 VH: VERY HIGH

(a)



VS: VERY SMALL S: SMALL A: AVERAGE L: LARGE VL: VERY LARGE

(b)

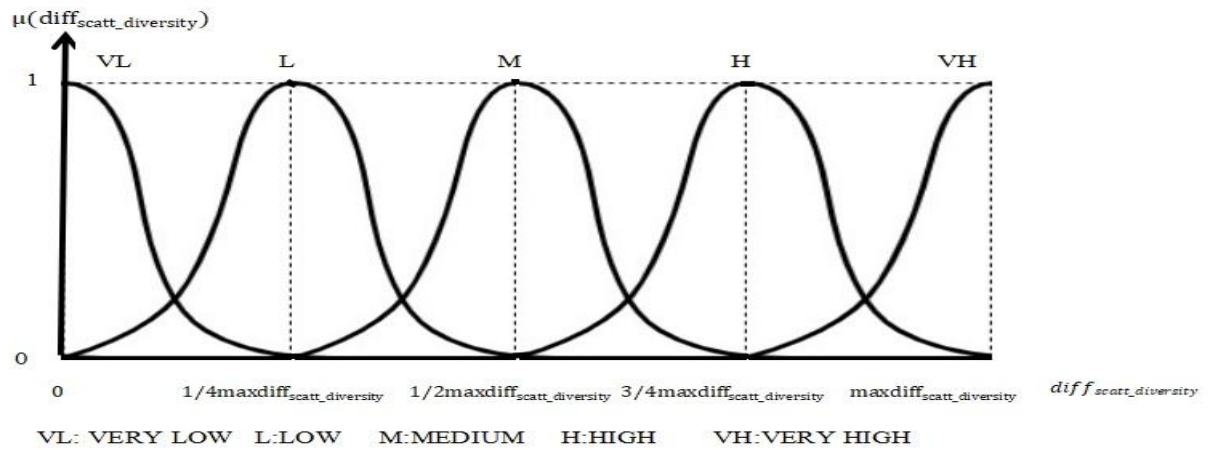


VSW: VERY SMALL WEIGHT
 SW: SMALL WEIGHT
 AW: AVERAGE WEIGHT
 LW: LARGE WEIGHT
 VLW: VERY LARGE WEIGHT

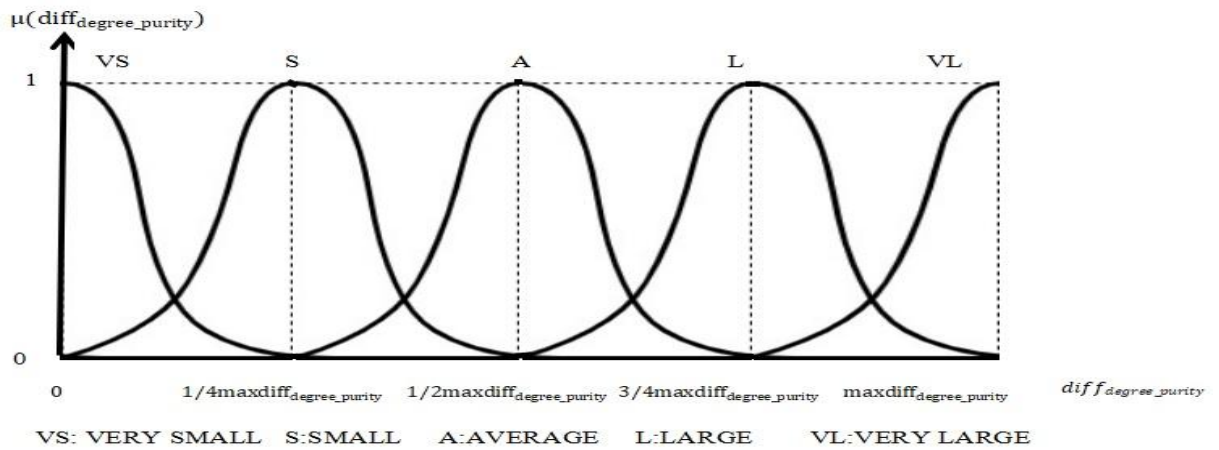
(c)

Fig.11.1.2. Graphical representation of fuzzy membership function (Triangular) used to represent (a) $diff_{scatt_diversity}$, (b) $diff_{degree_purity}$ and (c) w

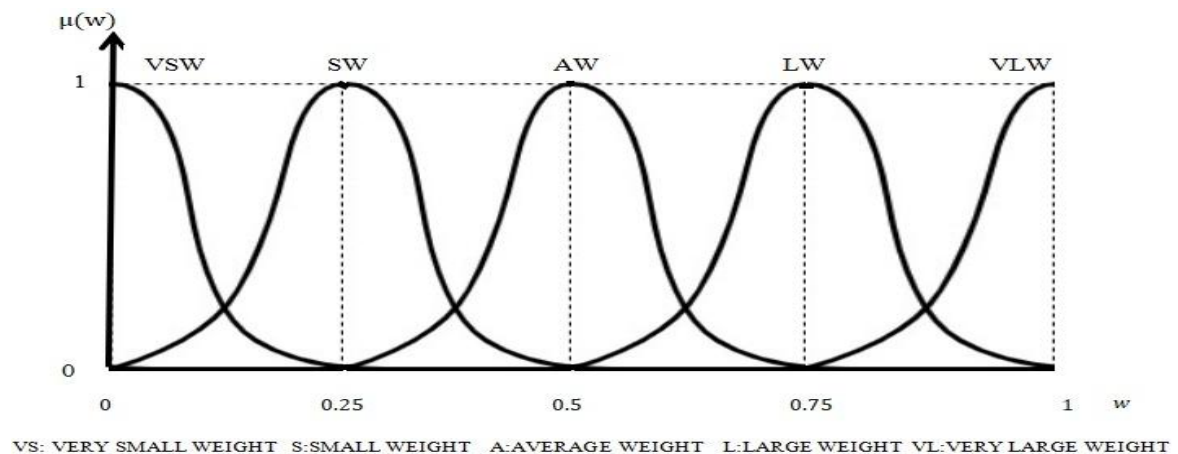
Case 2:



(a)



(b)



(c)

Fig.11.1.3. Graphical representation of fuzzy membership function (Gaussian) used to represent (a) $diff_{scatt_diversity}$, (b) $diff_{degree_purity}$ and (c) w

**Note: In this work, despeckling operation is carried out twice using Triangular membership function (given in Fig.11.1.2 (Case a)) and Gaussian membership function (given in Fig.11.1.3 (Case b)) respectively to obtain the best results.*

Dynamic Range (DR) of linguistic variable, $diff_{scatt_diversity} = [0 \quad maxdiff_{scatt_diversity}]$

$$(maxdiff_{scatt_diversity} = |\max(scatt_diversity) - \min(scatt_diversity)|)$$

DR of linguistic variable, $diff_{degree_purity} = [0 \quad maxdiff_{degree_purity}]$

$$(maxdiff_{degree_purity} = |\max(degree_purity) - \min(degree_purity)|)$$

DR of linguistic variable, $w = [0 \quad 1]$

$\max(scatt_diversity)$ and $\min(scatt_diversity)$ represent the maximum and minimum values of the generated 2D scattering diversity file whose height and width are equal to the height and width of the corresponding POLSAR image. Similarly, $\max(degree_purity)$ and $\min(degree_purity)$ also represent the maximum and minimum values of the generated 2D degree purity file.

The input linguistic variables, $diff_{scatt_diversity}$ and $diff_{degree_purity}$ are computed using the following mathematical equations:

$$diff_{scatt_diversity} = |scatt_diversity(x, y) - scatt_diversity(i, j)| \quad (11.1.1)$$

$$diff_{degree_purity} = |degree_purity(x, y) - degree_purity(i, j)| \quad (11.1.2)$$

Where,

$scatt_diversity(x, y)$: Scattering diversity value of the center pixel of any patch which is centered at pixel position, (x, y) .

$scatt_diversity(i, j)$: Scattering diversity value of any neighboring pixel located at pixel position say, (i, j) within a patch centered at pixel position, (x, y) .

$degree_purity(x, y)$: Degree purity value of the center pixel of any patch which is centered at pixel position, (x, y) .

$degree_purity(i, j)$: Degree purity value of any neighboring pixel located at pixel position say, (i, j) within a patch centered at pixel position, (x, y) .



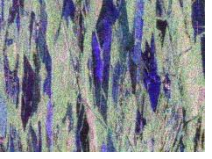


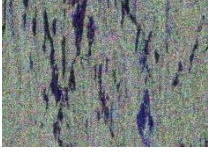
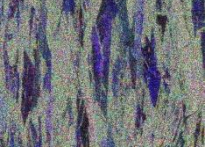



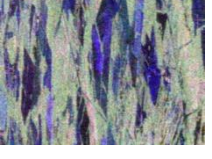


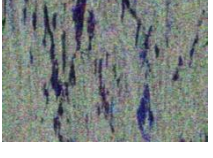
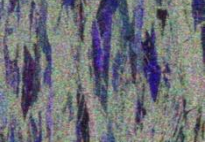

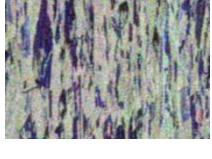

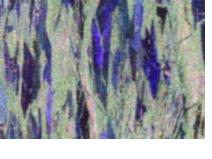

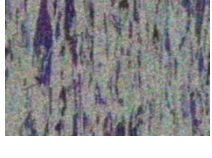

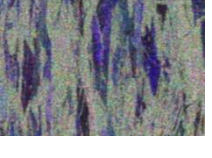

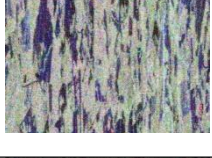

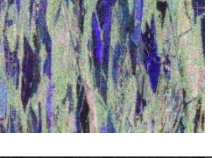

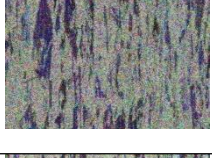


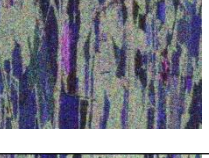








The use of these parameters as input linguistic variables of the FI rules designed for performing weight estimation enables one to assign a weight to each neighboring pixel depending upon its' similarities in scattering properties with the center pixel of the patch in which it is located.

At the end of this step, a defuzzified weight value (w^*) is obtained for each pixel present in an image.

Step4: In this step, despeckling of the centre pixel of each patch is performed similarly as done in Lee filter but using weighted mean and weighted standard deviation values of pixels lying within the pre-defined neighbourhood of that centre pixel.

The adaptation of this type of despeckling strategy enables this filter to produce better results compared to existing filters as proved by the qualitative and quantitative comparative analyses' results given in Table 11.1.2 and Table 11.1.3 respectively.

Table 11.1.2. Comparative qualitative analysis results

Image No	1	2	3	4
Original image				
Noisy image				
Original image filtered using Frost filter [201]				
Noisy Image filtered using Frost filter [201]				
Original image filtered using Lee filter [185]				
Noisy Image filtered using Lee filter [185]				
Original image filtered using Improved Lee Sigma filter [188]				
Noisy Image filtered using Improved Lee Sigma filter [188]				
Original image filtered using Non-Local Means filter [191]				
Noisy Image filtered using Non-Local Means filter [191]				


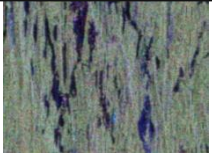
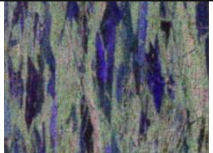

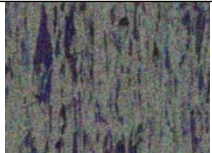
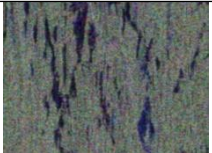
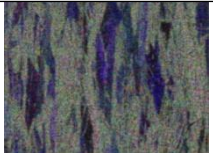

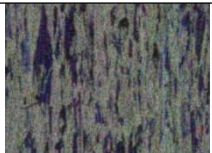
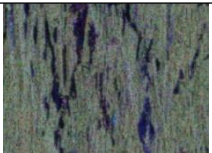
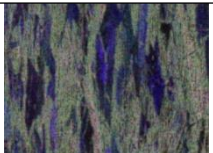

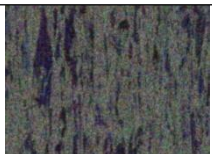
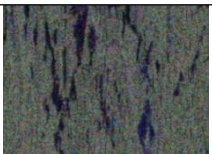
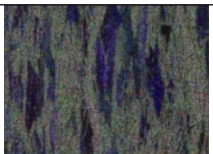
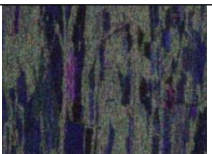
Original image filtered using designed filter (using Triangular membership function)				
Noisy Image filtered using designed filter (using Triangular membership function)				
Original image filtered using designed filter (using Gaussian membership function)				
Noisy Image filtered using designed filter (using Gaussian membership function)				

Table 11.1.3. Comparative quantitative analysis results

Image No	1	2	3	4
Frost filter	SSIM: 0.5106 PSNR:17.3059 EPI: 0.8992	SSIM: 0.5878 PSNR:67.0606 EPI: 0.8618	SSIM: 0.6393 PSNR:67.2868 EPI: 0.9245	SSIM: 0.6946 PSNR:67.9495 EPI:0.9547
Lee filter	SSIM:0.6322 PSNR:67.3958 EPI:0.9254	SSIM:0.6338 PSNR:68.3973 EPI:0.8812	SSIM:0.6892 PSNR:69.0192 EPI: 0.9469	SSIM: 0.7434 PSNR:70.3621 EPI: 0.9691
Kuan filter	SSIM: 0.4164 PSNR: 61.9480 EPI: 0.7563	SSIM: 0.4164 PSNR:62.1677 EPI: 0.7618	SSIM: 0.4359 PSNR:62.4248 EPI: 0.7952	SSIM: 0.4859 PSNR:62.8963 EPI:0.8208
Non-Local Means filter	SSIM: 0.4224 PSNR:59.9762 EPI:0.7346	SSIM: 0.4262 PSNR:60.0509 EPI:0.7312	SSIM: 0.4605 PSNR:60.4597 EPI:0.7531	SSIM: 0.4870 PSNR:61.0194 EPI: 0.7739
Designed filter (using Triangular membership function)	SSIM: 0.6983 PSNR:69.4393 EPI: 0.9203	SSIM: 0.6947 PSNR:69.9406 EPI:0.8932	SSIM: 0.7460 PSNR:70.3627 EPI: 0.9482	SSIM:0.7931 PSNR: 71.2970 EPI: 0.9685
Designed filter (using Gaussian membership function)	SSIM: 0.7489 PSNR: 70.9632 EPI: 0.9274	SSIM: 0.7398 PSNR: 71.3747 EPI: 0.9013	SSIM: 0.7871 PSNR: 71.9055 EPI: 0.9502	SSIM: 0.8250 PSNR: 72.9799 EPI: 0.9666

The details of the quantitative parameters, SSIM and PSNR used for performing quantitative analysis in this work are given in **Chapter 4**.

Alongside these quantitative parameters, Edge Preservation Index (EPI) [217] is used here to perform quantitative analysis. This parameter estimates the edge-preservation ability of filters. Hence, high value of this parameter suggests better result. ALOS PALSAR images used for performing comparative analyses are downloaded from the site mentioned in [218].

The consideration of weighted mean and weighted standard deviation values of any patch while performing despeckling of center pixel of that patch has improved the performance of the proposed Degree of purity & Scattering diversity based Advanced Lee filter as the computed weights help in estimation of the similarities between scattering properties of each of the neighboring pixels of any patch and the pixel on which the patch is centered (centre pixel). The improvement in the performance of the proposed filter achieved over existing filters by the use of these weighted statistics values while performing despeckling is also validated both qualitatively as well as quantitatively.

Chapter 12

In this Chapter, detailed discussion is carried out on the architecture of Crop-Net, the deep neural network which is designed to perform multi-class cropland (land growing multiple types of crops) classification in this thesis. The details of Ranking method, a novel feature selection method which is proposed in this thesis is also given in this Chapter. Improvement in performance achieved due to the use of Ranking method in combination with Crop-Net to perform classification over existing networks is also shown in terms of Precision, Recall, F1 score and Test Accuracy in this Chapter.

12.1. Crop-Net architecture

This network is designed in this thesis inspired by the architecture of FFS-CNN [215]. Like FFS-CNN, Crop-Net also performs classification of multiple pixels present within an image patch simultaneously by exploiting the similarities in their properties. The architecture of Crop-Net is shown in Fig.12.1.1.

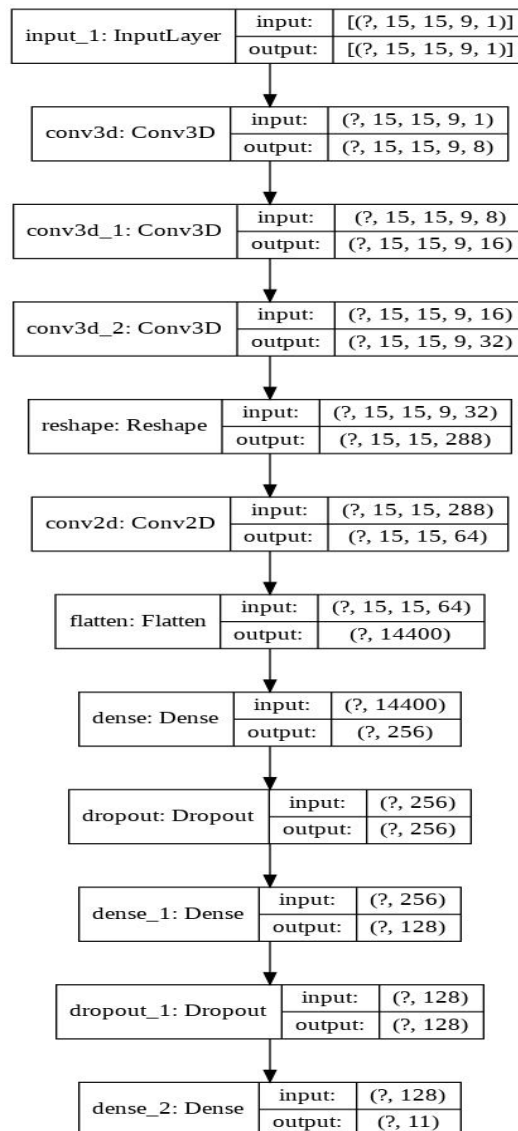


Fig.12.1.1. Crop-Net architecture

Training, validation and testing of Crop-Net is carried out using the patches of size 15×15 generated from AIRSAR L Band Flevoland POLSAR data (this image and its' GT are given in **Section 1.3.4** of **Chapter 1.**) 70% of the generated patches are used for training Crop-Net while validation of the network is carried out using 10% of the generated patches. Testing of Crop-Net is carried out using 20% of the generated patches. Patches used during testing of Crop-Net are not used during either

training phase or validation phase. Training of network is carried out using stochastic gradient descent algorithm [181] and back-propagation as a gradient computing technique. Early stopping criterion is imposed in this work to avoid over-fitting of the designed network. Here the training is stopped when no improvement in the performance of the network is achieved for ten consecutive iterations on validation set. The learning rate of the network is initially set to 1e-4 but later adjusted with iterations using Slanted Triangular Learning Rate [219] technique. PReLU [167] is used as an activation function in Crop-Net.

