SOME STUDIES ON IMPROVED IMAGE DENOISING TECHNIQUE USING SLIME MOULD ALGORITHM

By
ARPANA DEBNATH
Registration No-150135 of 2019-2020
Exam Roll No-M6VLS22007

Under the Guidance of:
Prof. P. VENKATESWARAN

Thesis submitted in partial fulfillment of the requirements for the award of the Degree of Master of Technology in VLSI Design and Microelectronics Technology

Department of Electronics and Telecommunication Engineering
Jadavpur University
Kolkata-700032

August 2022

DECLARATION OF ORIGINALITY AND COMPLIANCE OF ACADEMIC ETHICS

I hereby declare that this thesis contains a literature survey and original work by the undersigned candidate, as part of her Master of Technology in VLSI Design and Microelectronics Technology.

All information in this document have been obtained and presented in accordance with academic rules and ethical conduct.

I also declare that as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name of Candidate: ARPANA DEBNATH

Exam Roll No: M6VLS22007

Thesis Title: Some Studies on Improved Image Denoising Technique using Slime Mould Algorithm.

Signature of the Candidate:

____

Date:

FACULTY OF ENGINEERING AND TECHNOLOGY
JADAVPUR UNIVERSITY

CERTIFICATE OF RECOMMENDATION

I hereby recommended that this Thesis prepared by ARPANA DEBNATH entitled Some Studies on Improved Image Denoising Technique using Slime Mould Algorithm under my supervision be accepted for partial fulfillment of the requirements for the award of the degree of Master of Technology in VLSI Design and Microelectronics Technology.

Dr. P. Venkateswaran
Professor
Dept. of Electronics and Tele-Communication Engineering
Jadavpur University, Kolkata-700032

Dr. Manotosh Biswas
Prof. & Head
Dept. of Electronics and Tele-Communication Engineering
Jadavpur University, Kolkata-700032

Prof. Chandan Mazumdar
Dean, Faculty of Engineering & Technology
Jadavpur University, Kolkata-700032

FACULTY OF ENGINEERING AND TECHNOLOGY

JADAVPUR UNIVERSITY

CERTIFICATE OF APPROVAL

The foregoing Thesis is hereby accepted as a credible study of an engineering subject carried out and presented in a manner that is satisfactory to warrant its acceptance as a prerequisite to the degree for which it has been submitted. It is understood that by the approval, the undersigned don't necessarily endorse or approve any statement made, opinion expressed, or conclusion drawn therein but approve the Thesis only for the purpose for which it is submitted.

Committee on Final Examination for

Evaluation of the Thesis of: ARPANA DEBNATH

Exam Roll no: M6VLS22007

____

Signature of Examiners

ACKNOWLEDGEMENT

I express my deep sense of gratitude and indebtedness to my Thesis Supervisor, Prof. P. Venkateswaran, Department of Electronics & Tele-communication Engineering (ETCE), Jadavpur University (JU), Kolkata, for providing me the opportunity to carry out the Thesis work. I am grateful to him for the valuable insights and suggestions that he gave me throughout my M. Tech Thesis work.

Further, I express my special thanks to Dr. Supriya Dhabal, Assistant Professor, Dept. of Electronics & Communication Engineering, Netaji Subhas Engineering College, Garia, Kolkata, for his constant and effortful guidance by providing necessary information whenever required.

I would like to thank, Prof. Subir kumar Sarkar, Dept. of ETCE, Jadavpur University for his compassionate approach. I would like to express my sincere gratitude to all the teaching and non-teaching staffs of the department for providing the necessary support.

I would also like to thank Prof. Manotosh Biswas, the Head, Dept. of ETCE, Jadavpur University for his compassionate approach during this Thesis work. I also convey my heartiest appreciation to Prof. Ananda Shankar Chowdhury, Former Head, Dept. of ETCE, JU, for his constant support and encouragement.

Last but not the least, this work would not have been possible without the support and encouragement of my family, classmates, and near and dear ones.

ARPANA DEBNATH

Computer Networking Lab (T-3-16A)
Dept. of ETCE, Jadavpur University
Kolkata-700032. West Bengal.

Date:

ABSTRACT

This Thesis presents an improved denoising method based on the order of cascaded filters. The amalgamations of different filters are obtained through the performance of Slime Mould Algorithm (SMA) and cascading four filters out of twelve filters. The proposed algorithm has several new features which provide enhanced denoised images. SMA has a new and individual mathematical model that uses adaptive measures to stimulate the affair of fabricating positive and negative feedback of the propagation wave of slime mould, and the entire model is based on a bio-oscillator to form the optical path for collecting food with excellent exploitation propensity and exploratory ability. Most of the existing denoising techniques are suitable for either Salt & Pepper, Gaussian, or Speckle noise but our proposed algorithm is helpful in removing all three noises. The denoised images are far better in terms of both visual and quantitative analysis. The proposed algorithm also provides a significant improvement compared to other algorithms.

CONTENTS

		Page no
Cover page………………………………………………………. …………………		i
Declaration of Originality and Compliance of Academic Ethics……………………...		ii
Certificate of Recommendation……………………………………………………….		iii
Certificate of Approval………………………………………………………………..		iv
Acknowledgment……………………………………………………………………...		v
Abstract…………………………………..………………………………………		vi
Contents……………………………………………………………………………….		vii
List of Figures…………………………………………………………………………		ix
List of Tables………………………………………………………………………….		xii
CHAPTER 1	INTRODUCTION	1-5
1.1	Preamble	2
1.2	Literature Survey	2
1.3	Thesis Motivation	4
1.4	Thesis Outline	5
CHAPTER 2	IMAGE DE-NOISING USING SLIME MOULD ALGORITHM	6-16
2.1	Introduction	7
2.2	Image De-Noising Using Slime Mold Algorithm	7
	2.2.1 Originality	7
	2.2.2 Mathematical Model	9
	2.2.3 Approach Food	9
	2.2.4 Wrap Food	10
	2.2.5 Computation Complexity Analysis	11
2.3	Brief Idea about Different Types of Denoising Filters	12
2.4	Quality Metrices	14
2.5	Proposed Cascaded Filter Design	15
2.6	Summary	16

CHAPTER 3	EXPERIMENTAL RESULT OF IMAGE DENOISING TECHNIQUE	17-44
3.1	Introduction	18
3.2	Experiment on Salt & Pepper Noise Reduction	18
	3.2.1. General Images	
	3.2.2. Medical Images	
3.3	Experiment on Gaussian Noise Reduction	29
	3.3.1. General Images	
	3.3.2. Medical Images	
3.4	Experiment on Speckle Noise Reduction	34
	3.4.1. General Images	
	3.4.2. Medical Images	
3.5	Summary	44
CHAPTER 4	CONCLUSION AND FUTURE SCOPE	45-47
4.1	Conclusion	46
4.2	Future scope	46
REFERENCES		48-54

List of Figures

Figure no		Page no
2.1	Foraging morphology of slime mould	8
2.2	Assessment of fitness	10
3.1	Resulting images after Denoising, for salt & pepper noise for Lena image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	20
3.2	Resulting images after Denoising, for salt & pepper noise for cameraman image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	20
3.3	Resulting images after Denoising, for salt & pepper noise for Einstein image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	21
3.4	Resulting images after Denoising, for gaussian noise for Lena image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	23
3.5	Resulting images after Denoising, for gaussian noise for cameraman image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	23
3.6	Resulting images after Denoising, for gaussian noise for Einstein image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	24
3.7	Resulting images after Denoising, for speckle noise for Lena image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	25
3.8	Resulting images after Denoising, for speckle noise for cameraman image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	26
3.9	Resulting images after Denoising, for speckle noise for Einstein image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	26

3.10	Resulting images after Denoising, for gaussian noise for DS1 image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	28
3.11	Resulting images after Denoising, for gaussian noise for DS2 image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	28
3.12	Resulting images after Denoising, for gaussian noise for DS3 image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	29
3.13	Resulting images after Denoising, for speckle noise for DS1 image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	30
3.14	Resulting images after Denoising, for speckle noise for DS2 image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	31
3.15	Resulting images after Denoising, for speckle noise for DS3 image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	31
3.16	Resulting images after Denoising, for salt & pepper noise for DS1 image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	33
3.17	Resulting images after Denoising, for salt & pepper noise for DS2 image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	33
3.18	Resulting images after Denoising, for salt & pepper noise for DS3 image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	34
3.19	Resulting images after Denoising, for salt & pepper noise for Einstein image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	35
3.20	Resulting images after Denoising, for salt & pepper noise for Lena image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	36

3.21	Resulting images after Denoising, for salt & pepper noise for cameraman image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	36
3.22	Resulting images after Denoising, for salt & pepper noise for Einstein image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	38
3.23	Resulting images after Denoising, for salt & pepper noise for Lena image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	38
3.24	Resulting images after Denoising, for salt & pepper noise for cameraman image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	39
3.25	Resulting images after Denoising, for Speckle noise for DS3 image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	40
3.26	Resulting images after Denoising, for Speckle noise for DS2 image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	41
3.27	Resulting images after Denoising, for Speckle noise for DS1 image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	41
3.28	Resulting images after Denoising, for salt & pepper noise for DS3 image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	43
3.29	Resulting images after Denoising, for salt & pepper noise for DS2 image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	43
3.30	Resulting images after Denoising, for salt & pepper noise for DS1 image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed	44

List of Tables

Table no	List of Tables	Page no
3.1	PSNR, IQI, SSIM values after removal of Salt & pepper noise with variance 0.1 for Lena, Cameraman and Einstein image.	19
3.2	PSNR, IQI, SSIM values after removal of Gaussian noise with variance 0.1 for Lena, Cameraman and Einstein image	22
3.3	PSNR, IQI, SSIM values after removal of Speckle noise with variance 0.1 for Lena, Cameraman and Einstein image.	25
3.4	PSNR, IQI, SSIM values after removal of gaussian noise with variance 0.1 for DS1, DS2, DS3 image.	27
3.5	PSNR, IQI, SSIM values after removal of Speckle noise with variance 0.1 for DS1, DS2, DS3 image	30
3.6	PSNR, IQI, SSIM values after removal of Salt & pepper noise with variance 0.1 for DS1, DS2, DS3 image	32
3.7	PSNR, IQI, SSIM values after removal of Speckle noise with variance 0.5 for Lena, Cameraman and Einstein image.	35
3.8	PSNR, IQI, SSIM values after removal of Salt & pepper noise with variance 0.5 for Lena, Cameraman and Einstein image.	37
3.9	PSNR, IQI, SSIM values after removal of Speckle noise with variance 0.5 for DS1, DS2, DS3 image.	40
3.10	PSNR, IQI, SSIM values after removal of Salt & pepper noise with variance 0.5 for DS1, DS2, DS3 image.	42

CHAPTER 1

INTRODUCTION

1.1. Preamble

Visual data transmitted in the form of the digital image now has become an important part of communication in modern ages and digital images are often corrupted by different types of noises. So, the removal of noise from digital images has become an important task in the field of image processing. It is a fundamental challenge of image processing and the ultimate goal is to evaluate the original image by repressing the noise from a noise-affected image. Image may be corrupted by distinct intrinsic (example: sensor) and extrinsic sources of noise (environment) which are not feasible to avoid in a practical situation. So, image denoising plays a vital role in different applications like image restoration, image registration, visual tracking, image segmentation, and image classification. However, the frequency components like noise, edge, and texture are difficult to differentiate in the process of image denoising. A denoised image could unpreventably lose some details. So, recovering significant information from noisy images is highly important to get high-quality images. For Strong performance, it is very necessary to obtain the original image content. There are so many algorithms that have been proposed for the purpose of image denoising. Still, it is an open challenge to suppress the image noise, especially in situations where the images are acquired by the high noise levels.

1.2. Literature Survey

For the last four decades researchers have proposed several denoising algorithms to remove noise from different practical images like photographic, magnetic resonance imaging, computer tomography, satellite, astronomy, etc. [1-3]. During the process of image enhancement, transmission, capturing, as well as in image restoration noise can appear in digital images. Hence filtering of noise has become an important step in most of the work of image processing [4]. There are two basic methods of the image denoising process, the first one is the spatial filtering method and the second one is the transformed domain filtering method. The difference between spatial domain filtering and transform domain filtering is, spatial domain denoising is carried out by filtering the image directly, although in the transform domain filtering the image needs to be changed into frequency or transform domain. A few well-known examples of spatial domain filters are linear filter, Gaussian, Average, Median, Min, Max, Average, Adaptive filter, etc. [5-7]. Similarly, some examples of transforming techniques popular transform domain methods are Wavelet [8], Curvelet [10], Contourlet [9], Stein's Unbiased Risk Estimate Linear Expansion of thresholds [11], Shearlet [12] and so on.

Several denoising techniques were offered to reduce both low-density and high-density noises in image processing. For example, Zeng et al. proposes a method to confine the noise point in an image with the help of rough sets. The algorithm which is proposed in [5] clear aways impulse noise by the help of threshold and filtering templates of Median filter. Lamhiri proposed Wiener filtering method which is based upon a constant multistep image denoising approach and this is the approach where denoised image from previous iteration works as input to the next iteration. The authors in [13] have used different applied Median filter (DAMF) to reduce salt & pepper noise by contemplating the value of adjacent pixels and adaptive window. This method obtains better estimation of the real pixel value in terms of structural similarity index (SSIM) and Peak Signal to Noise Ratio (PSNR). The constant process holds on until the energy of current image is smaller than the previously obtained denoised image. The authors in [14] offered Fuzzy Firefly Bayes filter to denoise grey level image which uses Bayes algorithm and probabilistic approach to find out noisy pixels. The conventional filtering techniques like Mean, Gaussian, and Wiener filtering are responsible for blurring the sharp edges, destroying image lines and finer details. To solve this problem Several non-linear and edge -preserving filters such as bilateral [15], weighted bilateral [16], median [13] and guided image filters [17] have been designed to remove different types of noises. The output of the bilateral filter is presented as a weighted average of local-neighborhood pixels.

The literature survey helps to determine the effectiveness of the algorithms of image denoising which includes Particle Swarm Optimization (PSO) [18], Artificial Bee Colony (ABC) [19], Cuckoo Search Algorithm (CSA) [20], Genetic algorithm [21], Differential evolution (DE) [3], Differential evolution-based PSO [22], and so on. The recent developments in the field of image denoising indicates the superiority of today's optimization algorithm compare to conventional methods. But still, very few steps were made by the researchers for designing the mechanism for denoise the corrupted image using these heuristic and meta-heuristic algorithms.