12.2. Feature extraction

In this work, classification is performed using features selected by the proposed Ranking Method from a set of 151 extracted features which includes decomposition methods' based features, coherency matrix, Radar Vegetation Index, etc. These features are extracted giving despeckled image files as inputs to the POLSARpro software. The detailed description of the extracted features are given in [220].

12.3. Feature selection

Following feature extraction, selection of significant features from the set of extracted features is performed using a novel Ranking Method which selects features based on their respective information content and contribution in classification estimated by Fuzzy Entropy [31] and TCS parameter respectively.

The details of TCS parameter is given in **Section 9.4.2** of **Chapter 9**.

The steps of the proposed Ranking Method are discussed as follows:

Step 1. Fuzzy entropy values of features are computed in order to estimate their respective information content.

Step 2. TCS values of features are estimated for n number of times (Here n=30).

Step 3. A rank namely, FE_{rank_j} is assigned to the j^{th} feature based on its Fuzzy entropy value. The feature having least Fuzzy entropy value have highest information content hence, that feature is assigned $FE_{rank_p} = 1$ (p represents a random feature index) while the feature having maximum Fuzzy entropy value have lowest information content hence, that feature is assigned $FE_{rank_p} = N$. (Here N = 151, as it signifies the total number of extracted features.)

Step 4. A rank namely, $TCS_{rank_{ij}}$ is assigned to the j^{th} feature based on its estimated TCS value computed in the i^{th} time. The feature having least TCS value in i^{th} time have minimum contribution in performing classification hence, that feature is assigned $TCS_{rank_{ip}} = N$ while the feature having maximum TCS value in the i^{th} time have maximum contribution in performing classification hence, that feature is assigned $TCS_{rank_{ip}} = 1$.

At the end of this step, 30 $TCS_{rank_{ip}}$ is assigned to each feature as TCS value of a feature is computed 30 times.

Step 5. In this step, a rank namely, $TCS_{avgrank_j}$ is assigned to the j^{th} feature by averaging its 30 $TCS_{rank_{ij}}$ using the following mathematical equation:

$$TCS_{avgrank_j} = \sum_{i=1}^n TCS_{rank_{ij}} \quad (12.3.1)$$

Estimated Fuzzy Entropy value computed for each feature and $TCS_{avgrank}$ assigned to them are given in Table 12.2.3.1. TCS values obtained for each feature during (n = 1) are also given in that Table as example.

Table 12.2.3.1. Estimated information content and $TCS_{avgrank}$ of each feature

Sl.No	Feature	Fuzzy Entropy value	$TCS_{avgrank}$	TCS value (n=1)
1.	An_Yang4_Dbl	0	71	8.49841106007807
2.	An_Yang4_Hlx	0	75	10.295231642201543
3.	An_Yang4_Odd	0	81	9.797471864148974
4.	An_Yang4_Vol	0	69	9.344510725233704
5.	Arii3_NNED_Dbl	0	78	9.806609922554344
6.	Arii3_NNED_Odd	0	77	8.183643751079217
7.	Arii3_NNED_Vol	0	81	9.529565813019872
8.	T12_imag	0	73	9.44232354266569
9.	T12_real	0	67	8.577941559487954
10.	T13_imag	0	69	8.629685698077083
11.	T13_real	0	62	9.614928477909416
12.	T23_imag	0	63	9.76933319773525
13.	T23_real	0	60	9.133505705744028
14.	Freeman_Dbl	0	60	8.54852494597435
15.	Freeman_Odd	0	74	8.451142421923578
16.	Freeman_Vol	0	75	9.278426544507965
17.	Krogager_Kh	0	84	8.898307613329962
18.	MCSM_Hlx	0	70	9.807809369638562
19.	MCSM_Vol	0	61	9.969269782304764
20.	MCSM_Wire	0	67	9.03043690789491
21.	feature 1	0	89	9.875435026944615
22.	VanZyl3_Dbl	0	80	8.14242148026824
23.	VanZyl3_Odd	0	91	9.12126418761909
24.	VanZyl3_Vol	0	72	9.518870156258345
25.	Yamaguchi3_Dbl	0	70	9.35662810318172
26.	Yamaguchi3_Odd	0	85	9.109856134513393
27.	Yamaguchi3_Vol	0	77	9.346989485900849
28.	Alpha_entropyNeg	0	83	8.712935447692871
29.	AlphaNeg_entropyNeg	0	71	9.021330130752176
30.	Aghababae_Orientation_max_SM1	0	77	8.885462042875588
31.	Singh4_G4U1_Hlx_dB	0	87	10.061425304040313
32.	conformity	0	64	8.883944875560701
33.	scatt_predominance	0	66	9.237723296508193
34.	Aghababae_M_SM1_dB	0.030337068	75	7.935561404097825

35.	Singh4_G4U1_Vol_dB	0.108431317	78	9.330919072031975
36.	An_Yang3_Vol_dB	0.251734103	65	9.604777349159122
37.	Aghababae M_SM2_dB	0.601285178	79	7.925651000696234
38.	Yamaguchi4_Y4O_Vol_dB	0.63635721	79	8.824525171890855
39.	Yamaguchi4_S4R_Vol_dB	0.67440892	73	9.091684993356466
40.	Bhattacharya_Frery_BF4_Vol_dB	0.682526103	76	9.616654423996806
41.	Barnes1_T33_dB	0.684288339	69	8.540034040808678
42.	Freeman2_Vol_dB	0.691314937	73	10.160788114881143
43.	Aghababae M_SM3_dB	0.692344657	65	9.40677657048218
44.	Barnes1_T22_dB	0.701110667	76	8.511308226268739
45.	Holm1_T11_dB	0.733656118	89	9.01528334338218
46.	An_Yang3_Odd_dB	0.788043826	79	8.862586219562218
47.	Aghababae_Alphap_mean	0.901139319	65	9.01270392909646
48.	Huynen_T11_dB	0.906649313	77	8.563165115192533
49.	Yamaguchi4_Y4O_Odd_dB	0.915592675	67	10.150295108556747
50.	Singh4_G4U1_Odd_dB	1.333117577	89	8.502417794428766
51.	Bhattacharya_Frery_BF4_Odd_dB	1.64256741	84	9.265424153767526
52.	Singh_i6SD_Odd_dB	1.645534405	68	8.8756586718373
53.	Yamaguchi4_S4R_Odd_dB	1.711172793	73	9.435016756877303
54.	Aghababae_Alphap_SM2	1.910797015	71	8.529411740601063
55.	Aghababae_Alphap_SM3	2.074944786	51	10.323421720415354
56.	Singh_i6SD_Dbl_dB	2.195479945	78	8.745614470914006
57.	Singh4_G4U1_Dbl_dB	2.211145516	90	9.273150190711021
58.	Bhattacharya_Frery_BF4_Dbl_dB	2.261971533	67	9.497788412496448
59.	Yamaguchi4_Y4O_Dbl_dB	2.275015988	65	8.976202882826328
60.	Yamaguchi4_S4R_Dbl_dB	2.286972957	70	9.105310541577637
61.	Aghababae_Phip_SM3	2.342814988	66	9.079544593114406
62.	Aghababae_Phip_mean	2.354850494	68	9.344676431501284
63.	Singh_i6SD_Vol_dB	2.363064416	66	9.347997817443684
64.	An_Yang3_Dbl_dB	2.366351356	88	8.407321334816515
65.	Freeman2_Ground_dB	2.366351356	67	9.050825419835746
66.	Holm1_T22_dB	2.393786483	72	9.019756309688091
67.	Cloude_T11_dB	2.405816868	77	9.494503281312063
68.	Holm2_T11_dB	2.413348762	74	9.30921037774533
69.	Barnes1_T11_dB	2.423304067	75	8.738561803940684
70.	Barnes2_T33_dB	2.439677792	86	9.435055688023567
71.	Aghababae_Phip_SM2	2.445834714	72	9.244047455955297

72.	Holm1_T33_dB	2.446343593	68	9.82300629466772
73.	Aghababaece_Orientation_max_SM3	2.448463927	77	9.68613121425733
74.	Barnes2_T22_dB	2.4563357	73	10.168867779895663
75.	Singh_i6SD_OD_dB	2.460760462	60	8.949465667828918
76.	Huynen_T33_dB	2.463384394	87	7.851741173770279
77.	Singh_i6SD_CD_dB	2.463573672	87	8.739310584962368
78.	Singh_i6SD_Hlx_dB	2.46805725	77	9.430187784135342
79.	Bhattacharya_Frery_BF4_Hlx_dB	2.49106806	63	9.763453794643283
80.	Yamaguchi4_Y4O_Hlx_dB	2.493500528	75	9.346348723396659
81.	Yamaguchi4_S4R_Hlx_dB	2.49957635	87	8.93141314946115
82.	Huynen_T22_dB	2.503669418	81	8.174996389774606
83.	Aghababaece_Tawp_SM1	2.507269525	86	8.953196186106652
84.	Aghababaece_Tawp_mean	2.515975046	78	8.793234590440989
85.	Cloude_T22_dB	2.518768459	81	9.485675308853388
86.	Cloude_T33_dB	2.531544874	79	9.27529773209244
87.	Holm2_T22_dB	2.532669443	76	8.548610750585794
88.	Holm2_T33_dB	2.536907957	94	8.56089044874534
89.	Aghababaece_Tawp_SM2	2.550429076	80	9.340628867037594
90.	Aghababaece_Orientation_max_mean	2.554387869	74	9.155076812952757
91.	Barnes2_T11_dB	2.597705595	94	8.880419051274657
92.	Aghababaece_Tawp_SM3	2.610356965	70	9.364790313877165
93.	Aghababaece_Alphap_SM1	2.628895976	76	9.512936653220095
94.	Aghababaece_Orientation_max_SM2	2.649672547	85	10.03840054711327
95.	scatt_diversity	2.66103826	67	9.338813008973375
96.	degree_purity	2.665296851	83	8.96094337105751
97.	CCNorm	2.668677462	74	8.90263532847166
98.	Aghababaece_Phip_SM1	2.67913289	78	9.567747370922007
99.	Neumann_delta_mod	2.708364985	82	9.971582145430148
100.	depolarisation_index	2.725174375	77	8.950973495841026
101.	TSVM_alpha_s	2.752332703	75	9.755773256532848
102.	alpha	2.805237019	62	8.351968116359785
103.	TSVM_alpha_s1	2.833400065	80	8.141138772945851
104.	Alpha_entropy	2.841210043	69	8.912595286499709
105.	Neumann_delta pha	2.846127199	76	8.509331441484392
106.	AlphaNeg_entropy	2.850294473	72	9.092022307449952
107.	rvi	2.861184332	70	9.422632409259677
108.	Freeman_rvi	2.945778154	70	8.554523355327547

109	entropy	2.996058268	62	8.784193216823041
110	pedestal height	3.288333544	66	8.728462477913126
111	Alpha_AnistropyNeg	3.320817856	74	8.898862646427006
112	AlphaNeg_Anisotropy	3.344450873	70	8.76123788766563
113	feature2	3.416203667	71	9.640161050483584
114	TSVM_alpha_s2	3.609491348	72	9.138045658823103
115	TSVM_psi2	3.845306333	67	9.156673038378358
116	feature5	3.989596175	76	9.871213790029287
117	Neumann_tau	4.155235669	75	9.760947803966701
118	feature6	4.155235669	85	9.06892828643322
119	TSVM_tau_m1	4.225736241	63	8.883734310977161
120	beta	4.244774673	64	9.454265059437603
121	AlphaNeg_AnisotropyNeg	4.273813488	83	9.883048639632761
122	TSVM_tau_m2	4.382727136	83	9.276523249223828
123	AnisotropyNeg_entropy	4.385116346	78	9.384660806506872
124	TSVM_tau_m	4.491737726	72	9.386454248335212
125	Anisotropy_entropy	4.511460382	74	9.094676777720451
126	TSVM_alpha_s3	4.561852268	85	8.642332427203655
127	AnisotropyNeg_entropyNeg	4.58917105	72	8.93830158887431
128	TSVM_tau_m3	4.60203522	81	10.43788734357804
129	Anisotropy_entropyNeg	4.618879973	75	9.451136153191328
130	TSVM_psi1	4.62097822	86	8.347071443451568
131	TSVM_psi3	4.623812027	76	8.82791235204786
132	feature4	4.62829571	64	9.31102753430605
133	feature3	4.634683266	79	8.606860561296344
134	delta	4.635938071	70	10.275018522050232
135	TSVM_psi	4.669276419	68	9.758043806999922
136	TSVM_phi_s2	4.694094571	72	9.544423955492675
137	TSVM_phi_sl	4.741363877	75	7.1768451649695635
138	TSVM_phi_s	4.774513533	67	10.355733229778707
139	MCSM_Dbl	4.787658833	74	8.53752325475216
140	MCSM_DblHlx	4.813413217	88	9.993599648005329
141	gamma	4.825853407	82	9.22515924833715
142	TSVM_phi_s3	4.856290229	81	10.032533168792725
143	lambda	4.872661561	70	9.069898310117424
144	T11	4.880169617	80	9.207554920576513
145	Alpha_Anistropy	4.884916685	77	9.400624967645854

146	Krogager_Kd	4.900826854	72	9.620540242409334
147	Krogager_Ks	4.91687013	68	9.50993026420474
148	MCSM_Odd	4.924100311	57	9.217991690151393
149	T22	4.943189601	81	9.323987868614495
150	T33	4.944835899	73	8.727150532184169
151	anisotropy	4.951557204	67	9.226749860448763

Features namely, feature 1, feature 2, feature 3, feature 4, feature 5 and feature 6 (given in Table 12.2.3.1.) are computed as stated below:

$$\text{feature 1} = 10 \log_{10} (\text{SPAN})$$

$$\text{feature 2} = T_{22}/\text{SPAN}$$

$$\text{feature 3} = T_{33}/\text{SPAN}$$

$$\text{feature 4} = |T_{12}|/\sqrt{T_{11} \cdot T_{22}}$$

$$\text{feature 5} = |T_{13}|/\sqrt{T_{11} \cdot T_{33}}$$

$$\text{feature 6} = |T_{23}|/\sqrt{T_{33} \cdot T_{22}}$$

Step 6: In this step, ten different sets of selected features are created as stated below:

First set comprises of features which possess $FE_{rank}=[1 \ 15]$ and $TCS_{avgrank}=[1 \ 15]$ simultaneously. Here the range is considered to be $[1 \ 15]$ to denote top 10% of features $[10\% \text{ of } N(151) \cong 15]$. The details of these features are given in Table 12.2.3.2.

Table 12.2.3.2. Features included in the first set

Sl.No	Features having $FE_{rank}=[1 \ 15]$	Features having $TCS_{avgrank}=[1 \ 15]$	List of selected features
1.	An_Yang4_Dbl	Aghababae_Alphap_SM3	T13_real
2.	An_Yang4_Hlx	MCSM_Odd	T23_imag
3.	An_Yang4_Odd	T23_real	T23_real
4.	An_Yang4_Vol	Freeman_Dbl	Freeman_Dbl
5.	Arii3_NNED_Dbl	Singh_i6SD_OD_dB	
6.	Arii3_NNED_Odd	MCSM_Vol	
7.	Arii3_NNED_Vol	T13_real	
8.	T12_imag	alpha	
9.	T12_real	entropy	
10.	T13_imag	T23_imag	
11.	T13_real	TSVM_tau_m1	
12.	T23_imag	Bhattacharya_Frery_BF4_Hlx_dB	
13.	T23_real	beta	
14.	Freeman_Dbl	feature4	
15.	Freeman_Odd	conformity	

Second set comprises of features which possess $FE_{rank}=[1\ 30]$ and $TCS_{avgrank}=[1\ 30]$ simultaneously. Here the range is considered to be [1 30] to denote top 20% of features [20%of N(151) \cong 30]. The details of these features are given in Table 12.2.3.3.

Table 12.2.3.3 Features included in the second set

Sl.No	Features having $FE_{rank}=[1\ 30]$	Features having $TCS_{avgrank}=[1\ 30]$	List of selected features
1.	An_Yang4_Dbl	Aghababae_Alphap_SM3	T13_real
2.	An_Yang4_Hlx	MCSM_Odd	T23_imag
3.	An_Yang4_Odd	T23_real	T23_real
4.	An_Yang4_Vol	Freeman_Dbl	Freeman_Dbl
5.	Arii3_NNED_Dbl	Singh_i6SD_OD_dB	T12_real
6.	Arii3_NNED_Odd	MCSM_Vol	MCSM_Vol
7.	Arii3_NNED_Vol	T13_real	MCSM_Wire
8.	T12_imag	alpha	
9.	T12_real	entropy	
10.	T13_imag	T23_imag	
11.	T13_real	TSVM_tau_m1	
12.	T23_imag	Bhattacharya_Frery_BF4_Hlx_dB	
13.	T23_real	beta	
14.	Freeman_Dbl	feature4	
15.	Freeman_Odd	conformity	
16.	Freeman_Vol	Aghababae_Alphap_mean	
17.	Krogager_Kh	Aghababae_M_SM3_dB	
18.	MCSM_Hlx	An_Yang3_Vol_dB	
19.	MCSM_Vol	Yamaguchi4_Y4O_Dbl_dB	
20.	MCSM_Wire	pedestal height	
21.	feature 1	Aghababae_Phip_SM3	
22.	VanZyl3_Dbl	Singh_i6SD_Vol_dB	
23.	VanZyl3_Odd	scatt_predominance	
24.	VanZyl3_Vol	T12_real	
25.	Yamaguchi3_Dbl	anisotropy	
26.	Yamaguchi3_Odd	MCSM_Wire	
27.	Yamaguchi3_Vol	TSVM_phi_s	
28.	Alpha_entropyNeg	TSVM_psi2	
29.	AlphaNeg_entropyNeg	Bhattacharya_Frery_BF4_Dbl_dB	
30.	Aghababae_Orientation_max_SM1	Freeman2_Ground_dB	

Third set comprises of features which possess $FE_{rank}=[1\ 45]$ and $TCS_{avgrank}=[1\ 45]$ simultaneously. Here the range is considered to be [1 45] to denote top 30% of features [30%of N(151) \cong 45]. The details of these features are given in Table 12.2.3.4.

Table 12.2.3.4. Features included in the third set

Sl.No	Features having $FE_{rank}=[1\ 45]$	Features having $TCS_{avrank}=[1\ 45]$	List of selected features
1.	An_Yang4_Dbl	Aghababaece_Alphap_SM3	
2.	An_Yang4_Hlx	MCSM_Odd	T13_real
3.	An_Yang4_Odd	T23_real	T23_imag
4.	An_Yang4_Vol	Freeman_Dbl	T23_real
5.	Arii3_NNED_Dbl	Singh_i6SD_OD_dB	Freeman_Dbl
6.	Arii3_NNED_Odd	MCSM_Vol	T12_real
7.	Arii3_NNED_Vol	T13_real	MCSM_Vol
8.	T12_imag	alpha	MCSM_Wire
9.	T12_real	entropy	An_Yang4_Vol
10.	T13_imag	T23_imag	Conformity
11.	T13_real	TSVM_tau_m1	Scatt_predominance
12.	T23_imag	Bhattacharya_Frery_BF4_Hlx_dB	pedstal height
13.	T23_real	beta	
14.	Freeman_Dbl	feature4	
15.	Freeman_Odd	conformity	
16.	Freeman_Vol	Aghababaece_Alphap_mean	
17.	Krogager_Kh	Aghababaece_M_SM3_dB	
18.	MCSM_Hlx	An_Yang3_Vol_dB	
19.	MCSM_Vol	Yamaguchi4_Y4O_Dbl_dB	
20.	MCSM_Wire	pedestal height	
21.	feature 1	Aghababaece_Phip_SM3	
22.	VanZyl3_Dbl	Singh_i6SD_Vol_dB	
23.	VanZyl3_Odd	scatt_predominance	
24.	VanZyl3_Vol	T12_real	
25.	Yamaguchi3_Dbl	anisotropy	
26.	Yamaguchi3_Odd	MCSM_Wire	
27.	Yamaguchi3_Vol	TSVM_phi_s	
28.	Alpha_entropyNeg	TSVM_psi2	
29.	AlphaNeg_entropyNeg	Bhattacharya_Frery_BF4_Dbl_dB	
30.	Aghababaece_Orientation_max_SM1	Freeman2_Ground_dB	
31.	Singh4_G4U1_Hlx_dB	Yamaguchi4_Y4O_Odd_dB	
32.	conformity	scatt_diversity	
33.	scatt_predominance	Krogager_Ks	
34.	pedestal height	TSVM_psi	
35.	Alpha_Anistropy	Aghababaece_Phip_mean	
36.	degree_purity	Holm1_T33_dB	

37.	feature3	Singh_i6SD_Odd_dB
38.	depolarisation_index	An_Yang4_Vol
39.	Holm2_T11_dB	T13_imag
40.	Barnes2_T22_dB	Alpha_entropy
41.	Barnes2_T33_dB	Barnes1_T33_dB
42.	Holm2_T33_dB	Freeman_rvi (unfiltered)
43.	Cloude_T33_dB	delta
44.	Cloude_T11_dB	lambda
45.	CCCnorm	MCSM_Hlx

Fourth set comprises of features which possess $FE_{rank}=[1\ 60]$ and $TCS_{avgrank}=[1\ 60]$ simultaneously. Here the range is considered to be [1 60] to denote top 40% of features [40% of $N(151) \cong 60$]. The details of these features are given in Table 12.2.3.5.

Table 12.2.3.5. Features included in the fourth set

Sl.No	Features having $FE_{rank}=[1\ 60]$	Features having $TCS_{avgrank}=[1\ 60]$	List of selected features
1.	An_Yang4_Dbl	Aghababaece_Alphap_SM3	An_Yang4_Dbl
2.	An_Yang4_Hlx	MCSM_Odd	An_Yang4_Vol
3.	An_Yang4_Odd	T23_real	T12_real
4.	An_Yang4_Vol	Freeman_Dbl	T13_imag
5.	Arii3_NNED_Dbl	Singh_i6SD_OD_dB	T13_real
6.	Arii3_NNED_Odd	MCSM_Vol	T23_imag
7.	Arii3_NNED_Vol	T13_real	T23_real
8.	T12_imag	alpha	Freeman_Dbl
9.	T12_real	entropy	MCSM_Hlx
10.	T13_imag	T23_imag	MCSM_Vol
11.	T13_real	TSVM_tau_m1	MCSM_Wire
12.	T23_imag	Bhattacharya_Frery_BF4_Hlx_dB	VanZyl3_Vol
13.	T23_real	beta	Yamaguchi3_Dbl
14.	Freeman_Dbl	feature4	AlphaNeg_entropyNeg
15.	Freeman_Odd	conformity	conformity
16.	Freeman_Vol	Aghababaece_Alphap_mean	scatt_predominance
17.	Krogager_Kh	Aghababaece_M_SM3_dB	pedestal height
18.	MCSM_Hlx	An_Yang3_Vol_dB	scatt_diversity
19.	MCSM_Vol	Yamaguchi4_Y4O_Dbl_dB	AlphaNeg_Anisotropy
20.	MCSM_Wire	pedestal height	Singh_i6SD_OD_dB
21.	feature 1	Aghababaece_Phip_SM3	
22.	VanZyl3_Dbl	Singh_i6SD_Vol_dB	
23.	VanZyl3_Odd	scatt_predominance	

24.	VanZyl3_Vol	T12_real
25.	Yamaguchi3_Dbl	anisotropy
26.	Yamaguchi3_Odd	MCSM_Wire
27.	Yamaguchi3_Vol	TSVM_phi_s
28.	Alpha_entropyNeg	TSVM_psi2
29.	AlphaNeg_entropyNeg	Bhattacharya_Frery_BF4_Dbl_dB
30.	Aghababae_Orientation_max_SM1	Freeman2_Ground_dB
31.	Singh4_G4U1_Hlx_dB	Yamaguchi4_Y4O_Odd_dB
32.	conformity	scatt_diversity
33.	scatt_predominance	Krogager_Ks
34.	pedestal height	TSVM_psi
35.	Alpha_Anistropy	Aghababae_Phip_mean
36.	degree_purity	Holm1_T33_dB
37.	feature3	Singh_i6SD_Odd_dB
38.	depolarisation_index	An_Yang4_Vol
39.	Holm2_T11_dB	T13_imag
40.	Barnes2_T22_dB	Alpha_entropy
41.	Barnes2_T33_dB	Barnes1_T33_dB
42.	Holm2_T33_dB	Freeman_rvi (unfiltered)
43.	Cloude_T33_dB	delta
44.	Cloude_T11_dB	lambda
45.	CCCnorm	MCSM_Hlx
46.	scatt_diversity	Yamaguchi3_Dbl
47.	Barnes2_T11_dB	AlphaNeg_Anistropy
48.	feature6	rvi
49.	AlphaNeg_Anistropy	Aghababae_Tawp_SM3
50.	feature5	Yamaguchi4_S4R_Dbl_dB
51.	Singh_i6SD_OD_dB	An_Yang4_Dbl
52.	Singh_i6SD_CD_dB	feature2
53.	Singh_i6SD_Hlx_dB	AlphaNeg_entropyNeg
54.	AnisotropyNeg_entropyNeg	Aghababae_Alphap_SM2
55.	Neumann_delta_mod	Krogager_Kd
56.	Aghababae_Tawp_SM2	TSVM_alpha_s2
57.	Cloude_T22_dB	TSVM_phi_s2
58.	Aghababae_Alphap_SM1	TSVM_tau_m
59.	Huynen_T33_dB	VanZyl3_Vol
60.	Aghababae_Orientation_max_mean	AlphaNeg_entropy

Fifth set comprises of features which possess $FE_{rank}=[1\ 75]$ and $TCS_{avgrank}=[1\ 75]$ simultaneously. Here the range is considered to be $[1\ 75]$ to denote top 50% of features $[50\% \text{ of } N(151) \cong 75]$. The details of these features are given in Table 12.2.3.6.

Table 12.2.3.6. Features included in the fifth set

Sl.No	Features having $FE_{rank}=[1\ 75]$	Features having $TCS_{avgrank}=[1\ 75]$	List of selected features
1.	An_Yang4_Dbl	Aghababaece_Alphap_SM3	An_Yang4_Dbl
2.	An_Yang4_Hlx	MCSM_Odd	An_Yang4_Vol
3.	An_Yang4_Odd	T23_real	T12_imag
4.	An_Yang4_Vol	Freeman_Dbl	T12_real
5.	Arii3_NNED_Dbl	Singh_i6SD_OD_dB	T13_imag
6.	Arii3_NNED_Odd	MCSM_Vol	T13_real
7.	Arii3_NNED_Vol	T13_real	T23_imag
8.	T12_imag	alpha	T23_real
9.	T12_real	entropy	Freeman_Dbl
10.	T13_imag	T23_imag	Freeman_Odd
11.	T13_real	TSVM_tau_m1	MCSM_Hlx
12.	T23_imag	Bhattacharya_Frery_BF4_Hlx_dB	MCSM_Vol
13.	T23_real	beta	MCSM_Wire
14.	Freeman_Dbl	feature4	VanZyl3_Vol
15.	Freeman_Odd	conformity	Yamaguchi3_Dbl
16.	Freeman_Vol	Aghababaece_Alphap_mean	AlphaNeg_entropyNeg
17.	Krogager_Kh	Aghababaece_M_SM3_dB	conformity
18.	MCSM_Hlx	An_Yang3_Vol_dB	scatt_predominance
19.	MCSM_Vol	Yamaguchi4_Y4O_Dbl_dB	pedestal height
20.	MCSM_Wire	pedestal height	Holm2_T11_dB
21.	feature 1	Aghababaece_Phip_SM3	Barnes2_T22_dB
22.	VanZyl3_Dbl	Singh_i6SD_Vol_dB	scatt_diversity
23.	VanZyl3_Odd	scatt_predominance	AlphaNeg_Anisotropy
24.	VanZyl3_Vol	T12_real	Singh_i6SD_OD_dB
25.	Yamaguchi3_Dbl	anisotropy	AnisotropyNeg_entropyNeg
26.	Yamaguchi3_Odd	MCSM_Wire	Aghababaece_Orientation_max_mean
27.	Yamaguchi3_Vol	TSVM_phi_s	Aghababaece_Tawp_SM3
28.	Alpha_entropyNeg	TSVM_psi2	Bhattacharya_Frery_BF4_Hlx_dB

29	AlphaNeg_entropyNeg	Bhattacharya_Frery_BF4_Dbl_dB	Aghababae_Phip_SM3
30	Aghababae_Orientation_max_SM1	Freeman2_Ground_dB	Freeman2_Vol_dB
31.	Singh4_G4U1_Hlx_dB	Yamaguchi4_Y4O_Odd_dB	
32.	conformity	scatt_diversity	
33.	scatt_predominance	Krogager_Ks	
34.	pedestal height	TSVM_psi	
35.	Alpha_Anistropy	Aghababae_Phip_mean	
36.	degree_purity	Holm1_T33_dB	
37.	feature3	Singh_i6SD_Odd_dB	
38.	depolarisation_index	An_Yang4_Vol	
39.	Holm2_T11_dB	T13_imag	
40.	Barnes2_T22_dB	Alpha_entropy	
41.	Barnes2_T33_dB	Barnes1_T33_dB	
42.	Holm2_T33_dB	Freeman_rvi (unfiltered)	
43.	Cloude_T33_dB	delta	
44.	Cloude_T11_dB	lambda	
45.	CCNorm	MCSM_Hlx	
46.	scatt_diversity	Yamaguchi3_Dbl	
47.	Barnes2_T11_dB	AlphaNeg_Anisotropy	
48.	feature6	rvi	
49.	AlphaNeg_Anisotropy	Aghababae_Tawp_SM3	
50.	feature5	Yamaguchi4_S4R_Dbl_dB	
51.	Singh_i6SD_OD_dB	An_Yang4_Dbl	
52.	Singh_i6SD_CD_dB	feature2	
53.	Singh_i6SD_Hlx_dB	AlphaNeg_entropyNeg	
54.	AnisotropyNeg_entropyNeg	Aghababae_Alphap_SM2	
55.	Neumann_delta_mod	Krogager_Kd	
56.	Aghababae_Tawp_SM2	TSVM_alpha_s2	
57.	Cloude_T22_dB	TSVM_phi_s2	
58.	Aghababae_Alphap_SM1	TSVM_tau_m	
59.	Huynen_T33_dB	VanZyl3_Vol	
60.	Aghababae_Orientation_max_mean	AlphaNeg_entropy	
61.	Aghababae_Tawp_SM3	AnisotropyNeg_entropyNeg	
62.	Holm2_T22_dB	Aghababae_Phip_SM2	
63.	Yamaguchi4_S4R_Hlx_dB	Holm1_T22_dB	
64.	Bhattacharya_Frery_BF4_Hlx_dB	Barnes2_T22_dB	
65.	Yamaguchi4_Y4O_Hlx_dB	T12_imag	

66.	Holm1_T33_dB	T33
67.	Aghababae_Tawp_SM1	Freeman2_Vol_dB
68.	Aghababae_Tawp_mean	Yamaguchi4_S4R_Odd_dB
69.	Yamaguchi4_Y4O_Dbl_dB	Yamaguchi4_S4R_Vol_dB
70.	Barnes1_T11_dB	Freeman_Odd
71.	Singh_i6SD_Vol_dB	Holm2_T11_dB
72.	feature2	MCSM_Dbl
73.	Yamaguchi4_S4R_Dbl_dB	Alpha_AnistropyNeg
74.	Aghababae_Phip_SM3	Anisotropy_entropy
75.	Freeman2_Vol_dB	Aghababae_Orientation_max_mean

Sixth set comprises of features which possess $FE_{rank}=[1\ 90]$ and $TCS_{avgrank}=[1\ 90]$ simultaneously. Here the range is considered to be $[1\ 90]$ to denote top 60% of features $[60\% \text{ of } N(151) \cong 90]$. The details of these features are given in Table 12.2.3.7.

Table 12.2.3.7. Features included in the sixth set

Sl.No	Features having $FE_{rank}=[1\ 90]$	Features having $TCS_{avgrank}=[1\ 90]$	List of selected features
1.	An_Yang4_Dbl	Aghababae_Alphap_SM3	An_Yang4_Dbl
2.	An_Yang4_Hlx	MCSM_Odd	An_Yang4_Hlx
3.	An_Yang4_Odd	T23_real	An_Yang4_Vol
4.	An_Yang4_Vol	Freeman_Dbl	T12_imag
5.	Arii3_NNED_Dbl	Singh_i6SD_OD_dB	T12_real
6.	Arii3_NNED_Odd	MCSM_Vol	T13_imag
7.	Arii3_NNED_Vol	T13_real	T13_real
8.	T12_imag	alpha	T23_imag
9.	T12_real	entropy	T23_real
10.	T13_imag	T23_imag	Freeman_Dbl
11.	T13_real	TSVM_tau_m1	Freeman_Odd
12.	T23_imag	Bhattacharya_Frery_BF4_Hlx_dB	Freeman_Vol
13.	T23_real	beta	MCSM_Hlx
14.	Freeman_Dbl	feature4	MCSM_Vol
15.	Freeman_Odd	conformity	MCSM_Wire
16.	Freeman_Vol	Aghababae_Alphap_mean	VanZyl3_Vol
17.	Krogager_Kh	Aghababae_M_SM3_dB	Yamaguchi3_Dbl
18.	MCSM_Hlx	An_Yang3_Vol_dB	AlphaNeg_entropyNeg
19.	MCSM_Vol	Yamaguchi4_Y4O_Dbl_dB	conformity
20.	MCSM_Wire	pedestal height	scatt_predominance
21.	feature 1	Aghababae_Phip_SM3	pedestal height
22.	VanZyl3_Dbl	Singh_i6SD_Vol_dB	Holm2_T11_dB

23.	VanZyl3_Odd	scatt_predominance	Barnes2_T22_dB
24.	VanZyl3_Vol	T12_real	CCCnorm
25.	Yamaguchi3_Dbl	anisotropy	scatt_diversity
26.	Yamaguchi3_Odd	MCSM_Wire	AlphaNeg_Anisotropy
27.	Yamaguchi3_Vol	TSVM_phi_s	feature5
28.	Alpha_entropyNeg	TSVM_psi2	Singh_i6SD_OD_dB
29.	AlphaNeg_entropyNeg	Bhattacharya_Frery_BF4_Dbl_dB	AnisotropyNeg_entropyNeg
30.	Aghababae_Orientation_max_SM1	Freeman2_Ground_dB	Aghababae_Alphap_SM1
31.	Singh4_G4U1_Hlx_dB	Yamaguchi4_Y4O_Odd_dB	Aghababae_Orientation_max_mean
32.	conformity	scatt_diversity	Aghababae_Tawp_SM3
33.	scatt_predominance	Krogager_Ks	Holm2_T22_dB
34.	pedestal height	TSVM_psi	Bhattacharya_Frery_BF4_Hlx_dB
35.	Alpha_Anistropy	Aghababae_Phip_mean	Yamaguchi4_Y4O_Hlx_dB
36.	degree_purity	Holm1_T33_dB	Holm1_T33_dB
37.	feature3	Singh_i6SD_Odd_dB	Yamaguchi4_Y4O_Dbl_dB
38.	depolarisation_index	An_Yang4_Vol	Barnes1_T11_dB
39.	Holm2_T11_dB	T13_imag	Singh_i6SD_Vol_dB
40.	Barnes2_T22_dB	Alpha_entropy	feature2
41.	Barnes2_T33_dB	Barnes1_T33_dB	Aghababae_Phip_SM3
42.	Holm2_T33_dB	Freeman_rvi (unfiltered)	Freeman2_Vol_dB
43.	Cloude_T33_dB	delta	Aghababae_Alphap_SM2
44.	Cloude_T11_dB	lambda	Bhattacharya_Frery_BF4_Dbl_dB
45.	CCCnorm	MCSM_Hlx	Aghababae_Phip_mean
46.	scatt_diversity	Yamaguchi3_Dbl	Yamaguchi4_S4R_Odd_dB
47.	Barnes2_T11_dB	AlphaNeg_Anisotropy	Yamaguchi4_Y4O_Odd_dB
48.	feature6	rvi	Alpha_AnistropyNeg
49.	AlphaNeg_Anisotropy	Aghababae_Tawp_SM3	Holm1_T22_dB
50.	feature5	Yamaguchi4_S4R_Dbl_dB	Barnes1_T33_dB
51.	Singh_i6SD_OD_dB	An_Yang4_Dbl	Singh_i6SD_Odd_dB
52.	Singh_i6SD_CD_dB	feature2	
53.	Singh_i6SD_Hlx_dB	AlphaNeg_entropyNeg	
54.	AnisotropyNeg_entropyNeg	Aghababae_Alphap_SM2	
55.	Neumann_delta_mod	Krogager_Kd	
56.	Aghababae_Tawp_SM2	TSVM_alpha_s2	
57.	Cloude_T22_dB	TSVM_phi_s2	
58.	Aghababae_Alphap_SM1	TSVM_tau_m	
59.	Huynen_T33_dB	VanZyl3_Vol	
60.	Aghababae_Orientation_max_mean	AlphaNeg_entropy	

61.	Aghababae_Tawp_SM3	AnisotropyNeg_entropyNeg
62.	Holm2_T22_dB	Aghababae_Phip_SM2
63.	Yamaguchi4_S4R_Hlx_dB	Holm1_T22_dB
64.	Bhattacharya_Frery_BF4_Hlx_dB	Barnes2_T22_dB
65.	Yamaguchi4_Y4O_Hlx_dB	T12_imag
66.	Holm1_T33_dB	T33
67.	Aghababae_Tawp_SM1	Freeman2_Vol_dB
68.	Aghababae_Tawp_mean	Yamaguchi4_S4R_Odd_dB
69.	Yamaguchi4_Y4O_Dbl_dB	Yamaguchi4_S4R_Vol_dB
70.	Barnes1_T11_dB	Freeman_Odd
71.	Singh_i6SD_Vol_dB	Holm2_T11_dB
72.	feature2	MCSM_Dbl
73.	Yamaguchi4_S4R_Dbl_dB	Alpha_AnistropyNeg
74.	Aghababae_Phip_SM3	Anisotropy_entropy
75.	Freeman2_Vol_dB	Aghababae_Orientation_max_mean
76.	Aghababae_M_SM2_dB	CCCNorm
77.	Aghababae_Alphap_SM2	An_Yang4_Hlx
78.	Bhattacharya_Frery_BF4_Dbl_dB	Freeman_Vol
79.	Aghababae_Phip_mean	Neumann_tau
80.	Yamaguchi4_S4R_Odd_dB	TSVM_alpha_s
81.	Singh_i6SD_Dbl_dB	TSVM_phi_sl
82.	Yamaguchi4_Y4O_Odd_dB	Anisotropy_entropyNeg
83.	An_Yang3_Dbl_dB	Aghababae_M_SM1_dB
84.	Singh4_G4U1_Dbl_dB	Barnes1_T11_dB
85.	An_Yang3_Odd_dB	Yamaguchi4_Y4O_Hlx_dB
86.	Alpha_AnistropyNeg	Holm2_T22_dB
87.	Singh4_G4U1_Odd_dB	Neumann_delta_pha
88.	Holm1_T22_dB	feature5
89.	Barnes1_T33_dB	TSVM_psi3
90.	Singh_i6SD_Odd_dB	Aghababae_Alphap_SM1

Seventh set comprises of features which possess $FE_{rank}=[1\ 105]$ and $TCS_{avgrank}=[1\ 105]$ simultaneously. Here the range is considered to be $[1\ 105]$ to denote top 70% of features $[70\% \text{ of } N(151) \cong 105]$. The details of these features are given in Table 12.2.3.8.