Recently Metaheuristic Algorithms (MAs) have become more powerful in several applied fields as they have high performance and low required computing capacity and time in various optimization problems. These MAs are mostly energized by real-world phenomena to find out the better heuristic solutions for optimization problems by simulating biological phenomena or physical rules. MAs are categorized in two types: swam-based methods and evolutionary techniques and the first category mainly simulates physical phenomena and applies mathematical rules which includes Multiverse optimizer (MVO) [22], Gravitational search algorithm (GSA) [23], Teaching-Learning-Based Optimization Search (TLBO) [24], Sine cosine algorithm (SCA) [25], and Central Force Optimization (CFO) [26].

In spite of the huge success of recent MAs none of them can assure finding a global optimum for all optimization problems and this logic is described beautifully in the "No-Free-Lunch" Theory [27]. This Theorem motivates numerous researchers to develop new types of algorithms and solve different classes of problems efficiently. With the aspiration, a new meta-heuristic algorithm was introduced which is known as Slime mold Algorithm. Many researchers used this algorithm for different types of problems [28-36]. The Slime mold algorithm performs better than the other compared algorithms. So, this chapter has proposed a novel algorithm, named as SMA to obtain faster convergence in complex optimization problems.

1.3. Thesis Motivation

Slime mold algorithm (SMA) has several new features with an identical mathematical model for causing positive and negative feedback of the propagation wave of SMA based on a bio-oscillator to form an excellent path for connecting food with optimal exploration and exploitation. So, we have used this algorithm for image denoising purposes for its excellent exploration and exploitation property. Image denoising using slime mould algorithm has never been designed before, this is the first time when we have used this algorithm for designing filters for image denoising purposes. The motivation behind this work is it gives better results than popular filters like Frost, Median, Lee, Kuan, Wiener, etc. This algorithm is suitable for all types of noise like gaussian, salt & pepper, speckle, etc. It is highly suitable for salt & pepper noise, the output of this algorithm for salt & pepper noise has a noticeable difference than well-known filters like median, Frost, Kuan, Lee, etc. This provides us the second level of motivation to use this algorithm for getting an improved denoising technique using SMA. In this chapter, the newly proposed algorithm has been discussed with supporting data, tables, and figures so that differences in the output can be easily noticed. Different types of image files have been taken. Different types of filter combinations and different types of results are obtained for different types of variances. PSNR, IQI, and SSIM these three parameters are used here for comparing all the filtering outputs with obtained results. The corresponding results are shown in tables.

1.4. Thesis Outline

The thesis has been systematically arranged and it is covered by the theoretical aspects in detail so that one can understand the practical work which was done using MATLAB. This thesis is about designing the filter using slime mold algorithm for image denoising purposes. The whole work is designed in 4 chapters.

Chapter 1: It presents the introduction which basically talks about the necessity of designing filters using slime mold algorithm in the field of image de-noising. Chapter 1 is also divided into 4 parts, they are introduction, literature survey, thesis motivation, and thesis outline. A literature survey provides information about the different research papers provided by different authors. Thesis motivation provides the aim behind designing the filter using SMA algorithm.

Chapter 2: It presents the overview of the slime mould algorithm and the mathematical model of SMA. Here different types of filters and noise measuring parameters like PSNR, IQI, and SSIM, etc. are also discussed.

Chapter 3: Chapter 3 provides the discussion about the performed experiment. In this chapter, the algorithm is discussed with the help of tables and figures and the best result is highlighted in red color. Ten different experiments are conducted with different degradation values to show the effectiveness of mentioned algorithm.

Chapter 4: Chapter 4 presents the conclusion of all experiments and it also provides the idea about the future work of proposed algorithm.

CHAPTER 2

IMAGE DE-NOISING USING SLIME MOULD ALGORITHM

2.1 Introduction

Visual images transmitted in the form of digital images have an important role in the field of modern technology, research area, and field of medical science such as satellite communication, X-ray imaging etc. But the images obtained after transmission are often corrupted by noise. Therefore, the removal of noise is utmost important in the field of image de-noising.

2.2 Image De-noising using Slime Mold Algorithm

In this section, the basic concept and behavior of slime mold algorithm will be described and then a mathematical model will be established depending upon its behavioral pattern.

2.2.1 Originality

Slime mold is an informal name given to several kinds of eukaryotic organisms which are unrelated and which can live freely as a single cell. They can aggregate [37] together to form multicellular reproductive structures. More than 900 species of slime mold occur globally. When food is in aplenty amount these slime molds exist as a single-cell organism and when food is in scarcity many of this single-cell organism will unite together and start moving as a single body. In this situation, they are more deplicate to airborne chemicals and this helps to find out food sources. They can easily change their shape, size, and function of parts. They may stalk that breed fruiting bodies, releasing countless spores and they are too light to be carried on the wind.

The SMA proposes a paper which mainly simulates the behavioral and morphological changes of slime mold physarum polycephalum [39] in foraging. It does not model its complete life cycle. Inspired by the diffusion of Slime mold Schmickland and Crailshein proposed a bio-inspired navigation principle for Swarm Robotics [37]. This algorithm also inspires researchers for graph optimization [38].

Kamiya and his colleagues [33] was the first team to research the detailed process of cytoplasmic flow of Slime mould. Their work was fabulous for our understanding of movement of slime mould and how they connect to food. When a vein proceeds towards a food source, the bio-oscillator fabricates a propagating wave which increases the cytoplasmic flow through the vein and more the thicker the vein is, the faster the cytoplasmic flow. Through the positive-negative response the slime can set up the minimal path to link food in a better way. Slime-mould was also mathematically modelled and applied in graph theory and path networks.

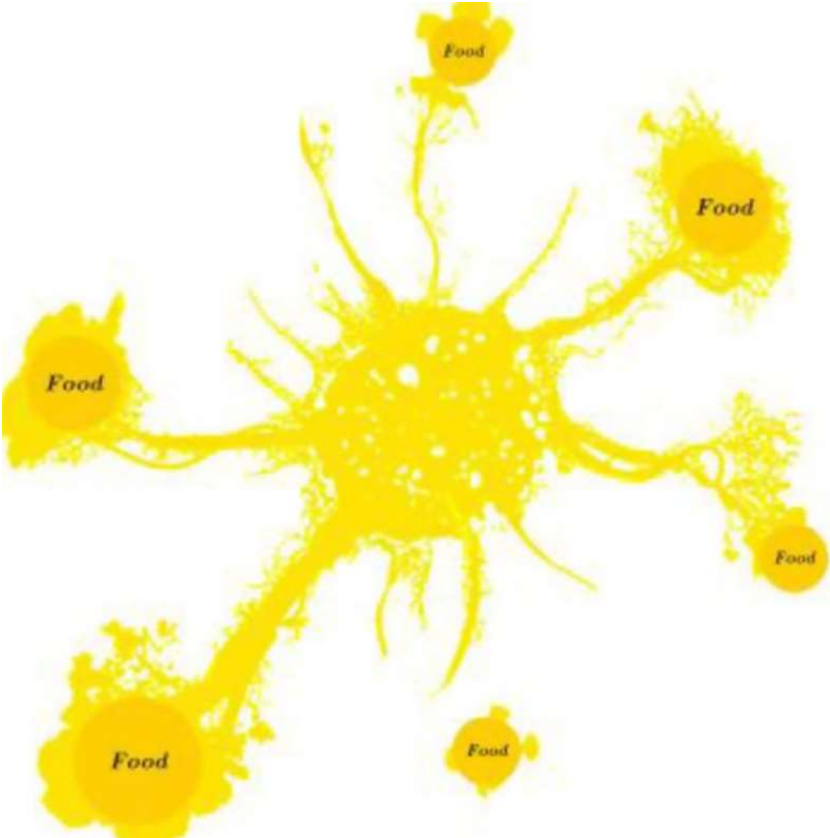


Figure 2.1: Foraging morphology of slime mould

The venous structure of slime mould creates along with the phase difference of the contraction mode. The Foraging morphology of Slime Mould is shown in Figure 2.1. There are three correlations in between the morphological changes of the venous structure and the contraction mode of slime mould as follows [32]:

a) Thick veins configure roughly along with the radius when the contraction frequencies range from outside to inside.
b) When the contraction mode is not stable, anisotropy come into sight.
c) When the contraction design of slime mold is no longer arrayed with time and space, the venous structure dis-appear.

Therefore, the correlation between venous structure and contraction design of slime mould is rigid with the shape of naturally formed cells. The width of the vein structure is dogged by the flow of cytoplasm in the physarum solver. Slime mould can establish a stable path where food concentration is more. So, they can get the utmost concentration of nutrients. In the recent studies it is seen that Slime mould have the competence of making foraging arrangements based on optimization theory [38-49]. When the quality of different types of foods is different from each other slime mould chooses the food with maximum concentration. Slime Mould can also hunt route according to the status of foodstuff provenience. When the amount of food source is maximum the slime mould will use the place -limited search algorithm, so focusing the search on the food sources that have been searched. If the denseness of food concentration is minimum the slime mould will depart from that place to find another food sources in the area.

2.2.2 Mathematical Model

In this section mathematical model and proposed method will be described in details.

2.2.3 Approach Food

Slime mould can search about their food in the air. To express its searching character in mathematical formulae, the following formulae are proposed.

$$\overrightarrow{X(t+1)} = \begin{cases} \overrightarrow{X_b(t)} + \overrightarrow{V_b} * \left(\overrightarrow{W} * \overrightarrow{X_A(t)} - \overrightarrow{X_B(t)}\right), r < p \\ \overrightarrow{v_c} * \overrightarrow{X(t)}\ , r \geq p \end{cases} \tag{2.1}$$

where $\overrightarrow{v_b}$is a parameter with range of (-a, +a), $\overrightarrow{v_c}$decreases linearly from 1 to 0, t is current iteration, $\overrightarrow{X_b}$ is indivisual location with the highest order concentration, X is location of Slime Mould, $\overrightarrow{X_A}$and $\overrightarrow{X_B}$are two individuals randomly pick out from Slime Mould.

$$P = \tanh|\ S(i)\text{-}DF| \tag{2.2}$$

where i belongs to 1,2,3,…….n, S(i) represents the fitness of x, DF represents the best fitness obtained in all iterations.

The formula $\overrightarrow{v_b}$ is as follows

$$\overrightarrow{v_b} = [-a, a] \tag{2.3}$$

$$a = \operatorname{arctanh}\left(-\left(\frac{t}{\max\_t}\right) + 1\right) \tag{2.4}$$

The formula of $\overrightarrow{W}$ is given below

$$\overrightarrow{W(Smellındex(ı))} = \begin{cases} 1 + r.\log\left(\frac{bF - S(i)}{bF - \omega F} + 1\right), condition \\ 1 - r.\log\left(\frac{bF - S(i)}{bF - \omega F} + 1\right), for\ others \end{cases} \tag{2.5}$$

Smellindex=sort(S) (2.6)

2.2.4 Wrap Food

This part mathematically simulates the venous tissue formation of slime mold during searching. The higher is the congregation of food approached by the vein, the stronger the propagating wave created by the bio-oscillator. The assessment of fitness is shown in Figure 2.2. Best fitness means the availability of food approached by the vein in this location is higher compared to other location.

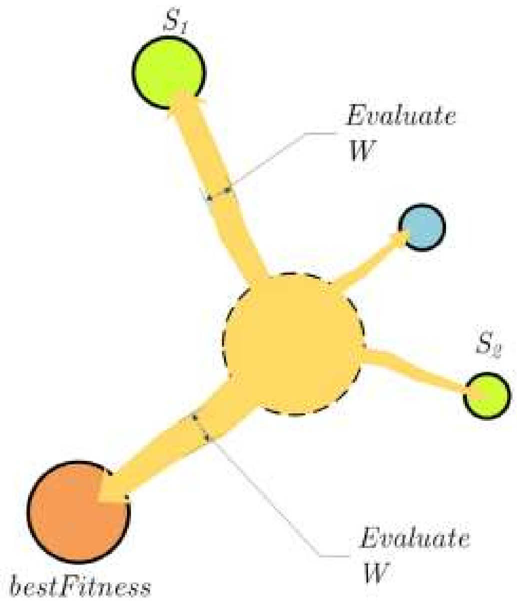


Figure 2.2: Assessment of fitness

$$\overrightarrow{X^*}=\begin{Bmatrix} rand.(UB-LB)+LB, rand < z \\ \overrightarrow{X_b(t)}+\overrightarrow{vb}\cdot\left(W\cdot\overrightarrow{X_A(t)}-\overrightarrow{X_B(t)}\right), r < p \\ \overrightarrow{vc}\cdot\overrightarrow{X(t)}, r \geq p \end{Bmatrix} \quad (2.7)$$

The pseudo-code of the SMA is shown in algorithm 1. There are still many mechanisms that can be added to the algorithm and also more comprehensive simulation can be added to the life cycle of slime mold. To enhance the ability of algorithm we simplify the process and operators of the algorithm.

Algorithm1: Pseudo-code of SMA algorithm

Do the initialization of the parameters popsize, Max_iteration;

Do the initialization of the positions of slime mold X_i (i =1, 2, …….., n);

While (t<=Max_iteration)

a) Find out the fitness of complete slime mold;
b) Update bestfitness, X_b
c) Find out the W by equation 4

For each search portion

a) Update p, vb, vc;

b) Update positions by equation 6

End for

t=t+1;

End While

Return bestfitness, X_b;

2.2.5 Computation Complexity Analysis

SAM generally has the subsequent components like initialization, sorting, fitness evaluation, location update and weight update. Among all of the components, N denotes number of cells of slime mold, D denotes dimension of functions, and T denotes maximum number of iterations. O(D) is computation complexity of initialization, O(N+NlogN) is computation complexity of fitness evaluation and sorting, O(N*D) is computational complexity of weight update, O(N*D) is the complexity of location update. So, O(D+T*N*(1+log N+D)) is the total complexity of SMA.

2.3 Brief Idea about Different Types of Denoising Filters

There are various types of filters which are used for image denoising purpose. Some of the filters are excellent in apparent interpretation where as few of them are used for smoothing purposes, noise reduction etc. Some of the popular examples of traditional filters are mean, median, wiener, lee, kuan, frost etc. Besides, several researchers have proposed algorithms [51-65] which provides the basic idea about image denoising [66-99] and filters which are used to denoise the degraded images.

Mean Filter: Mean filter [6] is also known as Average filter. Mean filter is a type of scaler filter. They are specially used for removing speckle noise. Mean filter was invented in 1984 by Pomalaza-Raez. It is a very simple and instinctive filter. This filter does not remove speckle noises at whole but removes at some extend. It works on the basis of average value of pixel where the center pixel is replaced by the average value of all the pixels. As a result, this type of filters provides blurring effect to the images. So, this filtering method is least useful in removing Speckle noise as it results in loss of details.

Mathematical expression is given by the equation as below

$$h(i,j)=\sum_{k\in m}\sum_{l\in n} f(k,l) \quad (2.8)$$

Median Filter: Median filter [5] basically works on the basis of center of the pixel means. It works by observing the pixels by pixel. This filter replaces each pixel value with its median value of neighboring pixel. This pattern of neighbor pixel is called as window that slides pixel by pixel through the entire image. The process of calculating median value is at the first we have to sort all the pixel value in numerical order from the window and secondly, we have to replace all the pixel by considering its middle pixel value. Median filtering technique is a nonlinear filtering technique. This median method is especially used to remove salt & pepper noise. This method mainly focuses on to preserve the end pixels. It is also effective in strong spike components. The main disadvantages of median methods are it consumes lot of time to execute, it also needs extra computational time for sorting the intensity value of each set.

Median filter follows the following algorithm:

a) It takes 3×3 or 5×5 region centered around the pixel (i,j).
b) The filter sorts the intensity of the pixel values in the described region in ascending order.
c) It considers the middle value as the new value of pixel (i,j).

Wiener Filter: Wiener filter [6] is also known as least mean square filter. The idea of wiener filter was put forward in the year of 1942 and this is compatibly used on the field of image denoising depending on the variance. It is also used to minimize the final mean square error in the process of noise smoothy and inverse filtering**.** It is the considered by the linear calculation of original image. The wiener filter mainly depends on the variance: if the variance is low the filter works smoothly; if the variance is high the filter works less smoothly. Wiener filter provides better result than the linear approach.

The expression of wiener filter is given below

$$F(u,v) = \left[\frac{H(u,v)*}{H(u,v)^2+\left[\frac{Sn(u,v)}{Sf(u,v)}\right]}\right]G(u,v) \tag{2.9}$$

where $H(u,v)^2$ is the decimation function and H(u,v)* is the conjugate complex, G(u,v) is the decimated image. Sn(u,v) and Sf(u,v) are power spectra of original image

Lee Filter: Lee filter works [6] on the basis of the presumption that the variance and the mean of pixel of interest is nearly equal to the variance and the local average of all pixels with in the moving kernel. Lee filter was invented in 1981 by Jong Sen Lee. It works depending on the multiplicative speckle model and uses local statistics to preserve the details. Lee filter also performs on the basis of variances: if the area of the variance is low it performs smoothly and if the area of the variance is high then it does not work smoothly. So, it means it can store the details in both low and high contrast and hence it is adaptive in nature.

Mathematical expression of Lee filter is given as follow:

$$I_{mg(i,j)} = i_m + W*(Cp - I_m) \tag{2.10}$$

where i_m is mean intensity of the filter

Kuan Filter: Kuan filter [26] is a type of local linear filter. This filter a kind of multiplicative in nature. This minimum square error filter is based on minimum multiplicative order and it does not make any approximation on the variance of noise within the filter window like Lee filter. It is multiplicative model of Speckle noise in the additive linear form. Its weighted function is calculated as follows:

$$W=\left(\frac{1-C_u}{C_i}\right)(1 + C_u) \tag{2.11}$$

C_u is calculated by the following expression

$C_u = \sqrt{\frac{1}{ENL}}$ (2.12)

And C_i is the variation coefficient of the image calculated as follows:

$C_i = \frac{S}{I_m}$ (2.13)

Frost Filter: Frost filter [28] helps to replace the pixel of interest with the Value's weighted sum within the n×n moving kernels. With the distance from the pixel of interest the weighting factor decreases. The weighted factor increases for the central pixel as variance with in the kernel increases. This spatial domain adaptive filter is based on multiplicative noise order. Frost filter has the following expression:

DN=$\varepsilon_{n\times n}$ $K\alpha e^{-\alpha}|t|$ (2.14)

where α= $(4/_{n\sigma^2})(\sigma^2/_{I^2})$ (2.15)

K=Normalization constant, α=image coefficient of variation value

σ=local variance, I=local mean

$|t|=|X - X_0|+|Y - Y_0|$ (2.16)

n=moving kernel size [5]

2.4 Quality Metrics

The following are the quality metrics that are calculated for the denoised images.

Peak Signal to Noise Ratio (PSNR): Peak signal-to-noise [82] is an expression which is defined as the ratio of maximum possible value (power) of a signal and the power of distorting noise that affects the quality of representation.

The PSNR in dB for a gray-scale image is given as

PSNR = $10\log_{10}(\frac{255^2}{MSE})$ (2.17)

where MSE denotes the Mean Square Error between the pixel values of the actual and noisy image. It is given by

MSE=$\frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}(f(i,j) - g(i,j))$ (2.18) where,

f (i, j) = original image,

g (i, j) =image obtained after filtering,

M * N = dimension of image

Image Quality Index (IQI): Image quality index [87] is defined as the measure for deriving the level of degradation present in an image. Without reference image measurement of such index is challenging.

$$IQI=\frac{\sigma_{ab}}{\sigma_a*\sigma_b}*\frac{2m_a*m_b}{m_a^2+m_b^2}*\frac{2\sigma_a\sigma_b}{\sigma_a^2+\sigma_b^2} \quad (2.19)$$

where σ_a is mean value of A image

and σ_b is mean value of B image;

σ_a^2, σ_b^2 and σ_{ab} are the variances of A and B and covariance of (A, B).

Structural Similarity Index (SSIM): The structural similarity index is a method for determining the perceived quality of digital television and cinematic pictures, as well as other kinds of digital image and videos. SSIM is used for determining the similarity between two images.

$$SSIM\ (x, y) = \frac{2\mu_a\mu_b+c_1}{\mu_a^2+\mu_b^2+c_1}*\frac{2\sigma_{ab}+c_2}{\sigma_a^2+\sigma_b^2+c_2} \quad (2.20)$$

where μ_a, μ_b denote the average value of A and B; σ_a^2, σ_b^2 are the variances of A and B; And σ_{ab} is covariance of A and B.

2.5. Proposed Cascaded Filter Design

The traditional filters like Average, Median, Wiener, Gaussian, Circular etc. have different types of drawbacks. By considering their drawbacks we have designed slime mold algorithm to eliminate commonly occurred noises such as gaussian, salt & pepper, speckle noise with different variance values. This cascaded filter is designed with any four combinations of the following filters:

1: Mean 3×3, 2: Mean 5×5, 3: Median 3×3, 4: Median 5×5, 5: Wiener 5×5, 6: Wiener 3×3,7: Wiener 7×7, 8: Gaussian, 9: Cone, 10: Pyramid, 11: Circular, 12: Unsharp Masking.

Slime mold algorithm is applied to denoise the three main standard images such as Lena, Cameraman, Einstein. The related filter outputs are simulated for 40 trials and to observe related quantitative analysis.

2.6 Summary

Chapter 2 discusses the details about slime mould algorithm, it's originality and its mathematical model. It also gives the detail idea about image de-noising, different types of popular filter such as Mean, Median, Lee, Frost etc. and different noise measuring parameters such PSNR, IQI, and SSIM etc. This chapter also discusses about the different filters used in proposed cascaded filter design.

CHAPTER 3

EXPERIMENTAL RESULTS OF PROPOSED IMAGE DE-NOISING TECHNIQUE

3.1 Introduction

Slime Mold Algorithm is used here to denoise the three main standard images such as Cameraman, Einstein, Lena and also three medical images DS4, DS5, DS6 etc. The related algorithms are simulated for 40 trials and to perform related quantitative analysis. To analyze the effectiveness of the proposed algorithm PSNR, IQI, and SSIM parameters are evaluated with the sample images corrupted with gaussian, salt & pepper and speckle noise with different value of variances. To show the computational results three different experiments were carried out. The complete simulation work is performed in MATLAB 2015a on a HP computer having windows 10, AMD Ryzen 3350U with Radeon Graphics 2.60 GHz, 4GB RAM, 64bit Operating System.

Total 10 experiments have been conducted to verify the effectiveness of proposed algorithm. Six different images are taken to do all the experiments. Three of the images are medical images and rest three are Lena, Cameraman, and Einstein. The first three experiments are done for variance 0.1 and Lena, Cameraman and Einstein these three images are took down. Next three images are medical images, these experiments are also performed for variance 0.1. The rest of the experiments are performed for variance 0.5 and these experiments are also done for Lena, Cameraman, Einstein and three medical images. All these experiments are performed for three standard noise and they are Gaussian, Salt & Pepper and Speckle noise. The proposed algorithm provides very good results for all the standard noises like Gaussian, Salt & Pepper and Speckle noise and this algorithm provides best output for Salt & Pepper noise. PSNR, IQI, and SSIM, these three parameters are taken for comparing with other filtering output. The details about each experiment are mentioned in Section 3.2, 3.3, and 3.4 respectively.

3.2 Experiments on Salt & Pepper Noise Reduction

In this section we have discussed the experimental results for the reduction of salt & pepper noise considering few general-purpose images and few medical images.

3.2.1 General Purpose Images (Variance 0.1)

In this experiment a set of test images like Lena, Cameraman, Einstein are corrupted with salt & pepper noise with variance 0.1 and then the degraded images are denoised with proposed algorithm and six other existing filters. To compare the results with all other filters PSNR, IQI, and SSIM these three parameters are used here. The best result is highlighted with red color and obtained sequence is also highlighted.

Simulated results given in Table 3.1 shows that the proposed algorithm gives better results for all three parameters, compared to six other filters which are used here. The obtained sequence 3-3-3-3 means Median 3×3 filter is used 4 times in sequence and produces the best result for all the three images which are corrupted with salt & pepper noise with variance 0.1. The resulted images are shown in Figure 3.1, Figure 3.2, and Figure 3.3 respectively.

Table 3.1: PSNR, IQI, and SSIM values after removal of Salt & Pepper noise with variance 0.1 for Lena, Cameraman and Einstein image.

Image	**Lena**			**Cameraman**			**Einstein**		
Method	**PSNR**	**IQI**	**SSIM**	**PSNR**	**IQI**	**SSIM**	**PSNR**	**IQI**	**SSIM**
Noisy	15.422	0.953	0.688	15.107	0.961	0.777	15.631	0.956	0.582
Median	19.669	0.953	0.856	25.050	0.949	0.849	30.901	0.969	0.853
Wiener	20.468	0.788	0.598	21.154	0.941	0.472	22.523	0.924	0.462
Lee	23.978	0.954	0.939	21.624	0.958	0.936	23.460	0.910	0.887
Kaun	24.540	0.968	0.944	20.897	0.956	0.921	23.810	0.938	0.887
Frost	25.753	0.972	0.958	21.200	0.957	0.928	24.590	0.937	0.907
Proposed	**32.271**	**0.993**	**0.992**	**25.289**	**0.977**	**0.974**	**28.988**	**0.974**	**0.969**
Sequence	**3-3-3-3**			**3-3-3-3**			**3-3-3-3**		

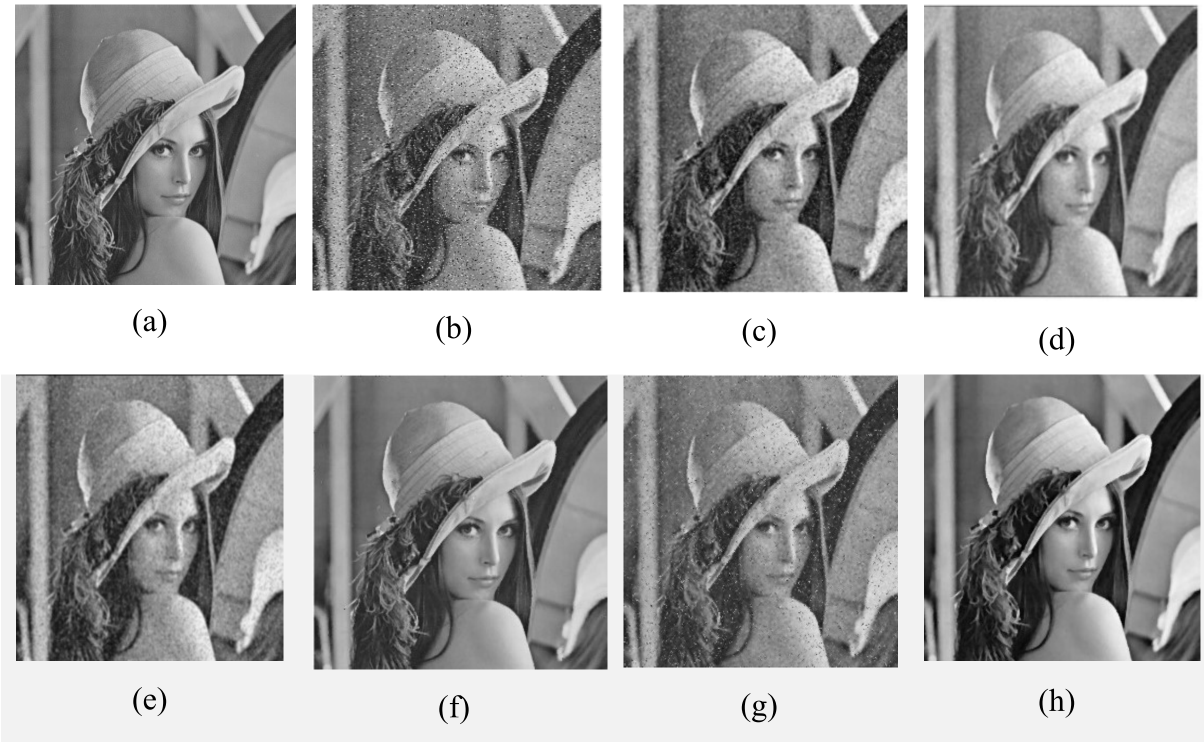


Figure 3.1: Resulting images after Denoising performance for salt & pepper noise for Lena image (a) original image, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