Table 12.2.3.8. Features included in the seventh set

Sl.No	Features having $FE_{rank}=[1\ 105]$	Features having $TCS_{avgrank}=[1\ 105]$	List of selected features
1.	An_Yang4_Dbl	Aghababae_Alphap_SM3	An_Yang4_Dbl
2.	An_Yang4_Hlx	MCSM_Odd	An_Yang4_Hlx

3.	An_Yang4_Odd	T23_real	An_Yang4_Vol
4.	An_Yang4_Vol	Freeman_Dbl	Arii3_NNED_Dbl
5.	Arii3_NNED_Dbl	Singh_i6SD_OD_dB	Arii3_NNED_Odd
6.	Arii3_NNED_Odd	MCSM_Vol	T12_imag
7.	Arii3_NNED_Vol	T13_real	T12_real
8.	T12_imag	alpha	T13_imag
9.	T12_real	entropy	T13_real
10.	T13_imag	T23_imag	T23_imag
11.	T13_real	TSVM_tau_m1	T23_real
12.	T23_imag	Bhattacharya_Frery_BF4_Hlx_dB	Freeman_Dbl
13.	T23_real	beta	Freeman_Odd
14.	Freeman_Dbl	feature4	Freeman_Vol
15.	Freeman_Odd	conformity	MCSM_Hlx
16.	Freeman_Vol	Aghababae_Alphap_mean	MCSM_Vol
17.	Krogager_Kh	Aghababae_M_SM3_dB	MCSM_Wire
18.	MCSM_Hlx	An_Yang3_Vol_dB	VanZyl3_Vol
19.	MCSM_Vol	Yamaguchi4_Y4O_Dbl_dB	Yamaguchi3_Dbl
20.	MCSM_Wire	pedestal height	Yamaguchi3_Vol
21.	feature 1	Aghababae_Phip_SM3	AlphaNeg_entropyNeg
22.	VanZyl3_Dbl	Singh_i6SD_Vol_dB	Aghababae_Orientation_max_S M1
23.	VanZyl3_Odd	scatt_predominance	conformity
24.	VanZyl3_Vol	T12_real	scatt_predominance
25.	Yamaguchi3_Dbl	anisotropy	pedestal height
26.	Yamaguchi3_Odd	MCSM_Wire	Alpha_Anistropy
27.	Yamaguchi3_Vol	TSVM_phi_s	depolarisation_index
28.	Alpha_entropyNeg	TSVM_psi2	Holm2_T11_dB
29.	AlphaNeg_entropyNeg	Bhattacharya_Frery_BF4_Dbl_dB	Barnes2_T22_dB
30.	Aghababae_Orientation_max_SM1	Freeman2_Ground_dB	Cloude_T11_dB
31.	Singh4_G4U1_Hlx_dB	Yamaguchi4_Y4O_Odd_dB	CCCnorm
32.	conformity	scatt_diversity	scatt_diversity
33.	scatt_predominance	Krogager_Ks	AlphaNeg_Anistropy
34.	pedestal height	TSVM_psi	feature5
35.	Alpha_Anistropy	Aghababae_Phip_mean	Singh_i6SD_OD_dB
36.	degree_purity	Holm1_T33_dB	Singh_i6SD_Hlx_dB
37.	feature3	Singh_i6SD_Odd_dB	AnisotropyNeg_entropyNeg
38.	depolarisation_index	An_Yang4_Vol	Aghababae_Alphap_SM1
39.	Holm2_T11_dB	T13_imag	Aghababae_Orientation_max_ mean
40.	Barnes2_T22_dB	Alpha_entropy	

41.	Barnes2_T33_dB	Barnes1_T33_dB	Aghababae_Tawp_SM3
42.	Holm2_T33_dB	Freeman_rvi	Holm2_T22_dB
43.	Cloude_T33_dB	delta	Bhattacharya_Frery_BF4_Hlx_dB
44.	Cloude_T11_dB	lambda	Yamaguchi4_Y4O_Hlx_dB
45.	CCCnorm	MCSM_Hlx	Holm1_T33_dB
46.	scatt_diversity	Yamaguchi3_Dbl	Aghababae_Tawp_mean
47.	Barnes2_T11_dB	AlphaNeg_Anisotropy	Yamaguchi4_Y4O_Dbl_dB
48.	feature6	rvi	Barnes1_T11_dB
49.	AlphaNeg_Anisotropy	Aghababae_Tawp_SM3	Singh_i6SD_Vol_dB
50.	feature5	Yamaguchi4_S4R_Dbl_dB	feature2
51.	Singh_i6SD_OD_dB	An_Yang4_Dbl	Aghababae_Phip_SM3
52.	Singh_i6SD_CD_dB	feature2	Freeman2_Vol_dB
53.	Singh_i6SD_Hlx_dB	AlphaNeg_entropyNeg	Aghababae_Alphap_SM2
54.	AnisotropyNeg_entropyNeg	Aghababae_Alphap_SM2	Bhattacharya_Frery_BF4_Dbl_dB
55.	Neumann_delta_mod	Krogager_Kd	Aghababae_Phip_mean
56.	Aghababae_Tawp_SM2	TSVM_alpha_s2	Yamaguchi4_S4R_Odd_dB
57.	Cloude_T22_dB	TSVM_phi_s2	Yamaguchi4_Y4O_Odd_dB
58.	Aghababae_Alphap_SM1	TSVM_tau_m	Alpha_AnistropyNeg
59.	Huynen_T33_dB	VanZyl3_Vol	Holm1_T22_dB
60.	Aghababae_Orientation_max_mean	AlphaNeg_entropy	Barnes1_T33_dB
61.	Aghababae_Tawp_SM3	AnisotropyNeg_entropyNeg	Singh_i6SD_Odd_dB
62.	Holm2_T22_dB	Aghababae_Phip_SM2	Aghababae_Alphap_SM3
63.	Yamaguchi4_S4R_Hlx_dB	Holm1_T22_dB	Freeman2_Ground_dB
64.	Bhattacharya_Frery_BF4_Hlx_dB	Barnes2_T22_dB	Barnes1_T22_dB
65.	Yamaguchi4_Y4O_Hlx_dB	T12_imag	Bhattacharya_Frery_BF4_Vol_dB
66.	Holm1_T33_dB	T33	Yamaguchi4_S4R_Vol_dB
67.	Aghababae_Tawp_SM1	Freeman2_Vol_dB	Aghababae_Alphap_mean
68.	Aghababae_Tawp_mean	Yamaguchi4_S4R_Odd_dB	Aghababae_M_SM3_dB
69.	Yamaguchi4_Y4O_Dbl_dB	Yamaguchi4_S4R_Vol_dB	Aghababae_Phip_SM2
70.	Barnes1_T11_dB	Freeman_Odd	An_Yang3_Vol_dB
71.	Singh_i6SD_Vol_dB	Holm2_T11_dB	Aghababae_M_SM1_dB
72.	feature2	MCSM_Dbl	
73.	Yamaguchi4_S4R_Dbl_dB	Alpha_AnistropyNeg	
74.	Aghababae_Phip_SM3	Anisotropy_entropy	
75.	Freeman2_Vol_dB	Aghababae_Orientation_max_mean	
76.	Aghababae_M_SM2_dB	CCCnorm	
77.	Aghababae_Alphap_SM2	An_Yang4_Hlx	
78.	Bhattacharya_Frery_BF4_Dbl_dB	Freeman_Vol	

79.	Aghababae_Phip_mean	Neumann_tau
80.	Yamaguchi4_S4R_Odd_dB	TSVM_alpha_s
81.	Singh_i6SD_Dbl_dB	TSVM_phi_sl
82.	Yamaguchi4_Y4O_Odd_dB	Anisotropy_entropyNeg
83.	An_Yang3_Dbl_dB	Aghababae_M_SM1_dB
84.	Singh4_G4U1_Dbl_dB	Barnes1_T11_dB
85.	An_Yang3_Odd_dB	Yamaguchi4_Y4O_Hlx_dB
86.	Alpha_AnistropyNeg	Holm2_T22_dB
87.	Singh4_G4U1_Odd_dB	Neumann_delta_pha
88.	Holm1_T22_dB	feature5
89.	Barnes1_T33_dB	TSVM_psi3
90.	Singh_i6SD_Odd_dB	Aghababae_Alphap_SM1
91.	Bhattacharya_Frery_BF4_Odd_dB	Barnes1_T22_dB
92.	Aghababae_Alphap_SM3	Bhattacharya_Frery_BF4_Vol_dB
93.	Freeman2_Ground_dB	Arii3_NNED_Odd
94.	Barnes1_T22_dB	Cloude_T11_dB
95.	Yamaguchi4_Y4O_Vol_dB	Huynen_T11_dB
96.	Holm1_T11_dB	Yamaguchi3_Vol
97.	Aghababae_Orientation_max_SM2	Alpha_Anistropy
98.	Bhattacharya_Frery_BF4_Vol_dB	Aghababae_Orientation_max_SM1
99.	Singh4_G4U1_Vol_dB	Aghababae_Orientation_max_SM3
100.	Yamaguchi4_S4R_Vol_dB	Singh_i6SD_Hlx_dB
101.	Aghababae_Alphap_mean	Singh_i6SD_Hlx_dB
102.	Aghababae_M_SM3_dB	depolarisation_index
103.	Aghababae_Phip_SM2	Arii3_NNED_Dbl
104.	An_Yang3_Vol_dB	AnisotropyNeg_entropy
105.	Aghababae_M_SM1_dB	Aghababae_Tawp_mean

Eighth set comprises of features which possess $FE_{rank}=[1\ 120]$ and $TCS_{avgrank}=[1\ 120]$ simultaneously. Here the range is considered to be $[1\ 120]$ to denote top 80% of features $[80\% \text{ of } N(151) \cong 120]$. The details of these features are given in Table 12.2.3.9.

Table 12.2.3.9. Features included in the eighth set

Sl.No	Features having $FE_{rank}=[1\ 120]$	Features having $TCS_{avgrank}=[1\ 120]$	List of selected features
1.	An_Yang4_Dbl	Aghababae_Alphap_SM3	An_Yang4_Dbl
2.	An_Yang4_Hlx	MCSM_Odd	An_Yang4_Hlx
3.	An_Yang4_Odd	T23_real	An_Yang4_Odd
4.	An_Yang4_Vol	Freeman_Dbl	An_Yang4_Vol
5.	Arii3_NNED_Dbl	Singh_i6SD_OD_dB	Arii3_NNED_Dbl

6.	Arii3_NNED_Odd	MCSM_Vol	Arii3_NNED_Odd
7.	Arii3_NNED_Vol	T13_real	Arii3_NNED_Vol
8.	T12_imag	alpha	T12_imag
9.	T12_real	entropy	T12_real
10.	T13_imag	T23_imag	T13_imag
11.	T13_real	TSVM_tau_m1	T13_real
12.	T23_imag	Bhattacharya_Frery_BF4_Hlx_dB	T23_imag
13.	T23_real	beta	T23_real
14.	Freeman_Dbl	feature4	Freeman_Dbl
15.	Freeman_Odd	conformity	Freeman_Odd
16.	Freeman_Vol	Aghababae_Alphap_mean	Freeman_Vol
17.	Krogager_Kh	Aghababae_M_SM3_dB	MCSM_Hlx
18.	MCSM_Hlx	An_Yang3_Vol_dB	MCSM_Vol
19.	MCSM_Vol	Yamaguchi4_Y4O_Dbl_dB	MCSM_Wire
20.	MCSM_Wire	pedestal height	VanZyl3_Dbl
21.	feature 1	Aghababae_Phip_SM3	VanZyl3_Vol
22.	VanZyl3_Dbl	Singh_i6SD_Vol_dB	Yamaguchi3_Dbl
23.	VanZyl3_Odd	scatt_predominance	Yamaguchi3_Vol
24.	VanZyl3_Vol	T12_real	AlphaNeg_entropyNeg
25.	Yamaguchi3_Dbl	anisotropy	Aghababae_Orientation_max_SM1
26.	Yamaguchi3_Odd	MCSM_Wire	conformity
27.	Yamaguchi3_Vol	TSVM_phi_s	scatt_predominance
28.	Alpha_entropyNeg	TSVM_psi2	pedestal height
29.	AlphaNeg_entropyNeg	Bhattacharya_Frery_BF4_Dbl_dB	Alpha_Anistropy
30.	Aghababae_Orientation_max_SM1	Freeman2_Ground_dB	feature3
31.	Singh4_G4U1_Hlx_dB	Yamaguchi4_Y4O_Odd_dB	depolarisation_index
32.	conformity	scatt_diversity	Holm2_T11_dB
33.	scatt_predominance	Krogager_Ks	Barnes2_T22_dB
34.	pedestal height	TSVM_psi	Cloude_T11_dB
35.	Alpha_Anistropy	Aghababae_Phip_mean	CCCnorm
36.	degree_purity	Holm1_T33_dB	scatt_diversity
37.	feature3	Singh_i6SD_Odd_dB	AlphaNeg_Anisotropy
38.	depolarisation_index	An_Yang4_Vol	feature5
39.	Holm2_T11_dB	T13_imag	Singh_i6SD_OD_dB
40.	Barnes2_T22_dB	Alpha_entropy	Singh_i6SD_Hlx_dB
41.	Barnes2_T33_dB	Barnes1_T33_dB	AnisotropyNeg_entropyNeg
42.	Holm2_T33_dB	Freeman_rvi (unfiltered)	Aghababae_Tawp_SM2
43.	Cloude_T33_dB	delta	Cloude_T22_dB
			Aghababae_Alphap_SM1

44.	Cloude_T11_dB	lambda	Aghababaece_Orientation_max_mean
45.	CCCnorm	MCSM_Hlx	Aghababaece_Tawp_SM3
46.	scatt_diversity	Yamaguchi3_Dbl	Holm2_T22_dB
47.	Barnes2_T11_dB	AlphaNeg_Anisotropy	Bhattacharya_Frery_BF4_Hlx_dB
48.	feature6	rvi	Yamaguchi4_Y4O_Hlx_dB
49.	AlphaNeg_Anisotropy	Aghababaece_Tawp_SM3	Holm1_T33_dB
50.	feature5	Yamaguchi4_S4R_Dbl_dB	Aghababaece_Tawp_mean
51.	Singh_i6SD_OD_dB	An_Yang4_Dbl	Yamaguchi4_Y4O_Dbl_dB
52.	Singh_i6SD_CD_dB	feature2	Barnes1_T11_dB
53.	Singh_i6SD_Hlx_dB	AlphaNeg_entropyNeg	Singh_i6SD_Vol_dB
54.	AnisotropyNeg_entropyNeg	Aghababaece_Alphap_SM2	feature2
55.	Neumann_delta_mod	Krogager_Kd	Aghababaece_Phip_SM3
56.	Aghababaece_Tawp_SM2	TSVM_alpha_s2	Freeman2_Vol_dB
57.	Cloude_T22_dB	TSVM_phi_s2	Aghababaece_M_SM2_dB
58.	Aghababaece_Alphap_SM1	TSVM_tau_m	Aghababaece_Alphap_SM2
59.	Huynen_T33_dB	VanZyl3_Vol	Bhattacharya_Frery_BF4_Dbl_dB
60.	Aghababaece_Orientation_max_mean	AlphaNeg_entropy	Aghababaece_Phip_mean
61.	Aghababaece_Tawp_SM3	AnisotropyNeg_entropyNeg	Yamaguchi4_S4R_Odd_dB
62.	Holm2_T22_dB	Aghababaece_Phip_SM2	Singh_i6SD_Dbl_dB
63.	Yamaguchi4_S4R_Hlx_dB	Holm1_T22_dB	Yamaguchi4_Y4O_Odd_dB
64.	Bhattacharya_Frery_BF4_Hlx_dB	Barnes2_T22_dB	Alpha_AnistropyNeg
65.	Yamaguchi4_Y4O_Hlx_dB	T12_imag	Holm1_T22_dB
66.	Holm1_T33_dB	T33	Barnes1_T33_dB
67.	Aghababaece_Tawp_SM1	Freeman2_Vol_dB	Singh_i6SD_Odd_dB
68.	Aghababaece_Tawp_mean	Yamaguchi4_S4R_Odd_dB	Aghababaece_Alphap_SM3
69.	Yamaguchi4_Y4O_Dbl_dB	Yamaguchi4_S4R_Vol_dB	Freeman2_Ground_dB
70.	Barnes1_T11_dB	Freeman_Odd	Barnes1_T22_dB
71.	Singh_i6SD_Vol_dB	Holm2_T11_dB	Bhattacharya_Frery_BF4_Vol_dB
72.	feature2	MCSM_Dbl	Yamaguchi4_S4R_Vol_dB
73.	Yamaguchi4_S4R_Dbl_dB	Alpha_AnistropyNeg	Aghababaece_Alphap_mean
74.	Aghababaece_Phip_SM3	Anisotropy_entropy	Aghababaece_M_SM3_dB
75.	Freeman2_Vol_dB	Aghababaece_Orientation_max_mean	Aghababaece_Phip_SM2
76.	Aghababaece_M_SM2_dB	CCCnorm	An_Yang3_Vol_dB
77.	Aghababaece_Alphap_SM2	An_Yang4_Hlx	Aghababaece_M_SM1_dB
78.	Bhattacharya_Frery_BF4_Dbl_dB	Freeman_Vol	Huynen_T11_dB
79.	Aghababaece_Phip_mean	Neumann_tau	lambda
80.	Yamaguchi4_S4R_Odd_dB	TSVM_alpha_s	Aghababaece_Orientation_max_SM3
81.	Singh_i6SD_Dbl_dB	TSVM_phi_s1	alpha
			TSVM_alpha_s