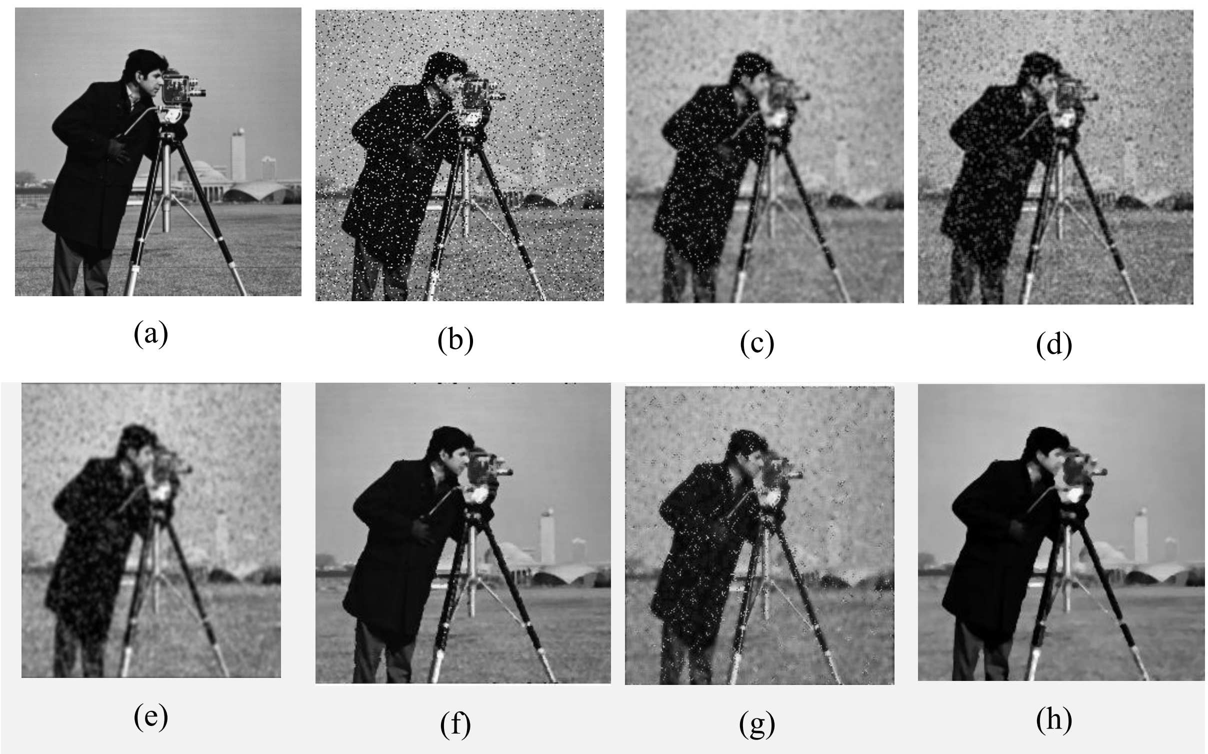


Figure 3.2: Resulting images after Denoising performance for salt & pepper noise for Cameraman image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

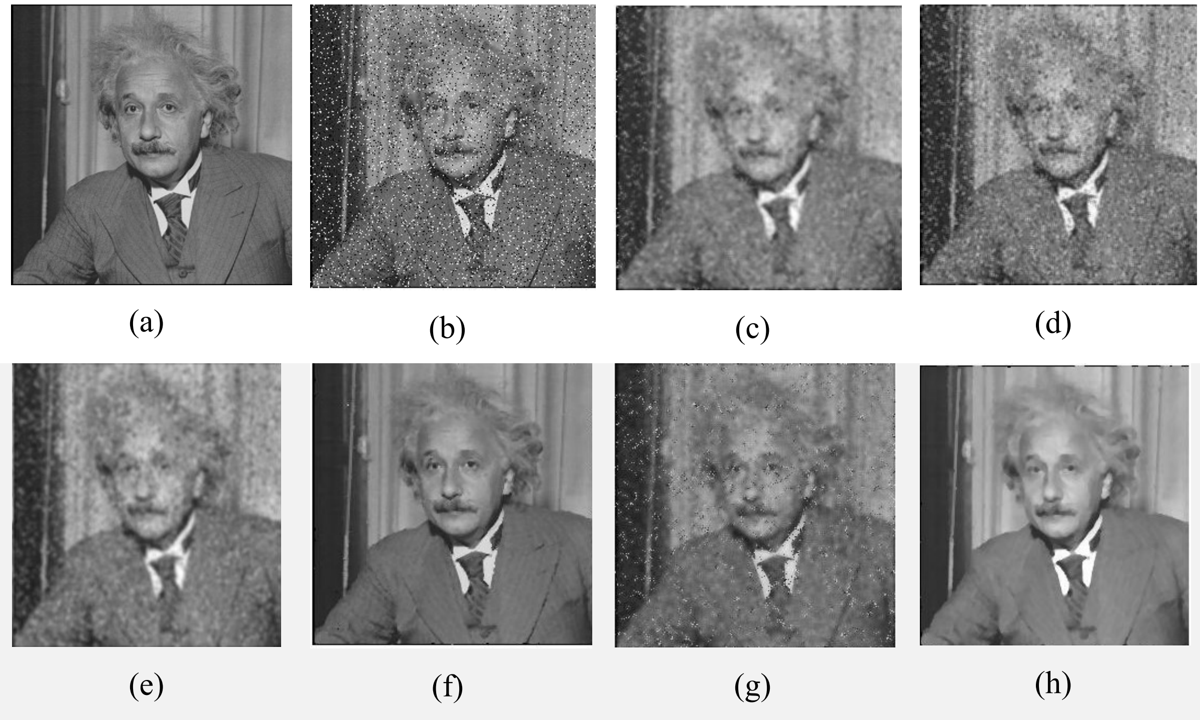


Figure 3.3: Resulting images after Denoising performance for salt & pepper noise for Einstein image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

3.2.2. Medical Images (Variance 0.1)

In this experiment a set of test images like DS1, DS2, and DS3 are corrupted with salt & pepper noise with variance 0.1 and then the degraded images are denoised with proposed algorithm and six other existing algorithms. To compare the results with all other algorithm PSNR, IQI, and SSIM these three parameters are used here. The best result is highlighted with red color and obtained sequence is also highlighted. Simulated results given in Table 3.2 shows that the proposed algorithm gives better results for all three parameters, compared to six other algorithms which are used here. The obtained sequence 3-3-3-3 means Median 3×3 filter is used 4 times in sequence and produces the best result for all the three images which are corrupted with salt & pepper noise with variance 0.1.

DS1, DS2, DS3 are medical images shown in Figure 3.4, Figure 3.5, and Figure 3.6. Nowadays image denoising plays a significant role in the medical field. They are specially used for detecting tumor, cancer cell, cyst etc. So, this experiment has used three medical images here to check out the effectiveness of our proposed filtering algorithm in the field of medical application. In Table 3.2, Table 3.6, and Table 3.8 all the three images are corrupted with salt & pepper, gaussian, speckle noise with variance 0.1 and denoised them with our proposed algorithm and six other well-known filters. The respected image outputs are shown below. From the output image it is clearly seen that compare to other image outputs the proposed algorithm provides clearer output. Therefore, any kind of tumor, cyst, cancer cell etc. can be easily detected by using the proposed algorithm. In Table 3.4 and Table 3.10 these medical images are corrupted with salt & pepper, speckle etc. noise with variance 0.5 and corresponding results shows that the proposed algorithm is also better than the other filtering methods for higher degradation values. Finally, the resulted images are presented in Figure 3.4, Figure 3.5, and Figure 3.6 respectively.

Table 3.2: PSNR, IQI, and SSIM values after removal of salt & pepper noise with variance 0.1 for DS4, DS5, and DS6.

Image	**DS1**			**DS2**			**DS3**		
Method	**PSNR**	**IQI**	**SSIM**	**PSNR**	**IQI**	**SSIM**	**PSNR**	**IQI**	**SSIM**
Noisy	13.8037	0.9290	0.7959	14.0948	0.9282	0.7291	14.1253	0.9124	0.7105
Median	28.0691	0.9910	0.9367	28.6344	0.9937	0.9260	35.9328	0.9952	0.9718
Wiener	19.6243	0.9543	0.3315	19.5560	0.9537	0.3211	20.2893	0.9398	0.4624
Lee	20.4705	0.9699	0.9441	21.3420	0.9707	0.9308	22.2023	0.9177	0.9385
Kaun	19.3828	0.9658	0.9243	21.9042	0.9759	0.9364	22.9332	0.9570	0.9440
Frost	15.8903	0.9401	0.8528	16.5641	0.9475	0.8154	18.6583	0.9186	0.8692
Proposed	**26.2664**	**0.9857**	**0.9869**	**26.5207**	**0.9930**	**0.9803**	**37.0165**	**0.9961**	**0.9981**
Sequence	**3-3-3-3**			**3-3-3-3**			**3-3-3-3**		

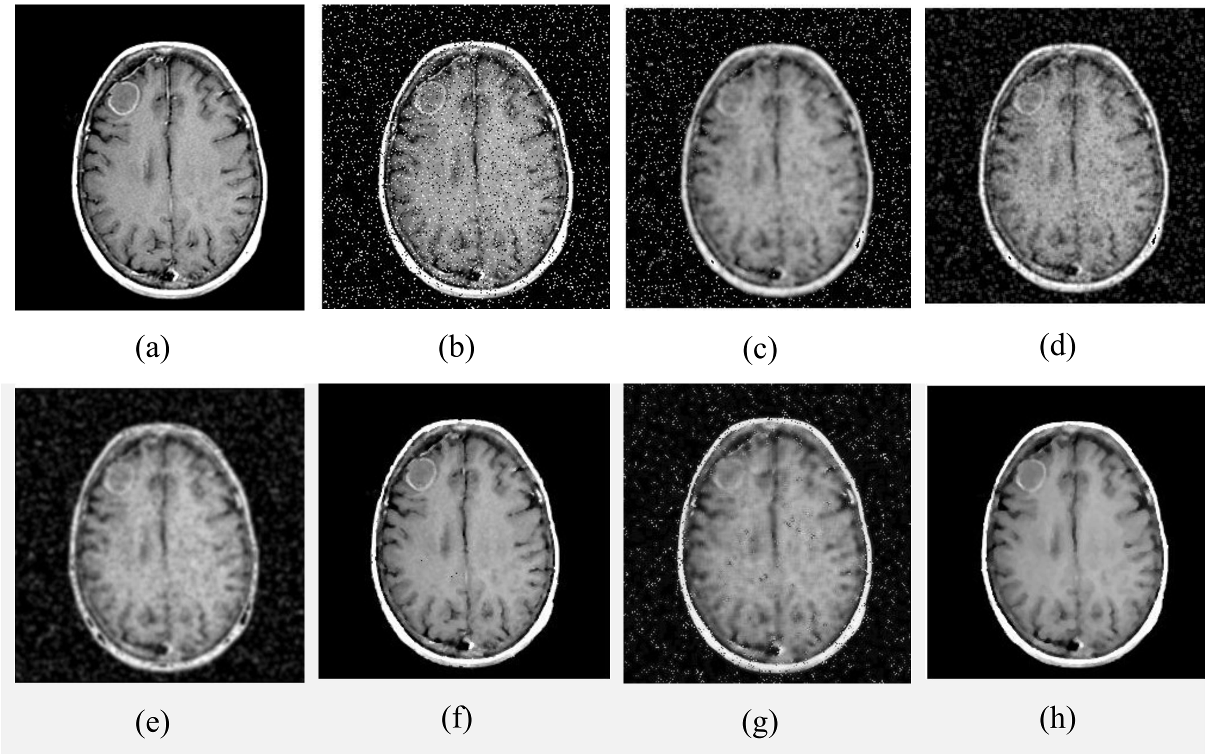


Figure 3.4: Resulting images after Denoising performance for salt & pepper noise for DS1 (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

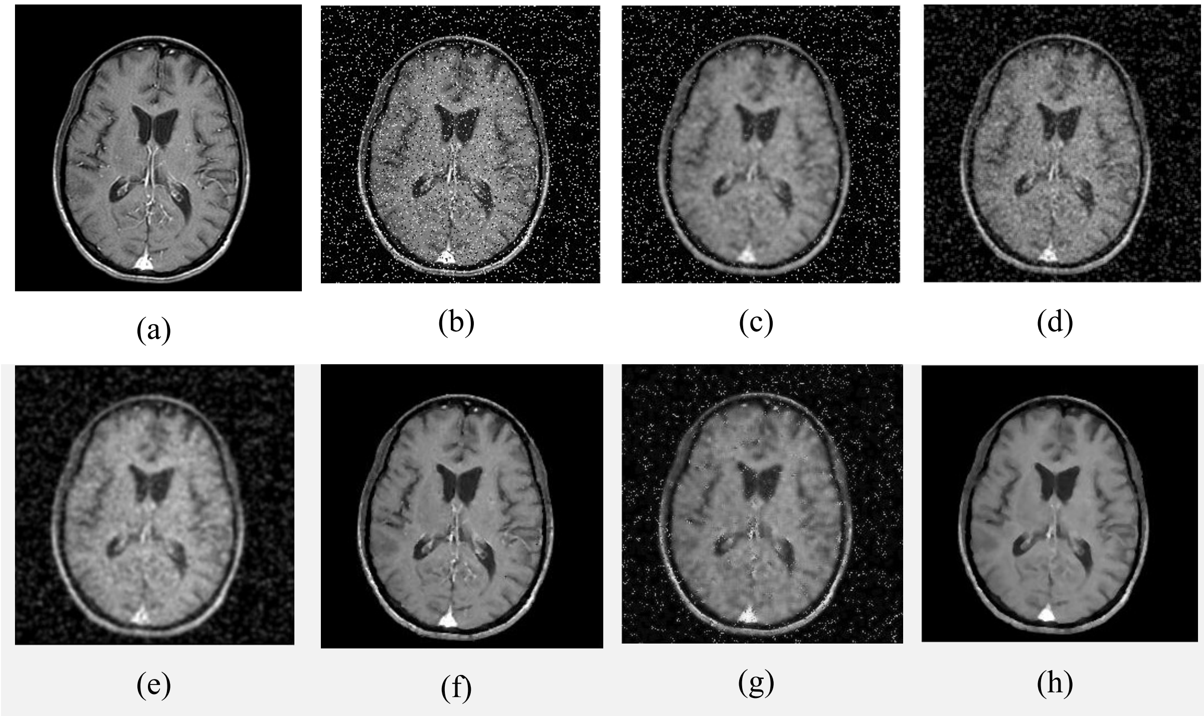


Figure 3.5: Resulting images after Denoising performance for salt & pepper noise for DS2 (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

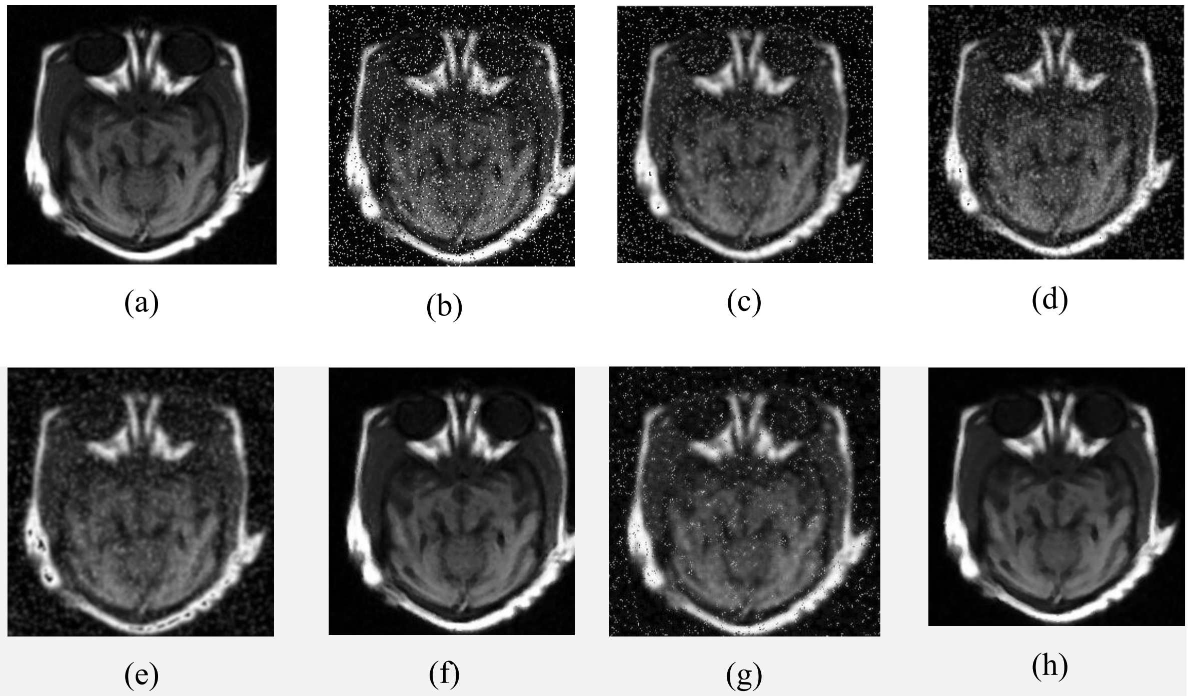


Figure 3.6: Resulting images after Denoising performance for salt & pepper noise for DS3 (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

3.2.3. General Purpose Images (Variance 0.5)

In this experiment a set of test images like are corrupted with salt & pepper noise with variance 0.5 and then the degraded images are denoised with proposed algorithm and six other existing filters. The best result is highlighted with red color and obtained sequence is also highlighted.

Simulated results given in Table 3.3 shows that the proposed algorithm gives better results for all three parameters, compared to six other filters which are used here. The three obtained sequence for Einstein, Lena and cameraman is 4-3-1-1, 4-3-1-4, 4-6-8-4. 4-3-1-1 means Median 5 ×5, Median 3×3, Mean 3×3 and Mean 3×3 filters are used sequentially to get the best output for Einstein image, 4-3-1-4 means Median 5 ×5, Median 3×3, Mean 3×3 and Median 5 ×5 filters are used sequentially to provide the best output for Lena image, and 4-6-8-4 means Median 5 ×5, Wiener 5 ×5, Gaussian and Median 5 ×5 filters are used sequentially and produces the best result for cameraman image which is corrupted with speckle noise with variance 0.5. Finally, the resulted images are presented in Figure 3.7, Figure 3.8, and Figure 3.9 respectively.

Table 3.3: PSNR, IQI, and SSIM values after removal of salt & pepper noise with variance 0.5 for Einstein, Lena, Cameraman images.

Image	Einstein			Lena			Cameraman		
Method	**PSNR**	**IQI**	**SSIM**	**PSNR**	**IQI**	**SSIM**	**PSNR**	**IQI**	**SSIM**
Noisy	8.5960	0.7536	0.1356	8.4445	0.7704	0.1955	8.1165	0.7997	0.2831
Median	15.3211	0.9204	0.2274	15.2155	0.9408	0.2276	14.2858	0.9240	0.2324
Wiener	18.0183	0.7535	0.2315	17.4166	0.8231	0.2348	15.5063	0.8621	0.2014
Lee	16.4820	0.7218	0.4826	16.1094	0.7728	0.5808	14.7992	0.8268	0.6337
Kuan	18.6507	0.7947	0.6068	17.9583	0.8671	0.6720	15.8579	0.8967	0.6744
Frost	17.7302	0.7578	0.5541	17.3587	0.8278	0.6461	15.2786	0.8586	0.6557
Proposed	**25.3981**	**0.9553**	**0.9270**	**27.7605**	**0.9830**	**0.9756**	**22.0872**	**0.9591**	**0.9460**
Sequence	**4-3-1-1**			**4-3-1-4**			**4-6-8-4**		

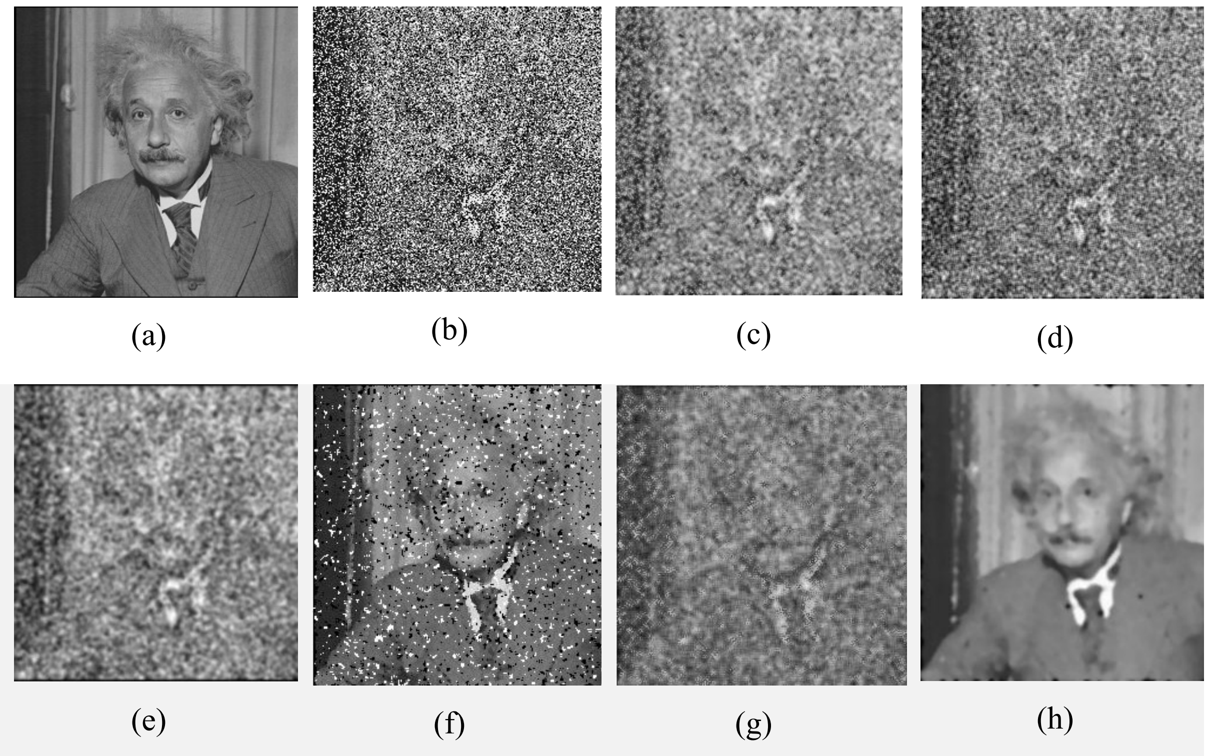


Figure 3.7: Resulting images after Denoising for salt & pepper noise for Einstein (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

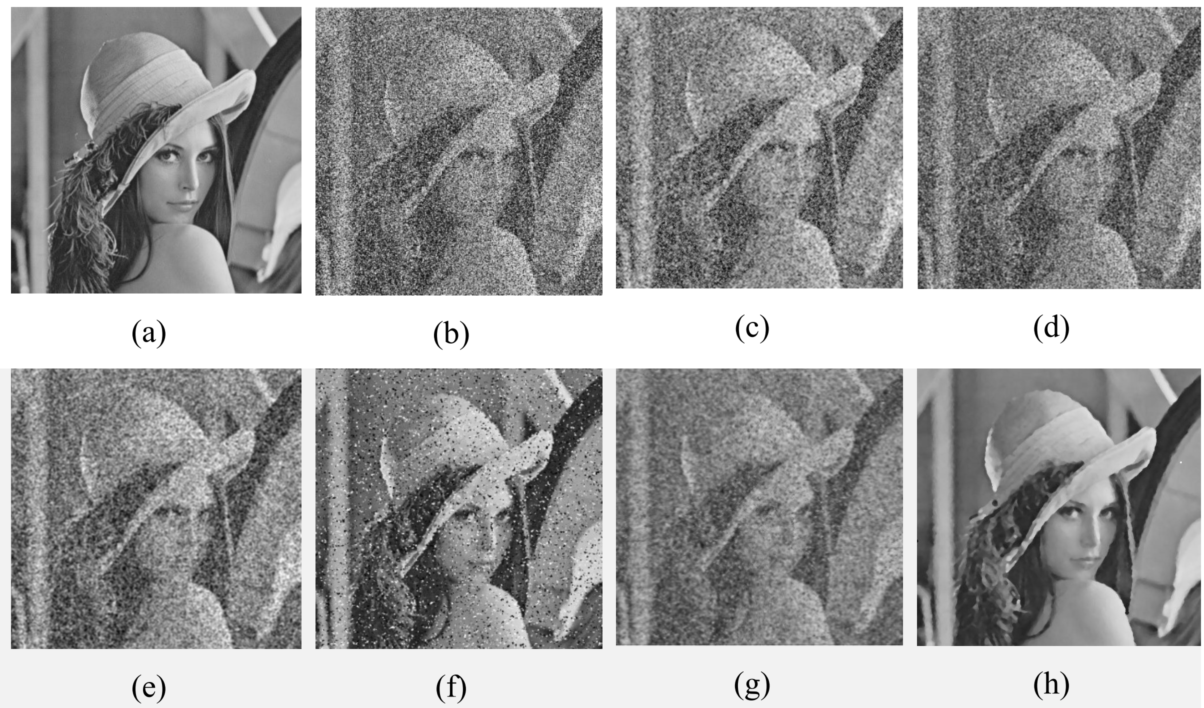


Figure 3.8: Resulting images after Denoising for salt & pepper noise for Lena (a) Original image, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

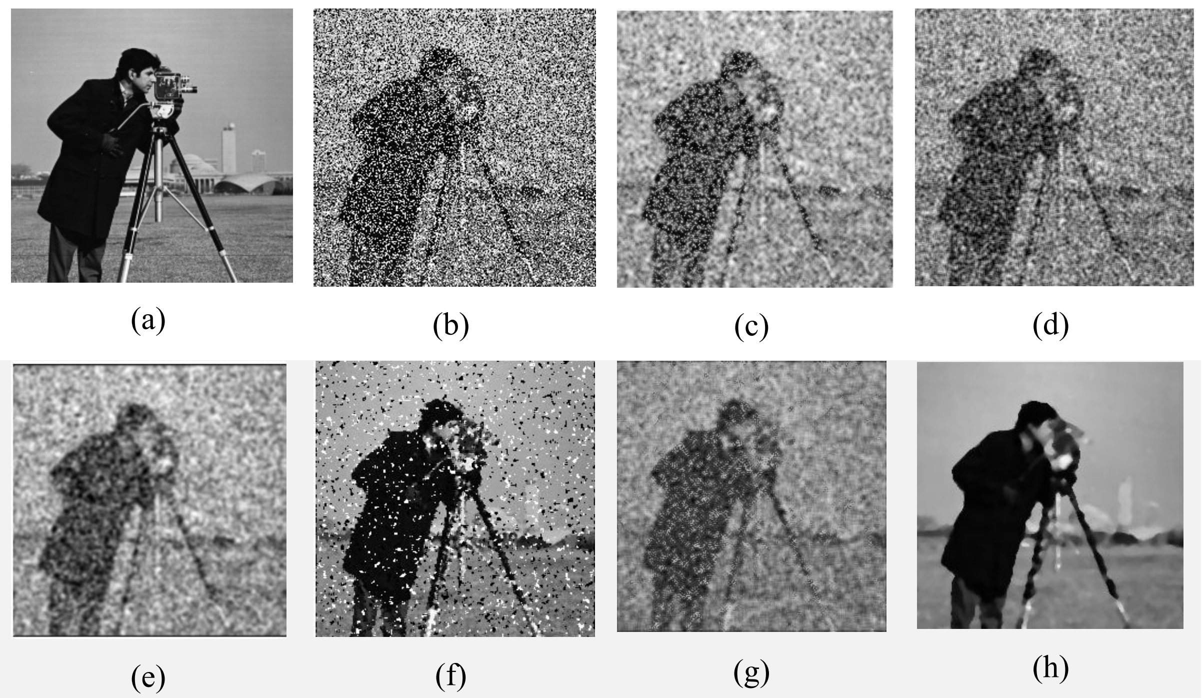


Figure 3.9: Resulting images after Denoising for salt & pepper noise for cameraman (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

3.2.4. Medical Images (Variance 0.5)

In this experiment a set of test images like DS3, DS2, DS1 are corrupted with salt & pepper noise with variance 0.5 and then the degraded images are denoised with proposed algorithm and six other existing filters. To compare the results with all other filters PSNR, IQI, and SSIM these three parameters are used here. The best result is highlighted with red color and obtained sequence is also highlighted.

Simulated results given in Table 3.4 shows that the proposed algorithm gives better results for all three parameters, compared to six other filters which are used here. The three obtained sequence for DS3, DS2, DS1 are 3-4-1-3, 4-4-1-12, 4-5-4-3. 3-4-1-3 means Median 3×3, Median 5 ×5, Mean 3×3, Median 3×3 filters are used sequentially to get the best output for DS3 image, 4-4-1-12 means Median 5 ×5, Median 5 ×5, Mean 3×3, Unsharp Masking filters are used sequentially to provide the best output for DS2 image, and 3-4-1-3 means Median 3×3, Median 5 ×5, Mean 3×3, Median 3×3 are used sequentially and produces the best result for DS1 image which is corrupted with salt & pepper noise with variance 0.5. Finally, the resulted images are presented in Figure 3.10, Figure 3.11, and Figure 3.12 respectively.