82.	Yamaguchi4_Y4O_Odd_dB	Anisotropy_entropyNeg	TSVM_alpha_s1 feature4 AlphaNeg_entropy
83.	An_Yang3_Dbl_dB	Aghababaece_M_SM1_dB	
84.	Singh4_G4U1_Dbl_dB	Barnes1_T11_dB	
85.	An_Yang3_Odd_dB	Yamaguchi4_Y4O_Hlx_dB	
86.	Alpha_AnistropyNeg	Holm2_T22_dB	
87.	Singh4_G4U1_Odd_dB	Neumann_delta_pha	
88.	Holm1_T22_dB	feature5	
89.	Barnes1_T33_dB	TSVM_psi3	
90.	Singh_i6SD_Odd_dB	Aghababaece_Alphap_SM1	
91.	Bhattacharya_Frery_BF4_Odd_dB	Barnes1_T22_dB	
92.	Aghababaece_Alphap_SM3	Bhattacharya_Frery_BF4_Vol_dB	
93.	Freeman2_Ground_dB	Arii3_NNED_Odd	
94.	Barnes1_T22_dB	Cloude_T11_dB	
95.	Yamaguchi4_Y4O_Vol_dB	Huynen_T11_dB	
96.	Holm1_T11_dB	Yamaguchi3_Vol	
97.	Aghababaece_Orientation_max_SM2	Alpha_Anistropy	
98.	Bhattacharya_Frery_BF4_Vol_dB	Aghababaece_Orientation_max_SM1	
99.	Singh4_G4U1_Vol_dB	Aghababaece_Orientation_max_SM3	
100.	Yamaguchi4_S4R_Vol_dB	Singh_i6SD_Hlx_dB	
101.	Aghababaece_Alphap_mean	depolarisation_index	
102.	Aghababaece_M_SM3_dB	Arii3_NNED_Dbl	
103.	Aghababaece_Phip_SM2	AnisotropyNeg_entropy	
104.	An_Yang3_Vol_dB	Aghababaece_Phip_SM1	
105.	Aghababaece_M_SM1_dB	Aghababaece_Tawp_mean	
106.	Huynen_T11_dB	Singh4_G4U1_Vol_dB	
107.	lambda	Singh_i6SD_Dbl_dB	
108.	Huynen_T22_dB	Cloude_T33_dB	
109.	Aghababaece_Orientation_max_SM3	feature3	
110.	alpha	Aghababaece_M_SM2_dB	
111.	Alpha_entropy	An_Yang3_Odd_dB	
112.	TSVM_alpha_s	Yamaguchi4_Y4O_Vol_dB	
113.	TSVM_alpha_s1	T11	
115.	feature4	TSVM_alpha_s1	
116.	AlphaNeg_entropy	VanZyl3_Dbl	
117.	Krogager_Ks	Aghababaece_Tawp_SM2	
118.	MCSM_Dbl	An_Yang4_Odd	
119.	MCSM_DblHlx	Arii3_NNED_Vol	
120.	rvi	Cloude_T22_dB	

Ninth set comprises of features which possess $FE_{rank}=[1\ 135]$ and $TCS_{avgrank}=[1\ 135]$ simultaneously. Here the range is considered to be $[1\ 135]$ to denote top 90% of features $[90\% \text{ of } N(151) \cong 135]$. The details of these features are given in Table 12.2.3.10.

Table 12.2.3.10. Features included in the ninth set

Sl.No	Features having $FE_{rank}=[1\ 120]$	Features having $TCS_{avgrank}=[1\ 120]$	List of selected features
1.	An_Yang4_Dbl	Aghababaece_Alphap_SM3	An_Yang4_Dbl An_Yang4_Hlx An_Yang4_Odd An_Yang4_Vol Arii3_NNED_Dbl Arii3_NNED_Odd Arii3_NNED_Vol T12_imag T12_real T13_imag T13_real T23_imag T23_real T23_real Freeman_Dbl Freeman_Odd Freeman_Vol Krogager_Kh MCSM_Hlx MCSM_Vol MCSM_Wire feature 1 VanZyl3_Dbl VanZyl3_Odd VanZyl3_Vol Yamaguchi3_Dbl Yamaguchi3_Odd Yamaguchi3_Vol Alpha_entropyNeg AlphaNeg_entropyNeg Aghababaece_Orientation_max_SM1 conformity scatt_predominance pedestal height Alpha_Anistropy degree_purity
2.	An_Yang4_Hlx	MCSM_Odd	
3.	An_Yang4_Odd	T23_real	
4.	An_Yang4_Vol	Freeman_Dbl	
5.	Arii3_NNED_Dbl	Singh_i6SD_OD_dB	
6.	Arii3_NNED_Odd	MCSM_Vol	
7.	Arii3_NNED_Vol	T13_real	
8.	T12_imag	alpha	
9.	T12_real	entropy	
10.	T13_imag	T23_imag	
11.	T13_real	TSVM_tau_m1	
12.	T23_imag	Bhattacharya_Frery_BF4_Hlx_dB	
13.	T23_real	beta	
14.	Freeman_Dbl	feature4	
15.	Freeman_Odd	conformity	
16.	Freeman_Vol	Aghababaece_Alphap_mean	
17.	Krogager_Kh	Aghababaece_M_SM3_dB	
18.	MCSM_Hlx	An_Yang3_Vol_dB	
19.	MCSM_Vol	Yamaguchi4_Y4O_Dbl_dB	
20.	MCSM_Wire	pedestal height	
21.	feature 1	Aghababaece_Phip_SM3	
22.	VanZyl3_Dbl	Singh_i6SD_Vol_dB	
23.	VanZyl3_Odd	scatt_predominance	
24.	VanZyl3_Vol	T12_real	
25.	Yamaguchi3_Dbl	anisotropy	
26.	Yamaguchi3_Odd	MCSM_Wire	
27.	Yamaguchi3_Vol	TSVM_phi_s	
28.	Alpha_entropyNeg	TSVM_psi2	
29.	AlphaNeg_entropyNeg	Bhattacharya_Frery_BF4_Dbl_dB	
30.	Aghababaece_Orientation_max_SM1	Freeman2_Ground_dB	
31.	Singh4_G4U1_Hlx_dB	Yamaguchi4_Y4O_Odd_dB	
32.	conformity	scatt_diversity	
33.	scatt_predominance	Krogager_Ks	

34.	pedestal height	TSVM_psi	feature3
35.	Alpha_Anistropy	Aghababae_Phip_mean	depolarisation_index
36.	degree_purity	Holm1_T33_dB	Holm2_T11_dB
37.	feature3	Singh_i6SD_Odd_dB	Barnes2_T22_dB
38.	depolarisation_index	An_Yang4_Vol	Cloude_T11_dB
39.	Holm2_T11_dB	T13_imag	CCCnorm
40.	Barnes2_T22_dB	Alpha_entropy	scatt_diversity
41.	Barnes2_T33_dB	Barnes1_T33_dB	feature6
42.	Holm2_T33_dB	Freeman_rvi	AlphaNeg_Anistropy
43.	Cloude_T33_dB	delta	feature5
44.	Cloude_T11_dB	lambda	Singh_i6SD_OD_dB
45.	CCCnorm	MCSM_Hlx	Singh_i6SD_Hlx_dB
46.	scatt_diversity	Yamaguchi3_Dbl	AnisotropyNeg_entropyNeg
47.	Barnes2_T11_dB	AlphaNeg_Anistropy	Neumann_delta_mod
48.	feature6	rvi	Aghababae_Tawp_SM2
49.	AlphaNeg_Anistropy	Aghababae_Tawp_SM3	Cloude_T22_dB
50.	feature5	Yamaguchi4_S4R_Dbl_dB	Aghababae_Alphap_SM1
51.	Singh_i6SD_OD_dB	An_Yang4_Dbl	Aghababae_Orientation_max_mean
52.	Singh_i6SD_CD_dB	feature2	Aghababae_Tawp_SM3
53.	Singh_i6SD_Hlx_dB	AlphaNeg_entropyNeg	Holm2_T22_dB
54.	AnisotropyNeg_entropyNeg	Aghababae_Alphap_SM2	Bhattacharya_Frery_BF4_Hlx_dB
55.	Neumann_delta_mod	Krogager_Kd	Yamaguchi4_Y4O_Hlx_dB
56.	Aghababae_Tawp_SM2	TSVM_alpha_s2	Holm1_T33_dB
57.	Cloude_T22_dB	TSVM_phi_s2	Aghababae_Tawp_mean
58.	Aghababae_Alphap_SM1	TSVM_tau_m	Yamaguchi4_Y4O_Dbl_dB
59.	Huynen_T33_dB	VanZyl3_Vol	Barnes1_T11_dB
60.	Aghababae_Orientation_max_mean	AlphaNeg_entropy	Singh_i6SD_Vol_dB
61.	Aghababae_Tawp_SM3	AnisotropyNeg_entropyNeg	feature2
62.	Holm2_T22_dB	Aghababae_Phip_SM2	Aghababae_Phip_SM3
63.	Yamaguchi4_S4R_Hlx_dB	Holm1_T22_dB	Freeman2_Vol_dB
64.	Bhattacharya_Frery_BF4_Hlx_dB	Barnes2_T22_dB	Aghababae_M_SM2_dB
65.	Yamaguchi4_Y4O_Hlx_dB	T12_imag	Aghababae_Alphap_SM2
66.	Holm1_T33_dB	T33	Bhattacharya_Frery_BF4_Dbl_dB
67.	Aghababae_Tawp_SM1	Freeman2_Vol_dB	Aghababae_Phip_mean
68.	Aghababae_Tawp_mean	Yamaguchi4_S4R_Odd_dB	Yamaguchi4_S4R_Odd_dB
69.	Yamaguchi4_Y4O_Dbl_dB	Yamaguchi4_S4R_Vol_dB	Singh_i6SD_Dbl_dB
70.	Barnes1_T11_dB	Freeman_Odd	Yamaguchi4_Y4O_Odd_dB
71.	Singh_i6SD_Vol_dB	Holm2_T11_dB	Alpha_AnistropyNeg
			Holm1_T22_dB

72.	feature2	MCSM_Dbl	Barnes1_T33_dB
73.	Yamaguchi4_S4R_Dbl_dB	Alpha_AnistropyNeg	Singh_i6SD_Odd_dB
74.	Aghababae_Phip_SM3	Anisotropy_entropy	Aghababae_Alphap_SM3
75.	Freeman2_Vol_dB	Aghababae_Orientation_max_mean	Freeman2_Ground_dB
76.	Aghababae_M_SM2_dB	CCCnorm	Barnes1_T22_dB
77.	Aghababae_Alphap_SM2	An_Yang4_Hlx	Aghababae_Orientation_max_SM2
78.	Bhattacharya_Frery_BF4_Dbl_dB	Freeman_Vol	Bhattacharya_Frery_BF4_Vol_dB
79.	Aghababae_Phip_mean	Neumann_tau	Yamaguchi4_S4R_Vol_dB
80.	Yamaguchi4_S4R_Odd_dB	TSVM_alpha_s	Aghababae_Alphap_mean
81.	Singh_i6SD_Dbl_dB	TSVM_phi_s1	Aghababae_M_SM3_dB
82.	Yamaguchi4_Y4O_Odd_dB	Anisotropy_entropyNeg	Aghababae_Phip_SM2
83.	An_Yang3_Dbl_dB	Aghababae_M_SM1_dB	An_Yang3_Vol_dB
84.	Singh4_G4U1_Dbl_dB	Barnes1_T11_dB	Aghababae_M_SM1_dB
85.	An_Yang3_Odd_dB	Yamaguchi4_Y4O_Hlx_dB	Huynen_T11_dB
86.	Alpha_AnistropyNeg	Holm2_T22_dB	lambda
87.	Singh4_G4U1_Odd_dB	Neumann_delta_pha	Huynen_T22_dB
88.	Holm1_T22_dB	feature5	Aghababae_Orientation_max_SM3
89.	Barnes1_T33_dB	TSVM_psi3	alpha
90.	Singh_i6SD_Odd_dB	Aghababae_Alphap_SM1	TSVM_alpha_s
91.	Bhattacharya_Frery_BF4_Odd_dB	Barnes1_T22_dB	TSVM_alpha_s1
92.	Aghababae_Alphap_SM3	Bhattacharya_Frery_BF4_Vol_dB	feature4
93.	Freeman2_Ground_dB	Arii3_NNED_Odd	AlphaNeg_entropy
94.	Barnes1_T22_dB	Cloude_T11_dB	Krogager_Ks
95.	Yamaguchi4_Y4O_Vol_dB	Huynen_T11_dB	MCSM_Dbl
96.	Holm1_T11_dB	Yamaguchi3_Vol	rvi
97.	Aghababae_Orientation_max_SM2	Alpha_Anistropy	T11
98.	Bhattacharya_Frery_BF4_Vol_dB	Aghababae_Orientation_max_SM1	TSVM_alpha_s2
99.	Singh4_G4U1_Vol_dB	Aghababae_Orientation_max_SM3	Krogager_Kd
100.	Yamaguchi4_S4R_Vol_dB	Singh_i6SD_Hlx_dB	Neumann_delta_pha
101.	Aghababae_Alphap_mean	depolarisation_index	TSVM_psi2
102.	Aghababae_M_SM3_dB	Arii3_NNED_Dbl	AnisotropyNeg_entropy
103.	Aghababae_Phip_SM2	AnisotropyNeg_entropy	MCSM_Odd
104.	An_Yang3_Vol_dB	Aghababae_Phip_SM1	AlphaNeg_AnisotropyNeg
105.	Aghababae_M_SM1_dB	Aghababae_Tawp_mean	Freeman_rvi
106.	Huynen_T11_dB	Singh4_G4U1_Vol_dB	beta
107.	lambda	Singh_i6SD_Dbl_dB	TSVM_tau_m1
108.	Huynen_T22_dB	Cloude_T33_dB	Anisotropy_entropy
109.	Aghababae_Orientation_max_SM3	feature3	Neumann_tau
			TSVM_tau_m

110.	alpha	Aghababaece_M_SM2_dB
111.	Alpha_entropy	An_Yang3_Odd_dB
112.	TSVM_alpha_s	Yamaguchi4_Y4O_Vol_dB
113.	TSVM_alpha_s1	T11
115.	feature4	TSVM_alpha_s1
116.	AlphaNeg_entropy	VanZyl3_Dbl
117.	Krogager_Ks	Aghababaece_Tawp_SM2
118.	MCSM_Dbl	An_Yang4_Odd
119.	MCSM_DblHlx	Arii3_NNED_Vol
120.	rvi	Cloude_T22_dB
121.	T11	Huynen_T22_dB
122.	TSVM_alpha_s2	TSVM_phi_s3
123.	Krogager_Kd	TSVM_tau_m3
124.	Neumann_delta pha	gamma
125.	TSVM_psi2	Neumann_delta_mod
126.	AnisotropyNeg_entropy	TSVM_tau_m2
127.	T22	AlphaNeg_AnisotropyNeg
128.	MCSM_Odd	Alpha_entropyNeg
129.	AlphaNeg_AnisotropyNeg	degree_purity
130.	Freeman_rvi	Krogager_Kh
131.	beta	Bhattacharya_Frery_BF4_Odd_dB
132.	TSVM_tau_m1	feature6
133.	Anisotropy_entropy	TSVM_alpha_s3
134.	Neumann_tau	Yamaguchi3_Odd
135.	TSVM_tau_m	Aghababaece_Orientation_max_SM2

* In Table 12.2.3.2-Table 12.2.3.10, the elements which gets selected in each set are marked with red.

Tenth set comprises of features which possess $FE_{rank}=[1\ 151]$ and $TCS_{avgrank}=[1\ 151]$ simultaneously. Here the range is considered to be $[1\ 151]$ to denote all features. This set comprises of all features given in Table 12.2.3.1.

After generating these ten feature sets, each set is given as input to LeNet 5, FFS-CNN and Crop-Net separately and the performances of each network is compared in each case. Here the results obtained till the sixth set are considered for comparison as it is not possible to obtain the results for all the networks from the seventh set onwards due to hardware constraint. The specifications of the computer which is used for the generation of the results are given below:

1. Processor: Intel Core i7-8700
2. Memory: 16 GB ((1×16 GB)) DDR4-2666 DIMM Memory
3. 64-bit Operating System
4. Graphics Card Information -NVIDIA GeForce RTX 2060 6GB

12.4. Experimental results

Case I:

Set No.	Network	Test Accuracy	Crop Class	Precision	Recall	F1 score
1	LeNet 5	18.51%	Steam beans	0.00	0.00	0.00
			Bare soil	0.00	0.00	0.00
			Rape seed	0.00	0.00	0.00
			Wheat A	0.70	0.02	0.04
			Potatoes	0.00	0.00	0.00
			Wheat B	0.00	0.00	0.00
			Wheat C	0.00	0.00	0.00
			Beet	0.18	1.00	0.31
			Peas	0.00	0.00	0.00
			Barley	0.00	0.00	0.00
			Lucerne	0.00	0.00	0.00

Set No.	Network	Test Accuracy	Crop Class	Precision	Recall	F1 score
1	FFS-CNN	18.93%	Steam beans	1.00	0.00	0.00
			Bare soil	0.00	0.00	0.00
			Rape seed	0.00	0.00	0.00
			Wheat A	0.69	0.04	0.08
			Potatoes	0.97	0.01	0.01
			Wheat B	0.00	0.00	0.00
			Wheat C	0.68	0.00	0.01
			Beet	0.18	1.00	0.31
			Peas	0.00	0.00	0.00
			Barley	0.00	0.00	0.00
			Lucerne	0.81	0.01	0.02

Set No.	Network	Test Accuracy	Crop Class	Precision	Recall	F1 score
1	Crop-Net	18.94%	Steam beans	1.00	0.00	0.01
			Bare soil	0.00	0.00	0.00
			Rape seed	0.00	0.00	0.00
			Wheat A	0.69	0.04	0.08
			Potatoes	1.00	0.01	0.01
			Wheat B	0.00	0.00	0.00
			Wheat C	0.68	0.00	0.01
			Beet	0.18	1.00	0.31

			Peas	0.00	0.00	0.00
			Barley	0.00	0.00	0.00
			Lucerne	0.81	0.01	0.02

Case II:

Set No.	Network	Test Accuracy	Crop Class	Precision	Recall	F1 score
2	LeNet 5	18.53%	Steam beans	0.00	0.00	0.00
			Bare soil	0.00	0.00	0.00
			Rape seed	0.00	0.00	0.00
			Wheat A	0.66	0.02	0.04
			Potatoes	0.12	0.00	0.00
			Wheat B	0.00	0.00	0.00
			Wheat C	0.00	0.00	0.00
			Beet	0.18	1.00	0.31
			Peas	0.00	0.00	0.00
			Barley	0.00	0.00	0.00
			Lucerne	0.00	0.00	0.00

Set No.	Network	Test Accuracy	Crop Class	Precision	Recall	F1 score
2	FFS-CNN	18.30%	Steam beans	0.00	0.00	0.00
			Bare soil	0.00	0.00	0.00
			Rape seed	0.00	0.00	0.00
			Wheat A	0.00	0.00	0.00
			Potatoes	0.00	0.00	0.00
			Wheat B	0.00	0.00	0.00
			Wheat C	0.00	0.00	0.00
			Beet	0.18	1.00	0.31
			Peas	0.00	0.00	0.00
			Barley	0.00	0.00	0.00
			Lucerne	0.00	0.00	0.00

Set No.	Network	Test Accuracy	Crop Class	Precision	Recall	F1 score
2	Crop-Net	18.54%	Steam beans	0.00	0.00	0.00
			Bare soil	0.00	0.00	0.00
			Rape seed	0.00	0.00	0.00
			Wheat A	0.60	0.02	0.04
			Potatoes	0.21	0.00	0.01

			Wheat B	0.00	0.00	0.00
			Wheat C	0.00	0.00	0.00
			Beet	0.18	1.00	0.31
			Peas	0.00	0.00	0.00
			Barley	0.00	0.00	0.00
			Lucerne	0.00	0.00	0.00

Case III.

Set No.	Network	Test Accuracy	Crop Class	Precision	Recall	F1 score
3	LeNet 5	43.35%	Steam beans	0.19	0.01	0.03
			Bare soil	0.49	0.10	0.17
			Rape seed	0.39	0.15	0.22
			Wheat A	0.62	0.65	0.64
			Potatoes	0.35	0.52	0.42
			Wheat B	0.49	0.42	0.45
			Wheat C	0.51	0.60	0.55
			Beet	0.40	0.47	0.43
			Peas	0.29	0.24	0.27
			Barley	0.33	0.19	0.24
			Lucerne	0.47	0.61	0.53

Set No.	Network	Test Accuracy	Crop Class	Precision	Recall	F1 score
3	FFS-CNN	42.95%	Steam beans	0.43	0.03	0.06
			Bare soil	0.32	0.23	0.27
			Rape seed	0.25	0.34	0.29
			Wheat A	0.55	0.73	0.63
			Potatoes	0.44	0.44	0.44
			Wheat B	0.45	0.36	0.40
			Wheat C	0.46	0.66	0.54
			Beet	0.40	0.39	0.40
			Peas	0.39	0.17	0.24
			Barley	0.35	0.32	0.34
			Lucerne	0.64	0.43	0.52

Set No.	Network	Test Accuracy	Crop Class	Precision	Recall	F1 score
3	Crop-Net	66.69%	Steam beans	0.70	0.37	0.49
			Bare soil	0.65	0.43	0.52

			Rape seed	0.61	0.61	0.61
			Wheat A	0.74	0.83	0.78
			Potatoes	0.67	0.66	0.67
			Wheat B	0.68	0.66	0.67
			Wheat C	0.70	0.76	0.73
			Beet	0.65	0.64	0.65
			Peas	0.66	0.52	0.58
			Barley	0.56	0.65	0.60
			Lucerne	0.67	0.85	0.74

Case IV:

Set No.	Network	Test Accuracy	Crop Class	Precision	Recall	F1 score
4	LeNet 5	68.38%	Steam beans	0.44	0.20	0.27
			Bare soil	0.75	0.85	0.80
			Rape seed	0.50	0.61	0.55
			Wheat A	0.75	0.77	0.76
			Potatoes	0.83	0.84	0.83
			Wheat B	0.69	0.57	0.63
			Wheat C	0.73	0.69	0.71
			Beet	0.65	0.76	0.70
			Peas	0.59	0.50	0.54
			Barley	0.56	0.50	0.53
			Lucerne	0.74	0.74	0.74

Set No.	Network	Test Accuracy	Crop Class	Precision	Recall	F1 score
4	FFS-CNN	59.25%	Steam beans	0.33	0.00	0.00
			Bare soil	0.60	0.83	0.70
			Rape seed	0.36	0.49	0.42
			Wheat A	0.65	0.60	0.63
			Potatoes	0.80	0.81	0.81
			Wheat B	0.52	0.52	0.52
			Wheat C	0.57	0.66	0.61
			Beet	0.59	0.71	0.65
			Peas	0.51	0.33	0.40
			Barley	0.48	0.29	0.36
			Lucerne	0.70	0.52	0.60

Set No.	Network	Test Accuracy	Crop Class	Precision	Recall	F1 score
4	Crop-Net	69.27%	Steam beans	0.57	0.12	0.20
			Bare soil	0.74	0.78	0.76
			Rape seed	0.60	0.42	0.49
			Wheat A	0.87	0.73	0.79
			Potatoes	0.80	0.88	0.84
			Wheat B	0.60	0.67	0.63
			Wheat C	0.72	0.74	0.73
			Beet	0.62	0.77	0.69
			Peas	0.58	0.60	0.59
			Barley	0.59	0.52	0.55
			Lucerne	0.81	0.75	0.78

Case V:

Set No.	Network	Test Accuracy	Crop Class	Precision	Recall	F1 score
5	LeNet 5	96.86%	Steam beans	0.96	0.97	0.97
			Bare soil	0.99	0.99	0.99
			Rape seed	0.97	0.97	0.97
			Wheat A	0.99	0.98	0.98
			Potatoes	0.96	0.96	0.96
			Wheat B	0.96	0.98	0.97
			Wheat C	0.98	0.98	0.98
			Beet	0.96	0.96	0.96
			Peas	0.98	0.96	0.97
			Barley	0.93	0.94	0.93
			Lucerne	0.99	0.98	0.99

Set No.	Network	Test Accuracy	Crop Class	Precision	Recall	F1 score
5	FFS-CNN	97.75%	Steam beans	0.98	0.97	0.98
			Bare soil	1.00	0.99	0.99
			Rape seed	0.98	0.99	0.98
			Wheat A	0.98	0.99	0.99
			Potatoes	0.97	0.97	0.97
			Wheat B	0.98	0.98	0.98
			Wheat C	0.99	0.99	0.99
			Beet	0.97	0.97	0.97

			Peas	0.99	0.97	0.98
			Barley	0.95	0.95	0.95
			Lucerne	1.00	0.99	0.99

Set No.	Network	Test Accuracy	Crop Class	Precision	Recall	F1 score
5	Crop-Net	98.33%	Steam beans	0.99	0.99	0.99
			Bare soil	1.00	0.99	0.99
			Rape seed	0.99	0.99	0.99
			Wheat A	0.99	1.00	0.99
			Potatoes	0.96	0.99	0.97
			Wheat B	0.99	0.98	0.99
			Wheat C	0.99	0.99	0.99
			Beet	0.98	0.97	0.98
			Peas	0.99	0.98	0.98
			Barley	0.97	0.96	0.97
			Lucerne	1.00	0.99	1.00

Case VI:

Set No.	Network	Test Accuracy	Crop Class	Precision	Recall	F1 score
6	LeNet5	98.37%	Steam beans	0.97	0.98	0.98
			Bare soil	1.00	0.99	0.99
			Rape seed	0.99	0.99	0.99
			Wheat A	0.99	0.99	0.99
			Potatoes	0.98	0.97	0.97
			Wheat B	0.99	0.99	0.99
			Wheat C	0.99	0.99	0.99
			Beet	0.97	0.98	0.98
			Peas	0.99	0.99	0.99
			Barley	0.97	0.96	0.96
			Lucerne	0.99	0.99	0.99

Set No.	Network	Test Accuracy	Crop Class	Precision	Recall	F1 score
6	FFS-CNN	98.82%	Steam beans	0.98	0.99	0.98
			Bare soil	1.00	0.98	0.99
			Rape seed	1.00	0.99	0.99
			Wheat A	1.00	0.99	1.00
			Potatoes	0.98	0.99	0.98

			Wheat B	0.99	0.99	0.99
			Wheat C	0.99	1.00	0.99
			Beet	0.99	0.98	0.98
			Peas	0.99	0.99	0.99
			Barley	0.98	0.97	0.97
			Lucerne	1.00	0.99	1.00

Set No.	Network	Test Accuracy	Crop Class	Precision	Recall	F1 score
6	Crop-Net	99.44%	Steam beans	0.99	0.99	0.99
			Bare soil	1.00	1.00	1.00
			Rape seed	0.99	1.00	1.00
			Wheat A	1.00	1.00	1.00
			Potatoes	0.99	0.99	0.99
			Wheat B	1.00	1.00	1.00
			Wheat C	1.00	1.00	1.00
			Beet	0.99	0.99	0.99
			Peas	0.99	1.00	1.00
			Barley	0.99	0.99	0.99
			Lucerne	1.00	1.00	1.00

Precision, Recall and F1 score are calculated as stated in [221]. The results given here proves that for each set, Crop-Net outperforms both FFS-CNN and LeNet 5.

Chapter 13

Conclusion & Future scope

In this thesis, three problem statements have been explored because of their immense practical significance and new methods are proposed to solve the drawbacks of the existing methods in each problem statement.

I. Daytime and Nighttime image dehazing

- The novel BF-Fuzzy synergism based daytime image dehazing method is proposed in this thesis focusing on all the crucial aspects of image dehazing. This is by far the first image dehazing method proposed where a set of novel FI rules is designed for performing each aspect of image dehazing which increases the robustness as well as the efficiency of the proposed method.
- Three benchmark databases are designed in this thesis focusing on different aspects of image dehazing problem to enable the researchers to evaluate the original scene restoration capabilities of dehazing methods in varied atmospheric and illumination conditions. GT image corresponding to each hazy image is present in these databases to facilitate researchers to perform qualitative and quantitative analyses of obtained outputs using full-reference and no-reference quantitative parameters.
- RCCT prior designed in this thesis has successfully overcome the drawbacks of CCT prior and proved to be an ideal pre-processing step of existing daytime image dehazing methods as its introduction as a pre-processing step of these methods enable them to dehaze nighttime hazy images effectively alongside daytime hazy images.
- Fuzzy Logic based sky segmentation method proposed in this thesis by exploiting the analogy between the sky segmentation problem and the image dehazing problem also has various real-life applications.

Some shortcomings of the algorithms proposed in this thesis under this problem statement are highlighted as follows:

- RCCT prior designed in this thesis although has successfully overcome the drawbacks of CCT prior but its' performance efficiency is vastly dependent on the image dehazing capability of the method in which it is introduced as a pre-processing step.
- Fuzzy Logic based sky segmentation method proposed in this thesis have achieved quite high efficiency in effectively segmenting sky region of an image from remaining regions but fails to give high quality results when the image contains white buildings as the proposed method erroneously also identifies the white buildings as sky regions because of their high intensities.

Future work on this problem statement can be focused on finding the probable solutions to these shortcomings like designing a method which can successfully shifts its' dehazing parameters based on the characteristics of daytime and nighttime images, so no pre-processing will be required and a single algorithm can effectively dehaze all types of hazy images irrespective of the time they are captured.

In the second case also if any edge-detection method is introduced in the designed Fuzzy Logic based sky segmentation method, then it can be able to differentiate between white buildings and sky regions as buildings usually contains more edges compared to sky regions.

II. Detection of structurally variant erythrocytes

- The feature ensemble method proposed in this part of the thesis is by far the first algorithm which is designed to perform inter-selection among FC, InverseReLU and ReLU features extracted from single CNN, thereby ensuring the selection of features having high information content and significant contribution in performing classification. The creation of feature ensemble by combining selected features extracted from multiple CNN networks having varied depths also further increase the information content of the feature ensemble. This method is designed to solve

the information loss problem inherent in traditional CNN networks occurring due to the use of ReLU as an activation function. In this thesis, the efficiency of the proposed feature ensemble method is validated by performing detection of nine different types of erythrocytes having varied morphology with the created feature ensemble.

This method is designed mainly considering the variations in shapes among several types of poikilocytes but there are some poikilocytes like macrocytes and microcytes which possess almost similar shapes as that of normal erythrocytes but differs in sizes. The designed method will not be able to detect those abnormalities.

Future work in this research area can be focused on designing a method which can detect all types of abnormal erythrocytes. For that blood smear images containing different types erythrocytes captured with similar microscopic resolution are required to train the network.

III. Land-cover classification using full-polarimetric image data

- Degree of purity & Scattering diversity based Advanced Lee filter designed in this thesis has successfully solved the limitations of existing filters as it facilitates the filter to perform despeckling taking into account the similarities in scattering properties between the center pixel (pixel to be filtered) of any patch and its neighboring pixels.
- Ranking method is designed in this thesis to select the features having high information content and significant contribution in performing classification to perform land cover classification.
- Crop-Net is designed in this thesis to perform POLSAR image classification. This network has proved its excellence over popular existing networks like LeNet 5 and FFS-CNN networks in performing multi-class cropland classification.

Crop-Net architecture designed in this thesis can effectively classify crops growing in large fields but its performance should be tested using data from smaller fields growing different crops of almost heights and canopies.

Future work in this research area can be focused on testing the performances of Crop-Net and other such deep neural networks using the data of smaller fields growing different crops of almost heights and canopies.

- In this thesis organization, it is written - “The first part of this thesis is dedicated to “Daytime and Nighttime image dehazing” problem. This part comprises of Chapter 2 to Chapter 7.” However, within the thesis, the Chapter 2-7 have actually been divided in 4 parts i.e. Part-A to Part D. On the other hand, Chapter 8-12 have not been segregated in parts. In this context, I suggest you to keep consistency of such statement with the actual thesis organization. Rather, I believe, the thesis should be segregated in three parts as written in the ‘thesis organization’ based on the domain of their applications.

References

- [1] H. Koschmieder, "Theorie der horizontalen sichtweite". Beitr. Phys. Freien Atm., vol.12:33–53, pp.171-181,1924.
- [2] <https://www.hindustantimes.com/delhi-news/winter-driving/story-CZ1SCqhX2J7kdFFnJb13MJ.html>
- [3] <https://www.nationalheraldindia.com/national/12-killed-in-separate-up-road-accidents-due-to-dense-fog-on-friday>
- [4] https://ops.fhwa.dot.gov/weather/q1_roadimpact.htm
- [5] M. Abdel-Atey, A-A.Ekram, H.Huang and K.Choi, "A study on crashes related to visibility obstruction due to fog and smoke". Accident Analysis and Prevention, vol. 43, pp. 1730-1737, 2011.
- [6] [36 hurt in fog-related accidents - Newspaper - DAWN.COM](#)
- [7] [China highway crash: At least three dead in a 43-vehicle pileup in thick fog | Daily Mail Online](#)
- [8] V.Ratanavaraha and S.Suangka, "Impacts of accident severity factors and loss values of crashes on expressways in Thailand". IATSS Research, vol.37, pp. 130-136,2014.
- [9] M.D.Keall, W.J.Frith and T.L. Patterson, "The contribution of alcohol to night time crash risk and other risks of night driving". Accident Analysis and Prevention, vol.37, pp. 816-824, 2005.
- [10] R.Tay, S.M.Rifaat and H.C.Chin "A logistic model of the effects of roadway, environmental, vehicle, crash and driver characteristics on hit-and-run crashes". Accident Analysis and Prevention, vol.40, pp. 1330-1336, 2008.
- [11] J. Long J, Z.W. Shi Z, W.Tang and C.S. Zhang, "Single remote sensing image dehazing". IEEE Geoscience Remote Sensing Letters, VOL. 11(1), pp. 59–63, 2014.
- [12] A. Makarau, R. Richter, R.Muller and P. Reinartz, "Haze detection and removal in remotely sensed multispectral imagery". IEEE Trans. Geoscience Remote Sensing, vol. 52(9), pp. 5895–5905, 2014.
- [13] X.X. Pan, F.Y. Xie, Z.G. Jiang and J.H. Yin, "Haze removal for a single remote sensing image based on deformed haze imaging model". IEEE Signal Processing Letters, vol. 22(10), pp.1806–1810, 2015.
- [14] A. Sabu A, N. Vishwanath, "An improved visibility restoration of single haze images for security surveillance systems". IEEE Online Int. Conf. on Green Engineering and Technologies, Coimbatore, India, 2016.
- [15] <https://www.cancer.gov/publications/dictionaries/cancer-terms/def/red-blood-cell>
- [16] <https://www.hematology.org/education/patients/blood-basics>
- [17] <https://www.redcrossblood.org/donate-blood/dlp/whole-blood.html>
- [18] <https://www.who.int/health-topics/anaemia>
- [19] <https://www.nhp.gov.in/disease/gynaecology-and-obstetrics/anaemia-during-pregnancy-maternal-anemia>
- [20] O. Didzun, J.-W. D. Neve, A. Awasthi, M. Dubey, M. Theilmann, T. Bärnighausen, S. Vollmer and P. Geldsetzer, "Anaemia among men in India: a nationally representative cross-sectional study". Lancet Global Health, vol.7, pp. E1685-e1694, 2019.
- [21] <https://scinfo.org/2020/03/17/sickle-cell-news-for-march-2020>
- [22] <https://www.who.int/bulletin/volumes/98/12/20-253583/en/>
- [23] <https://www.afro.who.int/health-topics/sickle-cell-disease>
- [24] <https://www.cdc.gov/ncbddd/sicklecell/data.html>
- [25] https://www.cancertherapyadvisor.com/home/decision-support-in-medicine/hematology_hereditary-elliptocytosis-and-hereditary-pyropoikilocytosis
- [26] <https://rare-diseases.org/rare-diseases/neuroacanthocytosis/>
- [27] https://crisp.nus.edu.sg/~research/tutorial/sar_int.htm
- [28] L.A. Zadeh, "Fuzzy sets". Information and Control, vol. 8:338-353,1965.
- [29] A.Konar, "Computational Intelligence Principles, Techniques and Applications". Springer, 2005.
- [30] Y. Liu Y, K.M. Passino, "Biomimicry of Social Foraging Bacteria for Distributed Optimization: Models, Principles, and Emergent Behaviors". Journal of Optimization Theory and Applications, vol. 115, pp. 603-628, 2002.