Table 3.4: PSNR, IQI, and SSIM values after removal of salt & pepper noise with variance 0.5 for DS3, DS2, and DS1 images

Image	**DS3**			**DS2**			**DS1**		
Method	**PSNR**	**IQI**	**SSIM**	**PSNR**	**IQI**	**SSIM**	**PSNR**	**IQI**	**SSIM**
Noisy	7.0258	0.4597	0.2019	7.0288	0.5797	0.2201	6.7600	0.6072	0.2860
Median	14.0517	0.8960	0.3305	13.6334	0.9115	0.2981	13.5059	0.9138	0.3169
Wiener	13.1931	0.6663	0.2047	12.8926	0.7558	0.1405	12.4862	0.7914	0.1409
Lee	12.5445	0.5923	0.5325	12.4995	0.7068	0.5427	11.8995	0.7404	0.5850
Kuan	13.5898	0.6909	0.6092	13.4594	0.8076	0.6108	12.6427	0.8463	0.6228
Frost	12.1622	0.5736	0.4978	11.8263	0.6873	0.4880	11.2890	0.7170	0.5391
Proposed	**28.1711**	**0.9772**	**0.9852**	**22.7986**	**0.9764**	**0.9541**	**20.4414**	**0.9668**	**0.9487**
Sequence	**3-4-1-3**			**4-4-1-12**			**4-5-4-3**		

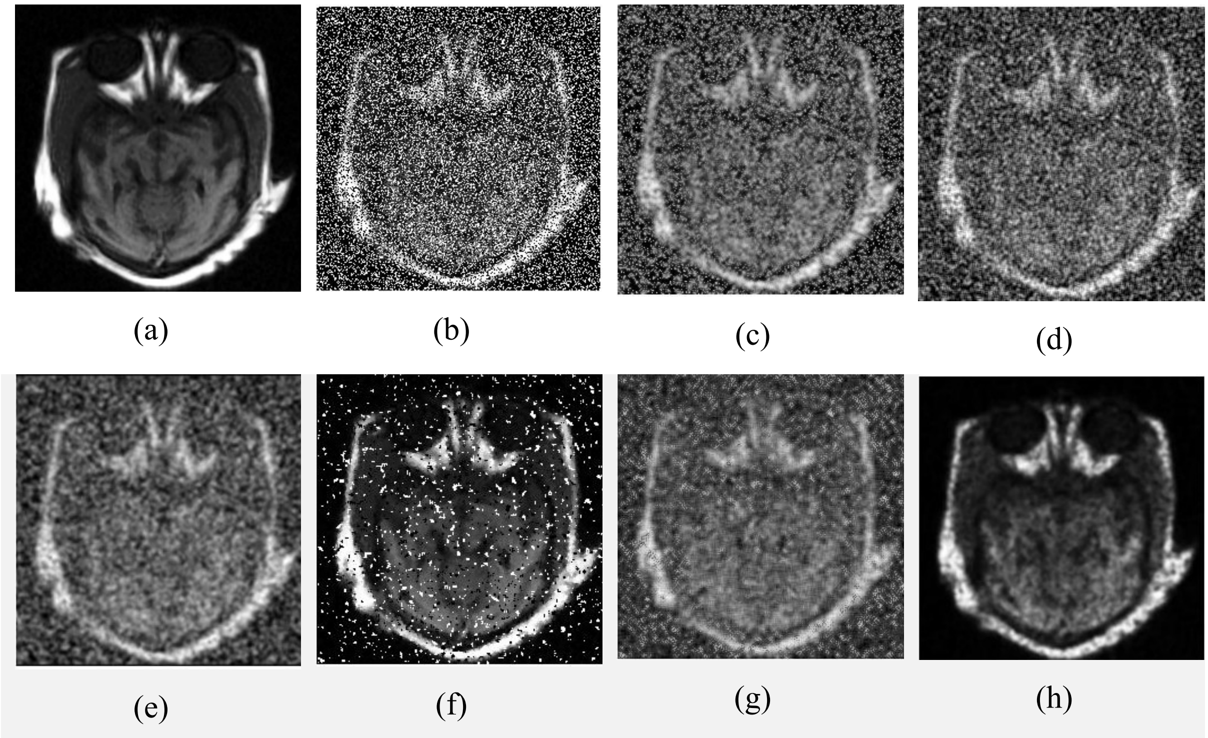


Figure 3.10: Resulting images after Denoising for salt & pepper noise for DS3 image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

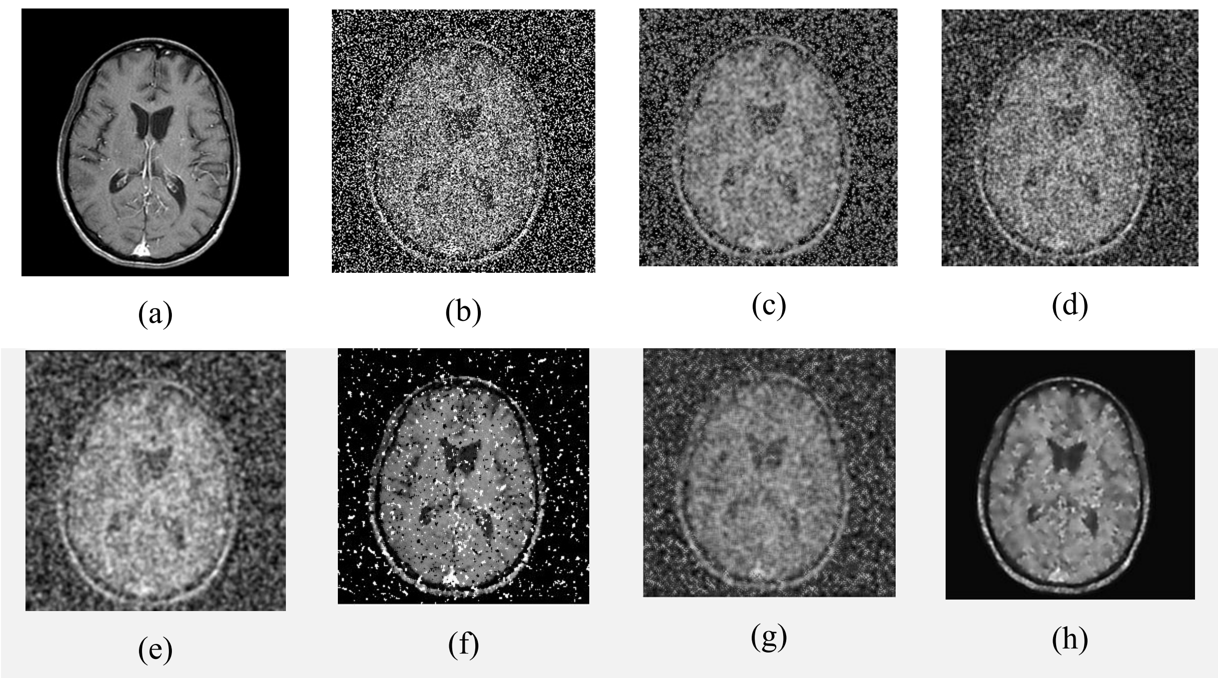


Figure 3.11: Resulting images after Denoising for salt & pepper noise for DS2 image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

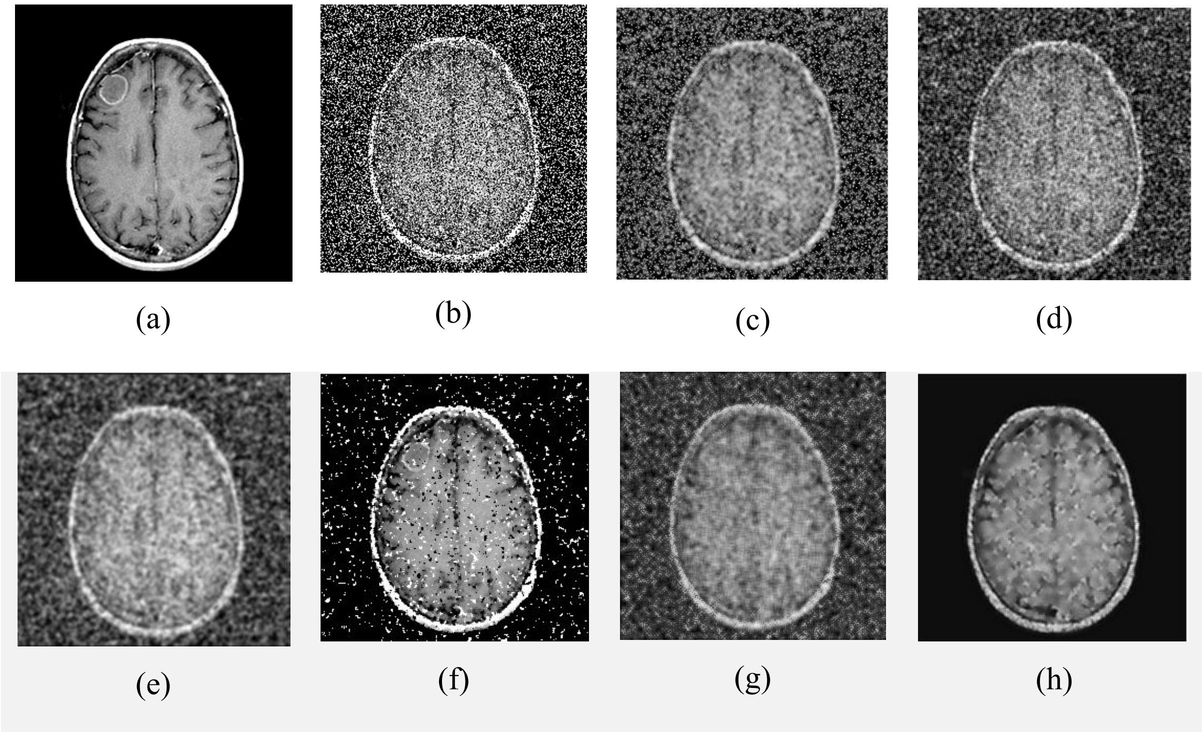


Figure 3.12: Resulting images after Denoising for salt & pepper noise for DS1 image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

3.3 Experiments on Gaussian Noise Reduction

In this section we have discussed the experimental results for the reduction of gaussian noise considering few general-purpose images and few medical images

3.3.1. General Purpose Images

In this experiment also a set of test images like Lena, Cameraman, Einstein are corrupted with gaussian noise assuming variance 0.1 and then the degraded images are denoised with proposed algorithm and six other existing algorithms. To compare the results with all other algorithms PSNR, IQI, and SSIM values are measured here. The best results are highlighted with red color and obtained sequence are also highlighted.

Simulated results given in Table 3.5 shows that the proposed algorithm gives better results for all three parameters, compared to other six algorithms used here. The obtained sequence 5-5-5-5 means wiener 3×3 filter is used 4 times in sequence and produces the best result for all the three images which are corrupted with gaussian noise with variance 0.1. Finally, the resulted images are presented in Figure 3.13, Figure 3.14, and Figure 3.15 respectively.

Table 3.5: PSNR, IQI, and SSIM values after removal of Gaussian noise with variance 0.1 for Lena, Cameraman and Einstein images.

Image	Lena			Cameraman			Einstein		
Method	**PSNR**	**IQI**	**SSIM**	**PSNR**	**IQI**	**SSIM**	**PSNR**	**IQI**	**SSIM**
Noisy	17.103	0.902	0.863	17.039	0.941	0.907	17.024	0.855	0.792
Median	19.669	0.908	0.856	17.317	0.905	0.215	19.108	0.916	0.549
Wiener	20.468	0.901	0.864	19.427	0.927	0.337	19.419	0.938	0.663
Lee	24.789	0.962	0.952	18.667	0.950	0.951	19.157	0.922	0.926
Kaun	25.108	0.972	0.955	17.983	0.942	0.921	18.759	0.929	0.893
Frost	26.880	0.972	0.969	18.521	0.948	0.945	25.996	0.945	0.936
Proposed	**27.290**	**0.987**	**0.978**	**19.556**	**0.965**	**0.970**	**26.617**	**0.954**	**0.947**
Sequence	**3-3-7-7**			**3-1-4-3**			**4-7-1-1**		

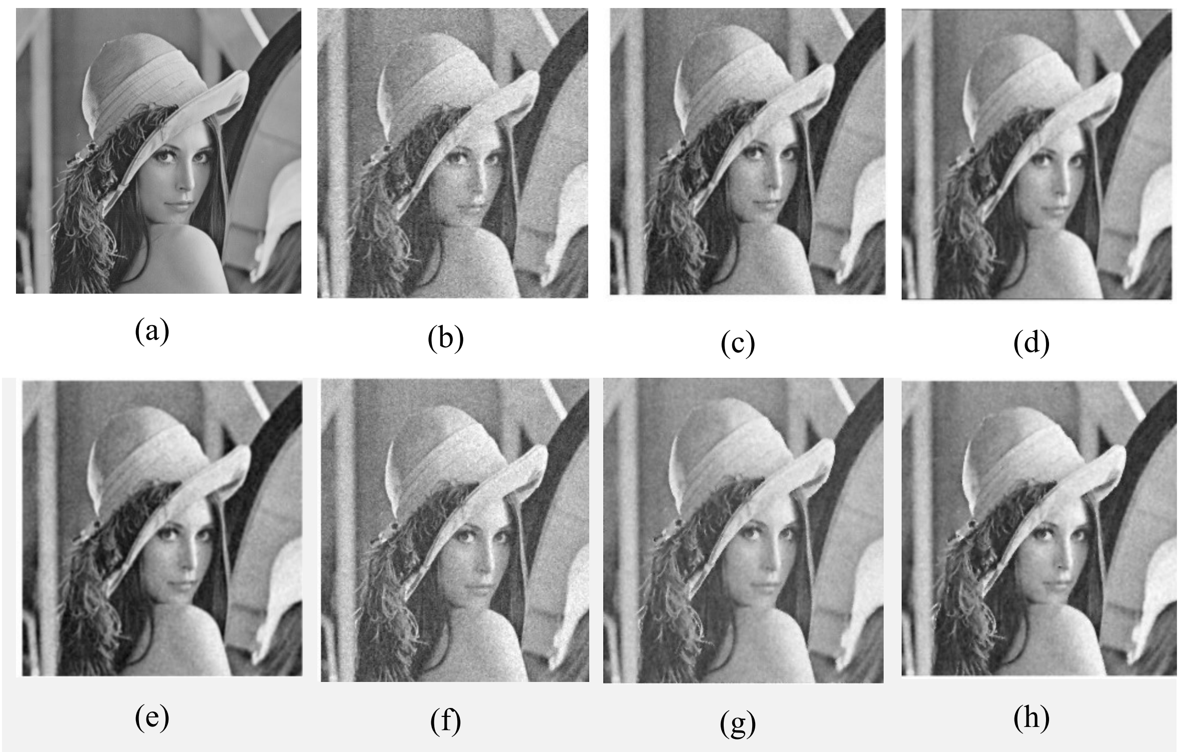


Figure 3.13: Resulting images after Denoising performance for gaussian noise for Lena image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

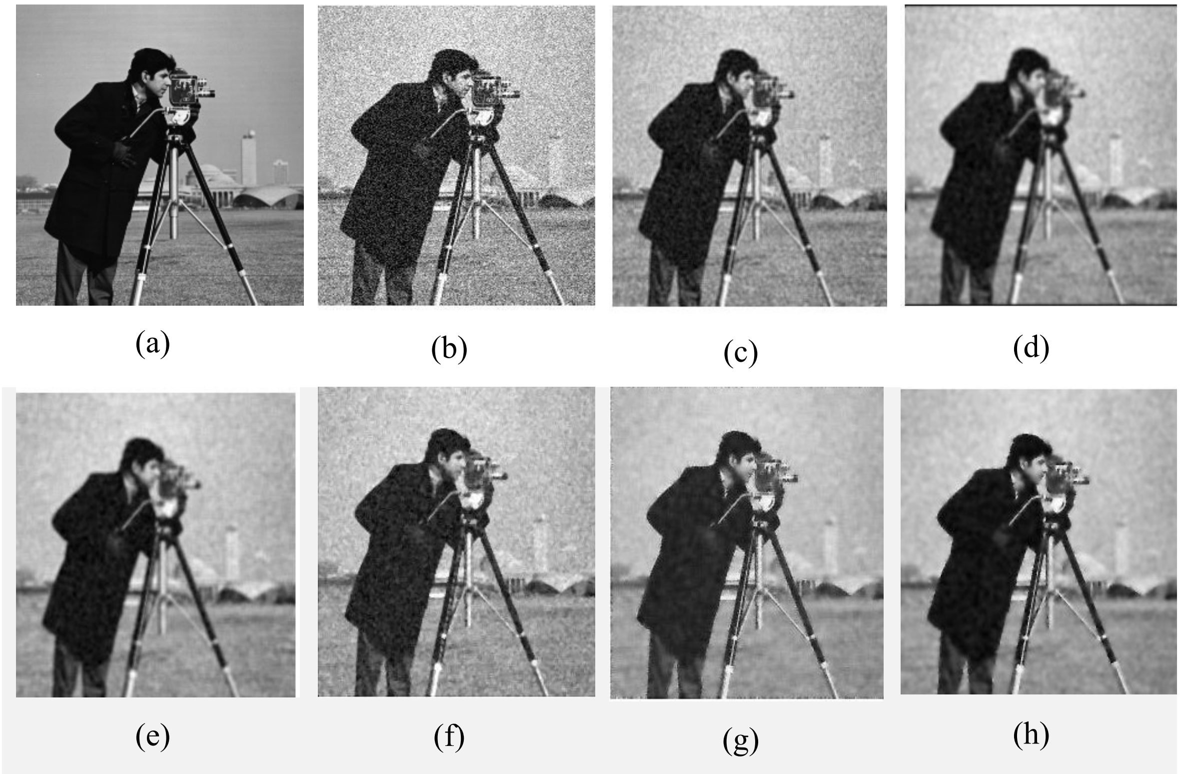


Figure 3.14: Resulting images after Denoising performance for gaussian noise for Cameraman image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

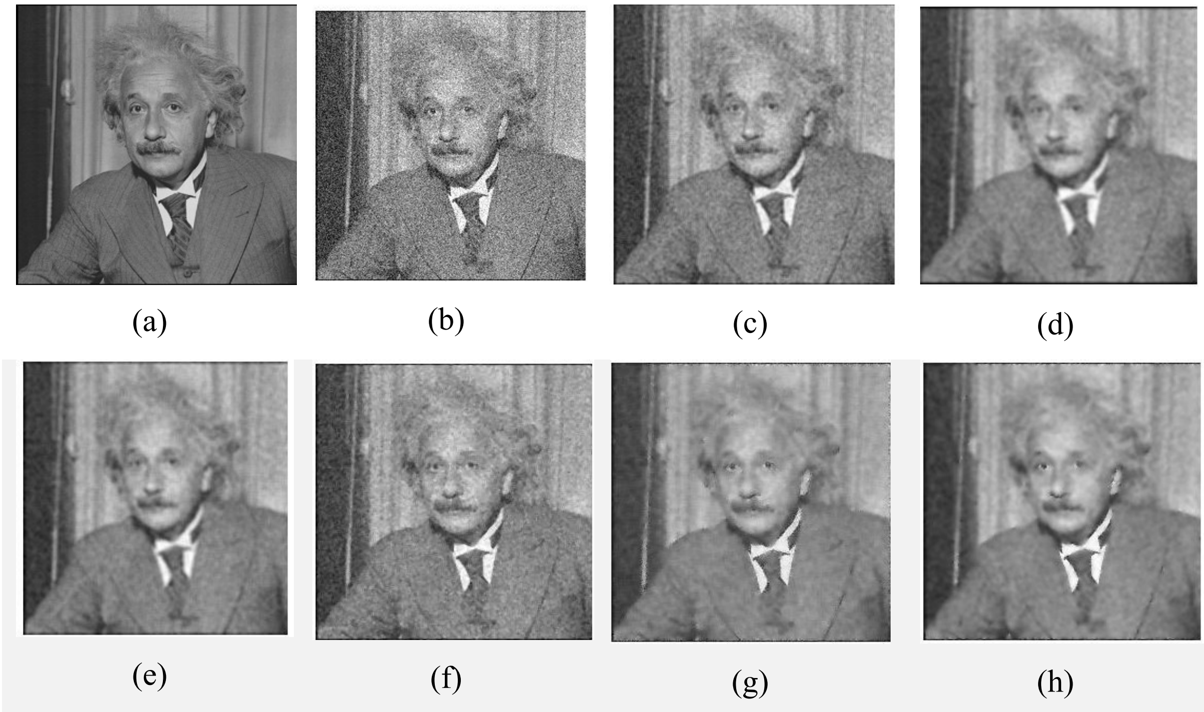


Figure 3.15: Resulting images after Denoising performance for gaussian noise for Einstein image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

3.3.2. Medical Images

In this experiment a set of test images like DS1, DS2, and DS3 are corrupted with gaussian noise with variance 0.1 and then the degraded images are denoised with proposed algorithm and six other existing filters. To compare the results with all other filters PSNR, IQI, and SSIM, these three parameters are used here. The best result is highlighted with red color and obtained sequence is also highlighted.

Simulated results given in Table 3.6 shows that the proposed algorithm provides better results for all three parameters, compared to six other filters which are used here. The obtained sequence 5-5-5-5 means Median 5×5 filter is used 4 times in sequence and produces the best result for all the three images which are corrupted with speckle noise with variance 0.1. The best result is highlighted with red color and obtained sequence is also highlighted. Finally, the resulted images are presented in Figure 3.16, Figure 17, and Figure 3.18 respectively.

Table 3.6: PSNR, IQI, and SSIM values after removal of gaussian noise with variance 0.1 for DS4, DS5 and DS6.

Image	**DS4**			**DS5**			**DS6**		
Method	**PSNR**	**IQI**	**SSIM**	**PSNR**	**IQI**	**SSIM**	**PSNR**	**IQI**	**SSIM**
Noisy	17.2235	0.9234	0.8936	17.1314	0.8937	0.8535	17.2451	0.8142	0.8437
Median	19.0054	0.9318	0.3264	18.9355	0.9151	0.3355	19.3647	0.8937	0.4648
Wiener	19.0324	0.9299	0.3399	18.9155	0.9074	0.3421	19.4420	0.9011	0.5518
Lee	18.4256	0.9635	0.9154	18.7275	0.9089	0.8957	19.3741	0.9010	0.9031
Kaun	17.5151	0.9145	0.8944	18.3938	0.9058	0.8855	19.0467	0.9063	0.8901
Frost	17.9905	0.9184	0.9060	18.4782	0.9073	0.8881	19.4309	0.9048	0.9037
Proposed	**18.8838**	**0.9255**	**0.9234**	**19.0267**	**0.9154**	**0.9004**	**19.7698**	**0.9096**	**0.9116**
Sequence	**5-5-5-5**			**5-5-5-5**			**5-5-5-5**		

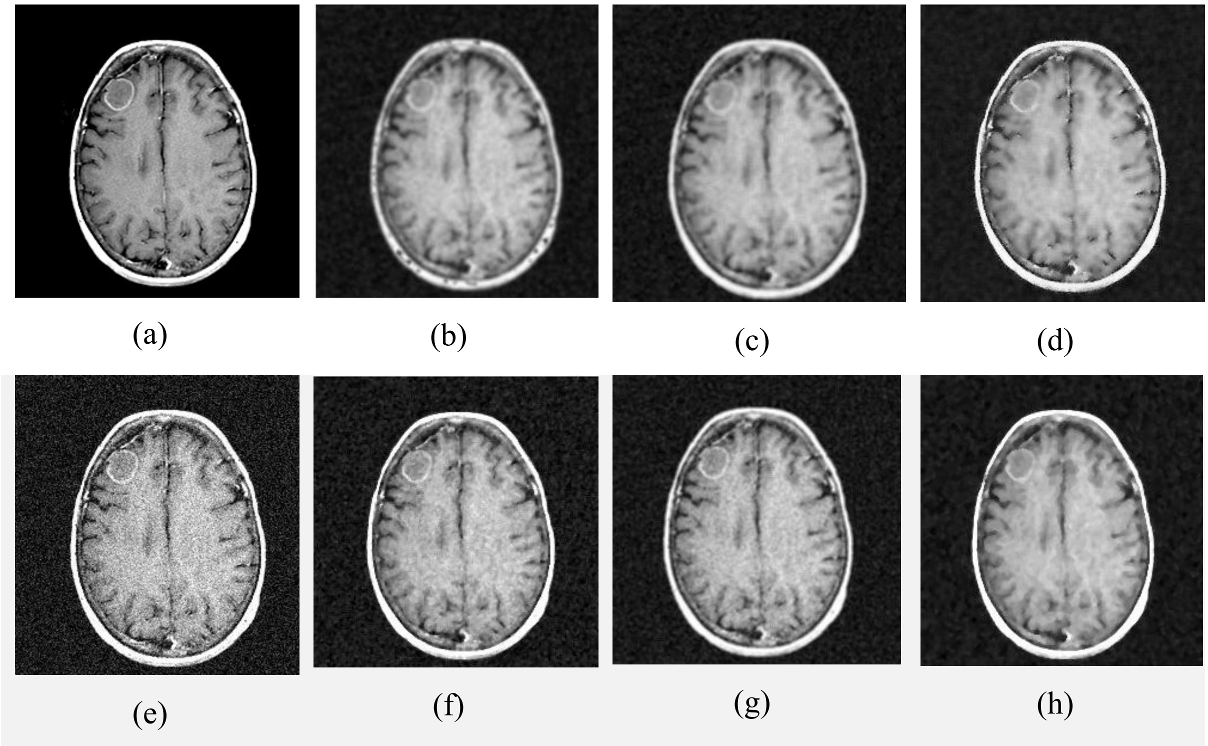


Figure 3.16: Resulting images after Denoising for gaussian noise for DS1 (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

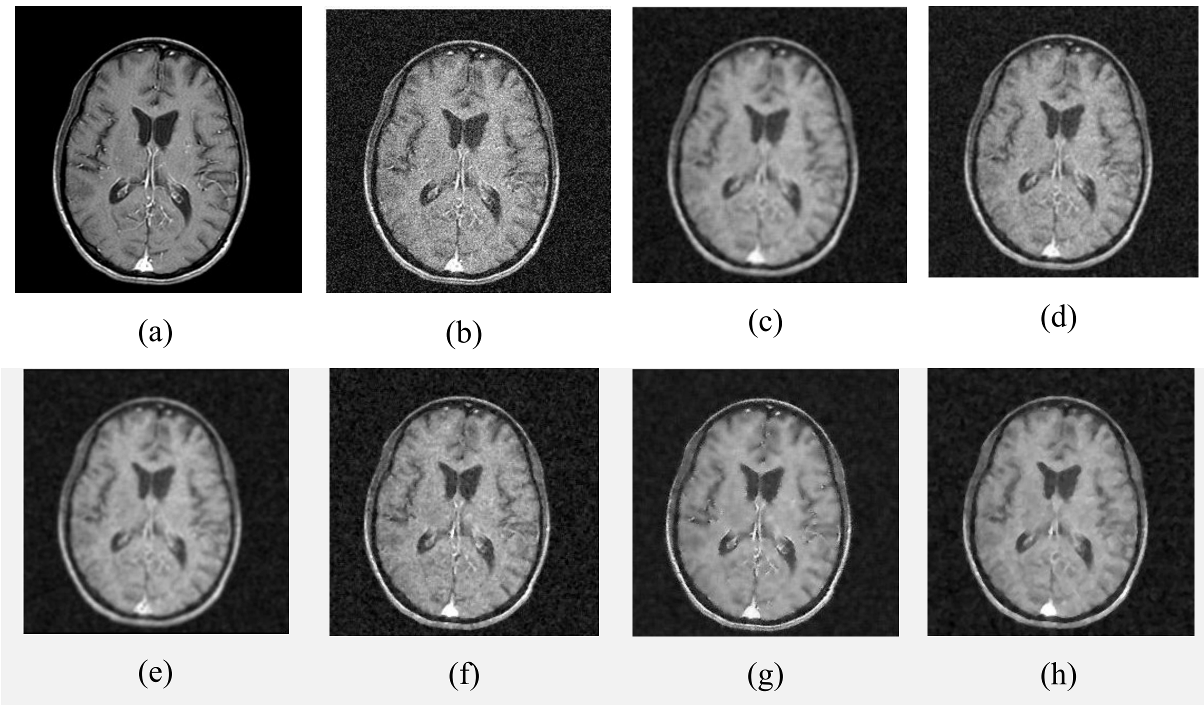
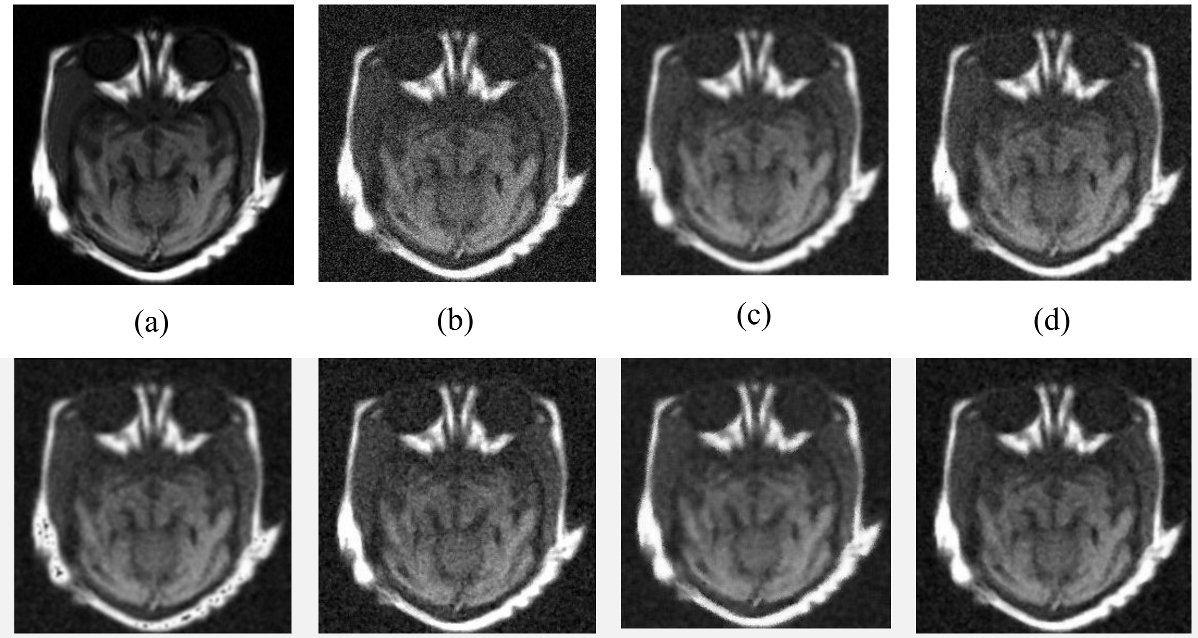


Figure 3.17: Resulting images after Denoising performance for gaussian noise for DS2 (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 3.18: Resulting images after Denoising performance for gaussian noise for DS3 (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

3.4. Experiments on Speckle Noise Reduction:

In this section we have discussed the experimental results for the reduction of gaussian noise considering few general-purpose images and few medical images.

3.4.1. General Purpose Images (Variance 0.1)

In this experiment also a set of test images like Lena, Cameraman, Einstein are corrupted with speckle noise with variance 0.1 and then the degraded images are denoised with proposed algorithm and six other existing filters. To compare the results with all other filters PSNR, IQI, and SSIM, these three parameters are used here. The best result is highlighted with red color and obtained sequence is also highlighted.

Simulated results given in Table 3.7 shows that the proposed algorithm provides better results for all three parameters, compared to other six filters used here. The obtained sequence 5-5-5-5 means wiener 3×3 filter is used 4 times in sequence and produces the best result for all the three images which are corrupted with speckle noise with variance 0.1. Finally, the resulted images are presented in Figure 3.19, Figure 3.20, and Figure 3.21 respectively.

Table 3.7: PSNR, IQI, and SSIM values after removal of speckle noise with variance 0.1 for Lena, cameraman and Einstein image.

Image	Lena			Cameraman			Einstein		
Method	**PSNR**	**IQI**	**SSIM**	**PSNR**	**IQI**	**SSIM**	**PSNR**	**IQI**	**SSIM**
Noisy	15.997	0.812	0.732	15.782	0.853	0.817	17.086	0.786	0.682
Median	22.299	0.943	0.920	17.317	0.905	0.215	19.108	0.916	0.549
Wiener	23.177	0.959	0.932	19.427	0.927	0.337	19.419	0.938	0.663
Lee	24.728	0.962	0.952	22.523	0.961	0.952	19.157	0.922	0.926
Kaun	25.108	0.972	0.955	21.372	0.953	0.935	18.759	0.929	0.893
Frost	26.880	0.972	0.969	23.423	0.962	0.960	25.996	0.945	0.936
Proposed	**27.290**	**0.987**	**0.978**	**24.284**	**0.965**	**0.970**	**26.617**	**0.954**	**0.947**
Sequence	**5-5-5-5**			**5-5-5-5**			**5-5-5-5**		

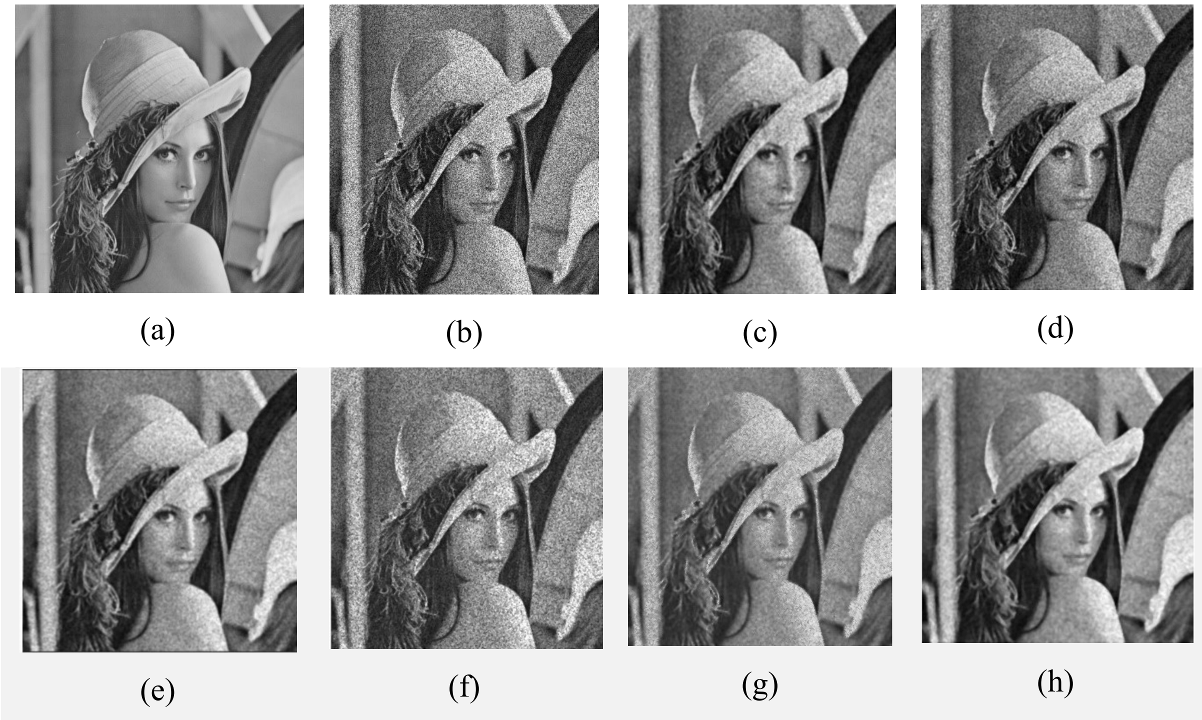


Figure 3.19: Resulting images after Denoising performance for speckle noise for Lena image a) Original, b) Noisy, c) Frost, d) Lee, e) Kuan, f) Median, g) Wiener, h) Proposed

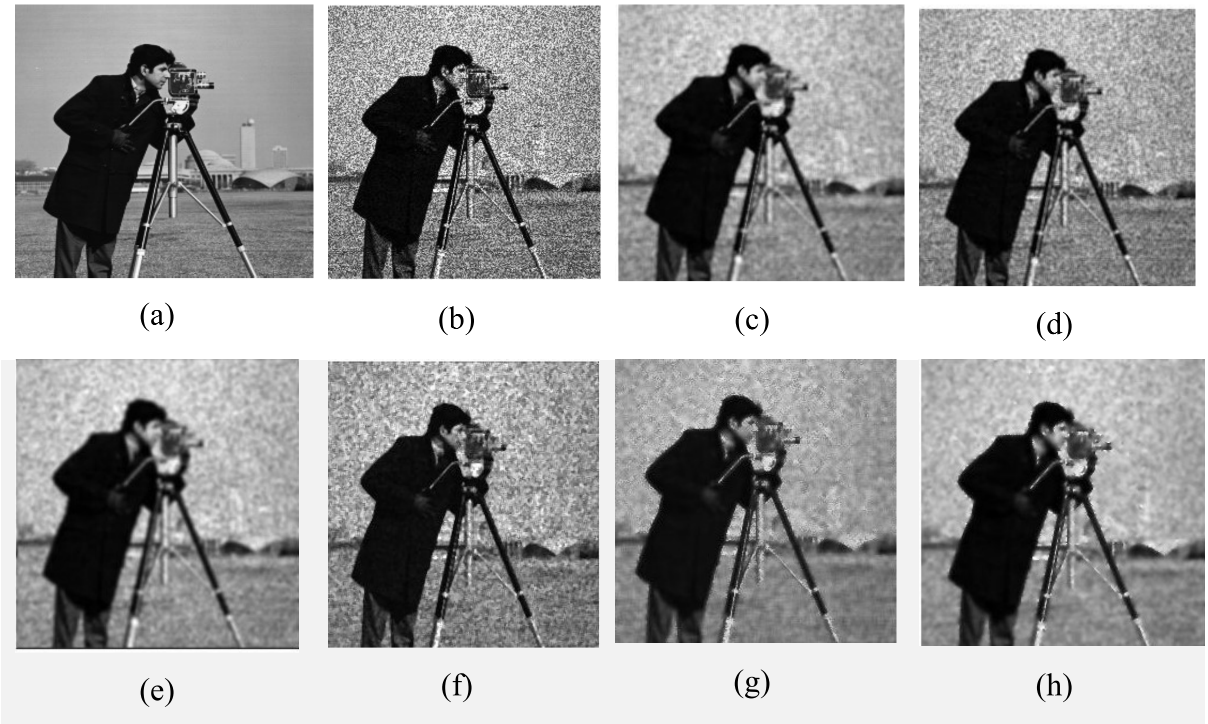


Figure 3.20: Resulting images after Denoising performance for speckle noise for Cameraman image a) Original, b) Noisy, c) Frost, d) Lee, e) Kuan, f) Median, g) Wiener, h) Proposed

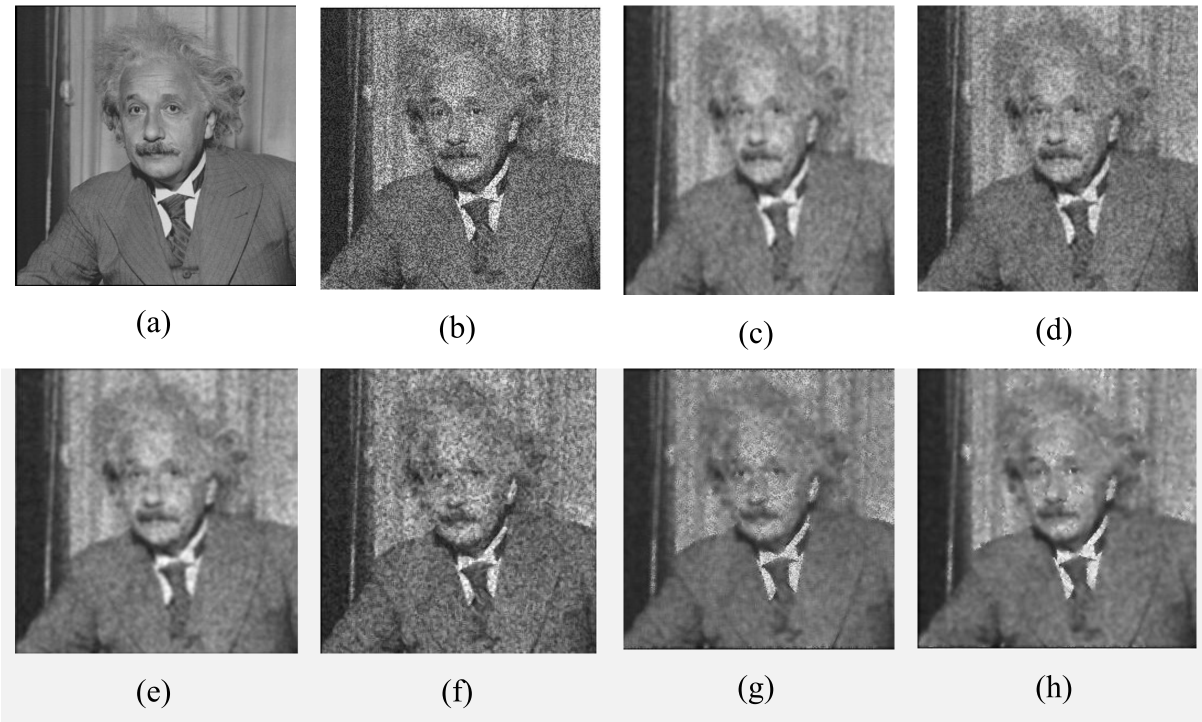


Figure 3.21: Resulting images after Denoising performance for speckle noise for Einstein image a) Original, b) Noisy, c) Frost, d) Lee, e) Kuan, f) Median, g) Wiener, h) Proposed

3.4.2. Medical Images (Variance 0.1)

In this experiment a set of test images like DS1, DS2, and DS3 are corrupted with speckle noise with variance 0.1 and then the degraded images are denoised with proposed algorithm and six other existing filters. To compare the results with all other filters PSNR, IQI, and SSIM these three parameters are used here. The best result is highlighted with red color and obtained sequence is also highlighted.

Simulated results given in Table 3.8 shows that the proposed algorithm provides better results for all three parameters, compared to six other filters which are used here. The obtained sequence 5-5-5-5 means Wiener 5×5 filter is used 4 times in sequence and produces the best result for all the three images which are corrupted with speckle noise with variance 0.1. Finally, the resulted images are presented in Figure 3.22, Figure 3.23, and Figure 3.24 respectively.

Table 3.8: PSNR, IQI, and SSIM values after removal of speckle noise with variance 0.1 for DS4, DS5, and DS6.

Image	**DS4**			**DS5**			**DS6**		
Method	**PSNR**	**IQI**	**SSIM**	**PSNR**	**IQI**	**SSIM**	**PSNR**	**IQI**	**SSIM**
Noisy	18.9862	0.9701	0.9328	20.3147	0.9901	0.9259	21.5773	0.9357	0.9340
Median	22.9316	0.9910	0.7986	24.0890	0.9924	0.7853	26.8433	0.9828	0.8366
Wiener	22.9901	0.9901	0.8004	24.5028	0.9918	0.7875	24.6769	0.9897	0.8493
Lee	23.2493	0.9859	0.9724	25.6625	0.9915	0.9760	28.4127	0.9891	0.9854
Kaun	21.5521	0.9724	0.9579	24.1293	0.9865	0.9649	27.9094	0.9841	0.9831
Frost	22.5330	0.9808	0.9666	25.0961	0.9901	0.9722	28.3441	0.9873	0.9848
Proposed	**24.2486**	**0.9867**	**0.9781**	**26.0197**	**0.9906**	**0.9780**	**29.3037**	**0.9906**	**0.9882**
Sequence	**5-5-5-5**			**5-5-5-5**			**5-5-5-5**		