- [31] R.N. Khushaba, A. Al-Jumaily, and A. AL-Ani, "Novel feature extraction method based on fuzzy entropy and wavelet packet transform for myoelectric control". *Int. Symp. Communications and Information Technologies*, Sydney, Australia, pp. 352–357, 2007.
- [32] N.Cristianini and E.Ricci, "Support Vector Machines". *Encyclopedia of Algorithms*, 2008 Edition.
- [33] G-B. Huang, Q-Y. Zhu and C-K. Siew, "Extreme Learning Machine: A New Learning Scheme of Feedforward Neural Networks". *Neurocomputing*, vol.70, pp. 480-501, 2008.
- [34] S.G. Narasimhan and S.K. Nayar, "Interactive (De) Weathering of an Image using Physical Models". *IEEE Workshop on Color and Photometric Methods in Computer Vision*, Nice, France, pp. 1-8,2003.
- [35] S.K. Nayar S.K AND s.g. Narasimhan, " Vision in Bad Weather". *Int. Conf. Computer Vision*, Kerkyra, Greece, pp. 1-8, 1999.
- [36] Y.Y. Schechner, S.G Narasimhan S.G and S.K. Nayar, " Instant Dehazing of Images Using Polarization". *Proc. Computer Vision and Pattern Recognition*, Kauai, USA, pp. 1-8, 2001.
- [37] S. Shwartz, E. Namer and Y.Y. Schechner, "Blind Haze Separation". *Proc. Computer Vision and Pattern Recognition*, New York, USA, pp. 1984-1991, 2006.
- [38] K. He, J. Sun and X. Tang X, "Single Image Haze Removal Using Dark Channel Prior". *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 33, pp.2341-2353,2011.
- [39] A. Levin, D.Lischinski and Y.Weiss, "A Closed Form Solution to Natural Image Matting". *IEEE Trans Pattern Analysis and Machine Intelligence*, vol. 30 (2), pp.228 – 242, 2008.
- [40] Y.Gao, H.Chen, H.Li and W.Zhang, " Single image dehazing using local linear fusion". *IET Image Processing*, vol. 12, pp.637-643,2018.
- [41] C.O. Ancuti and C.Ancuti, "Single Image Dehazing by Multi-Scale Fusion". *IEEE Trans. Image Processing*, vol. 22, pp. 3271-3282,2013.
- [42] D-W.Jang and R-H. Park, "Colour image dehazing using near-infrared fusion". *IET Image Processing*, vol. 11, pp. 587-594, 2017.
- [43] J-P.Tarel and N.Hautière, "Fast visibility restoration from a single color or gray level image". *Int. Conf. Computer Vision*, Japan, pp. 1-8, 2009.
- [44] G.Meng,Y. Wang, J. Duan, S. Xiang and C.Pan, "Efficient Image Dehazing with Boundary Constraint and Contextual Regularization". *Int. Conf. Computer Vision*, Sydney, Australia, pp. 617-624,2013.
- [45] C.Xiao and J. Gang, "Fast image dehazing using guided joint bilateral filter". *The Visual Computer: Int. Journal of Computer Graphics*, vol. 28, pp.713-721, 2012.
- [46] J-H. Kim, W-D.Jang, J-Y.Sim and C-S Kim, " Optimized contrast enhancement for real-time image and video dehazing". *J. Vis. Commun. Image R.* vol. 24, pp.410– 425, 2013.
- [47] S.Salazar-Colores, I. Cruz-Aceves and J-M.Ramos-Arreguin, "Single image dehazing using a multilayer perceptron". *J. of Electronic Imaging*, vol. 27, pp.043022-7 - 043022-11, 2018.
- [48] R. Fattal, " Dehazing using Color-Lines". *ACM Trans. Graphics*, vol. 34, pp. 1-14,2014.
- [49] D. Berman, T.Treibitz and S. Avidan, " Non-Local Image Dehazing". *Int. Conf. Computer Vision and Pattern Recognition*, USA, pp. 1-9,2016.
- [50] Q.Zhu, J.Mai and L.Shao, " A Fast Single Image Haze Removal Algorithm Using Color Attenuation Prior". *IEEE Trans. Image Processing*, vol. 24, pp. 3522 – 3533,2015.
- [51] K.Tang, J.Yang and J. Wang, "Investigating Haze-Relevant Features in A Learning Framework for Image Dehazing". *Int. Conf. Computer Vision and Pattern Recognition*, USA, pp. 1-8, 2014.
- [52] B.Cai, X. Xu, K.Jia, C.Qing and D.Tao, "DehazeNet: An End-to-End System for Single Image Haze Remova". *arXiv:1601.07661v2* pp.1-13,2016.
- [53] W.Ren, S. Liu, H.Zhang, J.Pan, C. Xiaochun, M-H. Yang, "Single Image Dehazing via Multi-Scale Convolutional Neural Networks". *European Conf. Computer Vision*, Netherlands, pp. 1-16,2016.
- [54] B.Li, X.Peng, Z.Wang, J.Xu and D.Feng, "AOD-Net: All-in One Dehazing Network". *Proc. Int. Conf. Computer Vision*, Italy, pp. 4780-4788, 2017.
- [55] I.Yoon, S. Kim, D. Ki, M.H. Hayas and J.Paik, "Adaptive Defogging with Color Correction in the HSV Color Space for Consumer Surveillance System". *IEEE Trans. Consumer Electronics*, vol. 58, pp. 111-116, 2012.
- [56] T. Zhang, H-M. Hu and B. Li, "A Naturalness Preserved Fast Dehazing Algorithm Using HSV Color Space". *IEEE Access*, vol. 6, pp. 10644 – 10649, 2018.

- [57] Y. Wan and Q. Chen, "Joint Image Dehazing and Contrast Enhancement using the HSV Color Space". Int. Conf. Visual Communications and Image Processing, Singapore, pp. 1-4, 2015.
- [58] J. Zhang, Y. Cao, Z. Wang, "Nighttime haze removal based on a new imaging model". IEEE International Conference on Image Processing (ICIP), Paris, France, pp. 4557-4561, 2014.
- [59] Y. Li, R.-T. Tan and M.S. Brown, "Nighttime Haze Removal with Glow and Multiple Light Colors". IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, pp. 226-234, 2015.
- [60] S.G. Narasimhan and S.K. Nayar, "Shedding light on the weather". IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), Madison, WI, USA, 2003.
- [61] S. Kuanar, K.R. Rao, D. Mahapatra, M. Bilas, "Night Time Haze and Glow Removal using Deep Dilated Convolutional Network". arXiv:1902.00855v1, pp. 1-13, 2019.
- [62] J. Lin, X. Zhang, H. Li and Z. Liu, "Nighttime image haze removal and enhancement based on improved atmospheric scattering model". International Conference on Image, Video Processing and Artificial Intelligence, Shanghai, China, pp. 10836:1-6, 2018.
- [63] L. Xu, Q. Yan, Y. Xia and J. Jia, "Structure extraction from texture via relative total variation". ACM Trans on Graphics (TOG), vol. 31(6), pp. 139:1-139:10, 2012.
- [64] S. Santra and B. Chanda, "Day/Night Unconstrained Image Dehazing". IEEE International Conference on Pattern Recognition (ICPR), Cancun, Mexico, 2016.
- [65] Y. Liao, Z. Su, X. Liang and B. Qu, "HDP-Net: Haze Density Prediction Network for Nighttime Dehazing". Pacific Rim Conference on Multimedia (PCM), Hefei, China, pp. 469-480, 2018.
- [66] S.-C. Pei and T.-Y. Lee, "Nighttime haze removal using color transfer pre-processing and Dark Channel Prior". IEEE International Conference on Image Processing (ICIP), Orlando, FL, USA, pp. 957-960, 2012.
- [67] E. Reinhard, M. Adhikhmin, B. Gooch and P. Shirley, "Color Transfer between Images". IEEE Trans Computer Graphics and Applications, vol. 21(5), pp. 31-41, 2001.
- [68] K. He K, J. Sun and X. Tang, "Guided Image Filtering". IEEE Trans Pattern Analysis and Machine Intelligence, vol. 35 (6), pp. 1397 – 1409, 2013.
- [69] R. Schettini R, F. Gasparini F, S. Corchs S, F. Marini, "Contrast image correction method". J Electronic Imaging, vol. 19(2), pp. 023005-1 – 023005-11, 2010.
- [70] B. Jiang, H. Men, Z. Ma, L. Wang, Y. Zhou, X. Pengfei, X. Jiang and X. Meng, "Nighttime image Dehazing with modified models of color transfer and guided image filter". Multimedia Tools and Applications, vol. 77 (3), pp. 3125–3141, 2018.
- [71] F. Kou F, W. Chen, C. Wen and Z. Li, "Gradient domain guided image filtering". IEEE Trans Image Process, vol. 24(11), pp. 4528–4539, 2015.
- [72] J. M.S. Prewitt, "Object Enhancement and Extraction". Picture processing and Psychopictorics. Academic Press, 1970.
- [73] C.O. Ancuti, C. Ancuti, C.D. Vleeschouwer and M. Sbrer, "Color Channel Transfer for Image Dehazing". IEEE Signal Processing Letters, vol. 26 (9), pp. 1413:1417, 2019.
- [74] R. Achanta, S. Hemamiz, F. Estraday and S. Susstruncky, "Frequency tuned salient region detection". IEEE International Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, pp. 1597-1604, 2009.
- [75] C. Ancuti, C.O. Ancuti, C.D. Vleeschouwer, A.C. Bovik, "Night-time dehazing by fusion". IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, pp. 2256-2260, 2016.
- [76] J. Zhang, Y. Cao and Z. Wang, "Nighttime Haze Removal with Illumination Correction". arXiv:1606.01460v1, 2016.
- [77] M. Elad, "Retinex by two bilateral filters". International Conference on Scale-Space Theories in Computer Vision (Springer), Hofgeismar, Germany, pp. 217-229, 2005.
- [78] L. Wang, L. Xiao, H. Liu and Z. Wei, "Variational Bayesian Method for Retinex". IEEE Trans Image Processing, vol. 23(8), pp. 3381–3396, 2014.
- [79] J. Zhang, Y. Cao, S. Fang, Y. Kang and C. W. Chen, "Fast Haze Removal for Nighttime Image Using Maximum Reflectance Prior". IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, pp. 7418-7426, 2017.
- [80] Y. Li and M.S. Brown, "Single image layer separation using relative smoothness". IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 2014.
- [81] D. Park, D.K. Han and H. Ko, "Nighttime image dehazing using local atmospheric selection rule and weighted entropy for visible-light systems". Optical Engineering, vol. 56(5), pp. 05050-1-05050-4, 2017.
- [82] P. Strobach, "Quadtree-structured recursive plane decomposition coding of images". IEEE Trans Signal Processing, vol. 39(6), pp. 1380–1397, 1991.

- [83] D.Park, D.K.Han and H.Ko, "Nighttime image dehazing with local atmospheric light and weighted entropy". IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, pp. 2261- 2265,2016.
- [84] X.Jin, X.Yan, J.Zhang and Z. Li, "Nighttime Haze Removal with Fusion Atmospheric Light and Improved Entropy". Chinese Conference on Computer Vision (CCF), Tianjin, China, pp. 323-333,2017.
- [85] M.Yang, J.Liu and Z. Li, "Super-pixel Based Single Nighttime Image Haze Removal". IEEE Trans Multimedia, vol. 20 (11), pp. 3008 – 3018, 2018.
- [86] R.Achanta, A.Shaji, K.Smith, A. Lucchi and P.Fua, "SLIC superpixels compared to state-of-the-art superpixel methods". IEEE Trans Pattern Analysis and Machine Intelligence, vol. 34(11), pp. 2274-2282, 2012.
- [87] L. Xu, Q.Yan, Y. Xia and J.Jia (2012) Structure extraction from texture via relative total variation. ACM Trans on Graphics, vol. 31(6), pp.139:1-139:10, 2012.
- [88] C.D. Boor, "Bicubic Spline Interpolation". Journal of Mathematics and Physics, vol. 41, pp. 212–218 1962.
- [89] I.Omer and M. Werman, "Color lines: image specific color representation". IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Washington, DC, USA,2004.
- [90] S. Santra and B. Chanda, "Single image dehazing with varying atmospheric light intensity". IEEE National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG), Patna, India,2015.
- [91] T.Yu, K.Song, P.Miao, G.Yang, H.Yang and C. Chen, "Nighttime Single Image Dehazing via Pixel-Wise Alpha Blending". IEEE Access, vol. 7, pp. 114619 – 114630,2019.
- [92] S.Sun and X.Guo, "Image Enhancement Using Bright Channel Prior". IEEE International Conference on Industrial Informatics - Computing Technology, Intelligent Technology, Industrial Information Integration (ICIICII), Wuhan, China,2017.
- [93] E.H. Land, " The retinex theory of color vision". Scientific American, vol. 237(6), pp. 108–128, 1977.
- [94] RC.Gonzalez and R.E.Woods, "Digital Image processing". Pearson Education Inc,2008.
- [95] O.P.Verma, M. Handmandlu, A.K.Sultania and A.S.Parihar, " A novel fuzzy system for edge detection in noisy image using bacterial foraging". Multidim Syst Sign Process, vol. 24, pp.181–198,2013.
- [96] H.A. David , H.N. Nagaraja, "Order Statistics". Wiley series in probability and statistics, 3rd Edition, 1970.
- [97] M.R.Rakesh, B.Ajeya and A.R. Mohan, "Hybrid Median Filter for Impulse Noise Removal of an Image in Image Restoration". Int. Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, vol. 2 (10), pp. 5117-5124.,2013.
- [98] M.S. Darus, S. N.Sulaiman, I. S. Isa, Z.Hussain, N. M. Tahir, N. A. M. Isa, "Modified Hybrid Median Filter for Removal of Low Density Random-Valued Impulse Noise in Images". IEEE Int. Conf. on Control System, Computing and Engineering. Penang, Malaysia, pp. 528-533,2016.
- [99] H. Hwang, R.A. Haddad, Adaptive median filters: new algorithms and results, IEEE Trans. on Image Processing, vol. 4(4), pp.499-502,1995.
- [100] Z. Wang and D. Zhang, "Progressive switching median filter for the removal of impulse noise from highly corrupted images". IEEE Trans. Circuits & Systems II: Analog & Digital Signal Processing, vol. 46(1), pp. 78-80,1990.
- [101] S. T. Boo, H. Ibrahim and K. K.V. Toh, "An Improved Progressive Switching Median Filter". Int. Conf. on Future Computer and Communication, Kuala Lumpur, Malaysia, pp. 136-139,2009.
- [102] [https:// eddie.via.cornell.edu/cgi-bin/datac/logon.cgi](https://eddie.via.cornell.edu/cgi-bin/datac/logon.cgi), Accessed 3 August, 2019.
- [103] A.Hore and D.Ziou, "Image Quality Metrics: PSNR vs. SSIM". Int. Conf. on Pattern Recognition, Istanbul, Turkey, pp.2366-2369,2010.
- [104] N.R.Pal and S.K.Pal, "Entropy: A New Definition and its Applications ". IEEE Trans. on systems, man and cybernetics, vol. 21(5),pp.1260-1270,1991.
- [105] K. Ma, W. Liu and Z Wang, "Perceptual evaluation of single image dehazing algorithms". Int. Conf. Image Processing, Québec, Canada, pp. 3600- 3604, 2015.
- [106] vision.middlebury.edu/stereo/, Accessed 21 October 2019
- [107] C.Ancuti and C.O. Ancuti, " D-HAZY: a dataset to evaluate quantitatively dehazing algorithms". Int. Conf. Image Processing, Arizona, pp. 2226-2230, 2016.
- [108] P.Kohli, N.Silberman, D.Hoiem and R.Fergus, "Indoor segmentation and support inference from rgbd images". European Conference on Computer Vision, Florence, Italy, pp. 1-5,2012.
- [109] C.O. Ancuti, C.Ancuti, R.Timofte and C.D.Vleeschouwer, "I-HAZE: a dehazing benchmark with real hazy and haze-free indoor images". arXiv:1804.05091v1, pp. 1-5,2018.
- [110] C.O. Ancuti, C.Ancuti, R.Timofte and C.D. Vleeschouwer, "O-HAZE: a dehazing benchmark with real hazy and haze-free outdoor images". arXiv: 1804.05101v1, pp. 1-9,2018.

- [111] Z.Wang, A.C. Bovik, H.R.Sheikh and E.P. Simoncelli, “ Image quality assessment: From error visibility to structural similarity”. IEEE Trans. Image Processing, vol. 13, pp.600-612,2004.
- [112] G.Sharma, W.Wu and E.Dalal, “The ciede2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations”. Color Research and Applications, vol. 30, pp. 21-30,2005.
- [113] G. Zhai G, K. Gu K, Y. Zhu Y, J. Zhou J, G. Guo, X. Yang X, X. Guan X, W. Zhang, “ Quality Evaluation of Image Dehazing Methods Using Synthetic Hazy Images”. IEEE Transactions on Multimedia, vol. 21(9), pp. 2319-2333, 2019.
- [114] D. Zhengguo, “A New Visibility Metric for Haze Images”. <https://www.mathworks.com/matlabcentral/fileexchange/33529-a-new-visibility-metric-for-haze-images>. Accessed 20 February, 2020.
- [115] L.K. Choi, J.You and A.C. Bovik, “ Referenceless Prediction of Perceptual Fog Density and Perceptual Image Defogging”. IEEE Trans Image Processing, VOL. 24 (11), PP.3888 – 3901,2015.
- [116] A.Mittal, A.K. Moorthy and A.C. Bovik, “No-reference image quality assessment in the spatial domain”. IEEE Trans Image Process, vol. 21(12), pp.4695–4708,2012.
- [117] A.Galdran, “Image dehazing by artificial multiple-exposure image fusion”, Signal Processing, vol. 149, pp. 135-147,2018.
- [118] D. Ngo, S. Lee, Q-H. Nguyen, T.M. Ngo, G-D. Lee and B. Kang , “Single Image Haze Removal from Image Enhancement Perspective for Real-Time Vision-Based Systems”, Sensors, vol. 20(18), 2020.
- [119] <https://in.mathworks.com/matlabcentral/fileexchange/46147-single-image-haze-removal-using-dark-channel-prior>, Accessed 3 March, 2019.
- [120] <https://github.com/JiamingMai/Color-Attenuation-Prior-Dehazing>, Accessed 3 March, 2019.
- [121] <http://www.sciweavers.org/read/matlab-source-code-for-visibility-restoration-from-a-single-image-184349>, Accessed 3 March, 2019.
- [122] <https://github.com/gfmeng/imagede haze>, Accessed 3 March, 2019.
- [123] <https://github.com/caibolun/DehazeNet>, Accessed 3 March, 2019.
- [124] <https://sites.google.com/site/renwenqi888/research/dehazing/mscndehazing>, Accessed 3 March, 2019.
- [125] <https://in.mathworks.com/matlabcentral/fileexchange/68553-single-image-dehazing-using-a-multilayer-perceptron>, Accessed 3, March 2019.
- [126] <https://www.mathworks.com/matlabcentral/fileexchange/66634-artificial-multiple-exposure-fusion-for-image-dehazing>, Accessed 17, March 2021.
- [127] <https://github.com/v1t0ry/mIFDH>, Accessed 17, March 2021.
- [128] N. Hautiere, J.-P. Tarel, D. Aubert and E. Dumont, “Blind Contrast Restoration Assessment by Gradient Ratioing at Visible Edges”. Int. Congress for Stereology, Saint Etienne, France, 2007.
- [129] J.He, C.Zhang, R.Yang and K.Zhu, “Convex optimization for fast image dehazing.” Int. Conf. on Image Processing, USA, pp. 1-5,2016.
- [130] M.Pharr and G.Humphreys, “ Physically Based Rendering: From Theory to Implementation”. Morgan Kaufmann, San Francisco,2010.
- [131] J. Shin, M. Kim, J. Paik, and S. Lee, “Radiance-Reflectance Combined Optimization and Structure-Guided L0-Norm for Single Image Dehazing.” IEEE Transactions on Multimedia, vol. 22(1), pp.30-44,2020.
- [132] Z. Li, X. Zheng, B. Bhanu, S. Long, Q. Zhang and Z. Huang, “Fast Region-Adaptive Defogging and Enhancement for Outdoor Images Containing Sky”.Int. Conf. on Pattern Recognition, Italy, 2021.
- [133] M. Ju, C. Ding, W. Ren, Y. Yang, D. Zhang and Y. Guo, “Fast Region-Adaptive Defogging and Enhancement for Outdoor Images Containing Sky”.Int. Conf. on Pattern Recognition, Italy, 2021.
- [134] S.M. Mazalan, N.M. Mahmood and M.A.A. Razak, ‘Automated Red Blood Cells Counting in Peripheral Blood Smear Image Using Circular Hough Transform’. Int. Conf. Artificial Intelligence, Modelling and Simulation, Kota Malaysia, pp.320-324,2013.
- [135] N. Cherabiti, F. Zohra Chelali, and A. Djeradi, “Circular Hough Transform for Iris localization,” Science and Technology, vol. 2(5), pp. 114–121, 2012.
- [136] B. Venkatalakshmi and K. Thilagavathi, K, “Automatic red blood cell counting using hough transform”. Int. Conf. Information & Communication Technologies, India, pp.267-271,2013.
- [137] Y.M. Alomari, S.N.H.S. Abdullah, R.Z. Azma and K.Omar, “Automatic Detection and Quantification of WBCs and RBCs Using Iterative Structured Circle Detection Algorithm”. Computational and Mathematical Methods in Medicine, pp. 1-17,2014.

- [138] D.Cruz, C. Jennifer, J. Valiente J., et al. "Determination of blood components (WBCs, RBCs, and Platelets) count in microscopic images using image processing and analysis". Int. Conf. Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management, Philippines, pp. 1-7,2017.
- [139] H. Berge, D.Taylor, S. Krishnan and T.S.Douglas, "Improved red blood cell counting in thin blood smears". Int. Sym. Biomedical Imaging: From Nano to Macro, , USA, pp. 204-207,2011.
- [140] R. Tomari, W.N.W Zakaria, M.M.A Jamil, et al., "Computer Aided System for Red Blood Cell Classification in Blood Smear Image". Proc. Computer Science, vol. 42, pp. 206-213,2014.
- [141] C. Rongtai, W. QingXiang, Z. Rui, F. Lijuan, and R. Chengmei, "Red blood cell segmentation using Active Appearance Model". Int. Conf. on Signal processing, China, pp. 1641-1644,2012.
- [142] N. Otsu, "A Threshold Selection Method from Grat-Level Histograms", IEEE Transaction on System, Man and Cybernetics, vol. 9(1), pp. 62-66, 1979.
- [143] M. Veluchamy, K. Perumal and T. Ponuchamy, "Feature Extraction and Classification of Blood Cells Using Artificial Neural Network." American J. Applied Sciences, vol. 9, (5), pp. 615-619,2012.
- [144] M. Buscema, "Back Propagation Neural Networks". Substance Use & Misuse, vol. 33(2), pp.233-270,1998.
- [145] S. Bhowmick, D.K. Das and A.K. Maiti, "Structural and textural classification of erythrocytes in anaemic cases: A scanning electron microscopic study". *Micron*, vol. 44, pp. 384-394,2013.
- [146] P. Dalvi and N. Vernekar, "Computer aided detection of abnormal red blood cells". Int. Conf. Recent Trends in Electronics, Information & Communication Technology, India, pp. 1741-1746,2016.
- [147] M.Maity, T. Mungle, D.Dhane, A.K. Maity and C.Chakraborty, "An Ensemble Rule Learning Approach for Automated Morphological Classification of Erythrocytes". *J Med Syst*, vol.41, (4), pp. 41-56, 2017.
- [148] J.R.Quinlan, "Induction of decision trees." Mach. Learn., vol. 1(1), pp.81– 106, 1986.
- [149] S.H. Shirazi, S.H., A.I. Umar, A.I., N. Haq, et al. , "Extreme learning machine based microscopic red blood cells classification." Cluster Computing, vol. 21, (1), pp. 691-701,2018.
- [150] J.Tang, C. Deng and G-B Huang, "Extreme learning machine for multilayer perceptron". IEEE Transactions on Neural Networks and Learning Systems, vol. 27(4), pp. 1–13,2015.
- [151] M.Lofii, B. Nazari, S. Sadri and N.K.Shichani, "The detection of Dacrocyte, Schistocyte and Elliptocyte cells in Iron Deficiency Anemia". Int. Conf. Pattern Recognition and Image Analysis, Iran, pp. 1-5,2015.
- [152] G. Guo, H. Wang , D. Bell, Y. Bi, and K. Greer, "KNN Model-Based Approach in Classification",2012.
- [153] D.Tyas, D., T. Ratnaningsih, A. Harjoko, A., et al., "The Classification of Abnormal Red Blood Cell on The Minor Thalassemia Case Using Artificial Neural Network and Convolutional Neural Network". Int. Conf. Video and Image Processing, Singapore, pp. 228-233,2017.
- [154] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient based learning applied to document recognition." Proc. IEEE, vol. 86(11), pp. 2278–2324, 1998.
- [155] H. Lee, H., P. Y-P. Chen, "Cell morphology based classification of red cells in blood smear images".Pattern Recognition Letters, vol. 49, pp. 155-161,2014.
- [156] T.Go, H. Byeon, J.L. Lee, "Label-free sensor for automatic identification of erythrocytes using digital in-line holographic microscopy and machine learning". Biosensors and Bioelectronics, vol. 103, pp. 12-18,2018.
- [157] M. Xu, D.P. Papageorgiou, S.Z. Abidi, S.Z., et al., "A deep convolutional neural network for classification of red blood cells in sickle cell anemia". PLoS. Comput. Biol., vol. 13, (10), pp. 1-27,2017.
- [158] Y.Dong, Z. Jiang, H. Shen, et al., "Evaluations of deep convolutional neural networks for automatic identification of malaria infected cells". Int. Conf. Biomedical & Health Informatics, USA, pp. 101-104, 2017.
- [159] A.Krizhevsky,I.Sutskever and G. E. Hinton,"ImageNet Classification with Deep Convolutional Neural Networks".Advances in Neural Information Processing Systems, pp. 1-9,2012.
- [160] C. Szegedy, W. Liu, Y. Jia, *et.al.*, " Going Deeper with Convolutions". arXiv:1409.4842v1 , pp.1-12 ,2014.
- [161] T.J.S. Durant, E.M. Olson and W.L. Schulz, "Very Deep Convolutional Neural Networks for Morphologic Classification of Erythrocytes". *Clinical Chemistry*, vol.63, (12), pp. 1-9,27.
- [162] G. Huang, Z. Liu, and L. Maaten, et. al., "Densely Connected Convolutional Networks". arXiv:1608.06993v5, 2017.
- [163] M. Zang, L. Xiang, X. Mengjia, et al., "Image segmentation and classification for sickle cell disease using deformable U-NET". arXiv:1710.08149v3, 2017.

- [164] O. Ronneberger, P. Fischer, T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation". arXiv:1505.04597v1, 2015.
- [165] A. Kihm, L. Kaestner, C. Wagner, et al., "Classification of red blood cell shapes in flow using outlier tolerant machine learning". PLoS. Comput. Biol., pp. 1-15, 2018.
- [166] L. Maas, A.Y. Hannun, A.Y. Ng., "Rectifier nonlinearities improve neural network acoustic models". *Proc. icml*, 30(1), pp.1-6, 2013.
- [167] K. He, X. Zhang, S. Ren, et al., "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet classification". arXiv:1502.01852v1, 2015.
- [168] D-A. Clevert, T. Unterthiner, S. Hochreiter, "Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)". arXiv:1511.07289v5, 2016.
- [169] W. Shang, K. Shon, D. Almeida, et al., "Understanding and Improving Convolutional Neural Networks via Concatenated Rectifier Linear Units". arXiv:1603.05201v2, 2016.
- [170] X. Dong, G. Kang, K. Zhan, et al., "EraseReLU: A Simple Way to Ease the Training of Deep Convolution Neural Networks". arXiv:1709.07634v2, 2017.
- [171] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition". arXiv:1409.1556v6, 2015.
- [172] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation". arXiv:1311.2524v5, 2014.
- [173] <https://www.shutterstock.com>, accessed 1 June 2018
- [174] <https://www.gettyimages.in>, accessed 1 June 2018
- [175] <https://bigpictureeducation.com>, accessed 1 June 2018
- [176] R.D. Labati, V. Piuri and F. Scotti, "All-IDB: The acute lymphoblastic leukemia image database for image processing". Int. Conf. Image Processing, Belgium, , pp.2045-2048, 2011.
- [177] 'Atlas of Hematology', <http://hematologyatlas.com>, accessed 1 June 2018
- [178] 'American Society of Hematology', <https://imagebank.hematology.org/>, accessed 2 July 2018
- [179] <https://librarymed.utah.edu>, accessed 24 June 2018
- [180] 'ImageNet', www.image-net.org/ accessed 1 May 2018
- [181] M. A. Cauchy, "Methode gen'erale pour la r' esolution des systemes ` d'equations simultanees". Comptes Rendus Hebd. Seances Acad. Sci. vol. 25, pp.536-538, 1847.
- [182] The Number of Hidden Layers', <https://www.heatonresearch.com/2017/06/01/hiddenlayers>
- [183] A. Ng, "Deep Learning". CS229 Lecture Notes, pp.1-30.
- [184] M.I. Razzak and S. Naz, "Microscopic Blood Smear Segmentation and Classification using Deep Contour Aware CNN and Extreme Machine Learning". Int. Conf. Computer Vision and Pattern Recognition Workshops, USA, pp. 49-55, 2017.
- [185] J.S. Lee, "Digital Image Enhancement and Noise Filtering by Use of Local Statistics". IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.2 (2), pp.165-168, 1980.
- [186] J.S. Lee, " Digital Image Smoothing and the Sigma Filter". Computer Vision, Graphics, and Image Processing, vol. 24, pp. 255-269, 1983.
- [187] A. Lopes, R. Touzi and E. Nezry, "Adaptive Speckle Filters and Scene Heterogeneity". IEEE Transactions on Geoscience and Remote Sensing, vol. 28 (6), pp.992-1000, 1990.
- [188] J.S. Lee, J.H. Wen, T.L. Ainsworth, K.S. Chen, and A.J. Chen, "Improved Sigma Filter for Speckle Filtering of SAR Imagery". IEEE Trans. Geoscience and Remote Sensing, vol.47(1), pp. 202-213, 2009.
- [189] H. Zhong, L. Lu and L. Jiao, "Fast Non-local Lee Filter for SAR Image Despeckling Using Directional Projection". Proceedings of CIE International Conference on Radar, China, pp. 1600-1603, 2011.
- [190] H. Zhong, L. Lu, J. Zhang, et al., "Nonlocal-Lee filter for SAR image despeckling based on hybrid patch similarity". International Geoscience and Remote Sensing Symposium, Australia, pp.1987-1990, 2013.
- [191] A. Buades, B. Col and J. Morel, "A non-local algorithm for image denoising". Computer Society Conference on Computer Vision and Pattern Recognition, U.S.A., pp.60-65. 2005.

- [192] A.S. Yommy, R. Liu and S. Wu, "SAR Image Despeckling using Refined Lee Filter". Int. Conf. on Intelligent Human-Machine Systems and Cybernetics, China, pp. 260-265, 2015.
- [193] H.A. David and H.N. Nagaraja, H.N.: Order Statistics. Wiley series in probability and statistics, 3rd Edition, 1970.
- [194] F.Qiu, J.Berglund, J.R. Jensen, *et.al.* "Speckle Noise Reduction in SAR Imagery Using a Local Adaptive Median Filter". GIScience & Remote Sensing, vol. 41(3), pp.244-266,2015.
- [195] E.M. Eliason and A.S. McEwen, "Adaptive Box Filters for Removal of Random Noise from Digital Images". Photogrammetric Engineering and Remote Sensing, vol. 56(4), pp. 453-458, 1990.
- [196] M. Jiang and D.Crookes, "High-performance 3D median filter architecture for medical image despeckling". IET Electronics Letters, vol.42(24), pp.1379 – 1380,2006.
- [197] C.Xiong, J. Hou, Z. Gao, *et.al.*, "Fast algorithm of 3D median filter for medical image despeckling". Proceedings of Medical Imaging, Parallel Processing of Images, and Optimization Techniques, pp. 678915-1:678915-5, 2007.
- [198] E.R. Davies, "Machine vision Theory, Algorithms". Practicalities. 3rd edition, Elsevier,2006.
- [199] R. Vanithamani, G. Umamaheswari and M. Ezhilarasi, "Modified Hybrid Median Filter for Effective Speckle Reduction in Ultrasound Images". Int. Conf. on Networking, VLSI and signal processing, United States, pp. 166-171,2010.
- [200] G.Umaheswari, R. Vanithamani, "An adaptive window hybrid median filter for despeckling of medical ultrasound images." Journal of Scientific & Industrial Research, vol.73(2), pp.100-102, 2014.
- [201] V. S. Frost, J. A. Stiles, K. S. Shanmugan and J. C. Holtzman, "A Model for Radar Images and Its Application to Adaptive Digital Filtering of Multiplicative Noise". IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. PAMI-4, pp. 157-166, 1982.
- [202] A. Lopes, R. Touzi and E. Nezry, "Adaptive Speckle Filters and Scene Heterogeneity". IEEE Trans. on Geoscience and Remote Sensing, vol. 28, pp. 992-1000, 1990.
- [203] G. Chen, X. Liu and Z. Zhou, "Modified frost speckle filter based on anisotropic diffusion". IET Int. Conf. on Radar Systems, U.K., pp. 1-4, 2007.
- [204] Y. Yu and S.T. Acton, "Speckle reducing anisotropic diffusion." IEEE Trans. on Image Processing, vol. 11, pp.1260-1270, 2002.
- [205] R. Touzi, A. Lopes, and P. Bousquet. "A statistical and geometrical edge detector for SAR imagery". IEEE Trans. on Geoscience and Remote Sensing, vol. 26, pp. 764-773, 1988.
- [206] H. Jia and T. Wang, "Modified Frost Filtering Based on Adaptive Parameter," Computer Engineering and Design, vol. 32, pp. 3793-3795, 2011.
- [207] Z. Sun, Z. Zhang, Y. Chen, S. Liu and Y. Song, "Frost Filtering Algorithm of SAR Images With Adaptive Windowing and Adaptive Tuning Factor". IEEE Geoscience and Remote Sensing Letters, pp. 1-5, 2019.
- [208] Y. LeCun, L.Bottou, Y.Bengio, *et.al.*, "Gradient-Based Learning Applied to Document Recognition." Proc. Of IEEE, pp. 1-46,1998.
- [209] Y. Zhou, H. Wang, F. Xu, Y-Q, Jin, "Polarimetric SAR Image Classification Using Deep Convolutional Neural Networks". IEEE Geosci. Remote Sens. Lett., vol. 13, pp.1935–1939, 2016.
- [210] F.Gao, T. Huang, J.Wang, J. Sun, A. Hussain and E. Yang, "Dual-Branch Deep Convolution Neural Network for Polarimetric SAR Image Classification". Appl. Sci., vol.7, 2017.
- [211] Z. Zhang, H. Wang, F. Xu, Y-Q, Jin, "Complex-Valued Convolutional Neural Network and Its Application in Polarimetric SAR Image Classification". IEEE Trans. Geosci. Remote Sens. vol. 55, pp.7177–7188,2017.
- [212] X.Mei, W.Nei, J.Liu and K.Huang, "PolSAR Image Crop Classification Based on Deep Residual Learning Network". Int. Conf. On Agro-Geoinformatics, China, pp.1-6,2018.
- [213] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition". arXiv:1512.03385v1, 2015.
- [214] <https://step.esa.int/main/download/polsarpro-v6-0-biomass-edition-toolbox-download/>
- [215] L.Wang, X.Xu, H.Dong, R.Gui and F.Pui, "Multi-Pixel Simultaneous Classification of PolSAR Image Using Convolutional Neural Networks". Sensors, vol. 18(3),pp.1-18 2018.
- [216] J. Long, E. Shelhamer and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation". arXiv:1411.4038v2, 2015.
- [217] R. Sudhakar, Edge preservation Index newfile
(<https://www.mathworks.com/matlabcentral/fileexchange/65261-edge-preservation-index-new-file>), MATLAB Central File Exchange.Retrieved March 2, 2021.

- [218] <https://search.asf.alaska.edu/#/?zoom=3¢er=-97.493959,39.672786>
- [219] J.Howard and S.Ruder, “Universal Language Model Fine-tuning for Text Classification”. arXiv:1801.06146v5, 2018.
- [220] D.Xiang, “Urban Area Information Extraction From Polarimetric SAR Data”.Doctoral Thesis, KTH architecture and the Built Environment, 2016.
- [221]<https://towardsdatascience.com/performance-metrics-confusion-matrix-precision-recall-and-f1-score-a8fe076a2262>
- [222] <https://towardsdatascience.com/activation-functions-neural-networks>
- [223] <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks>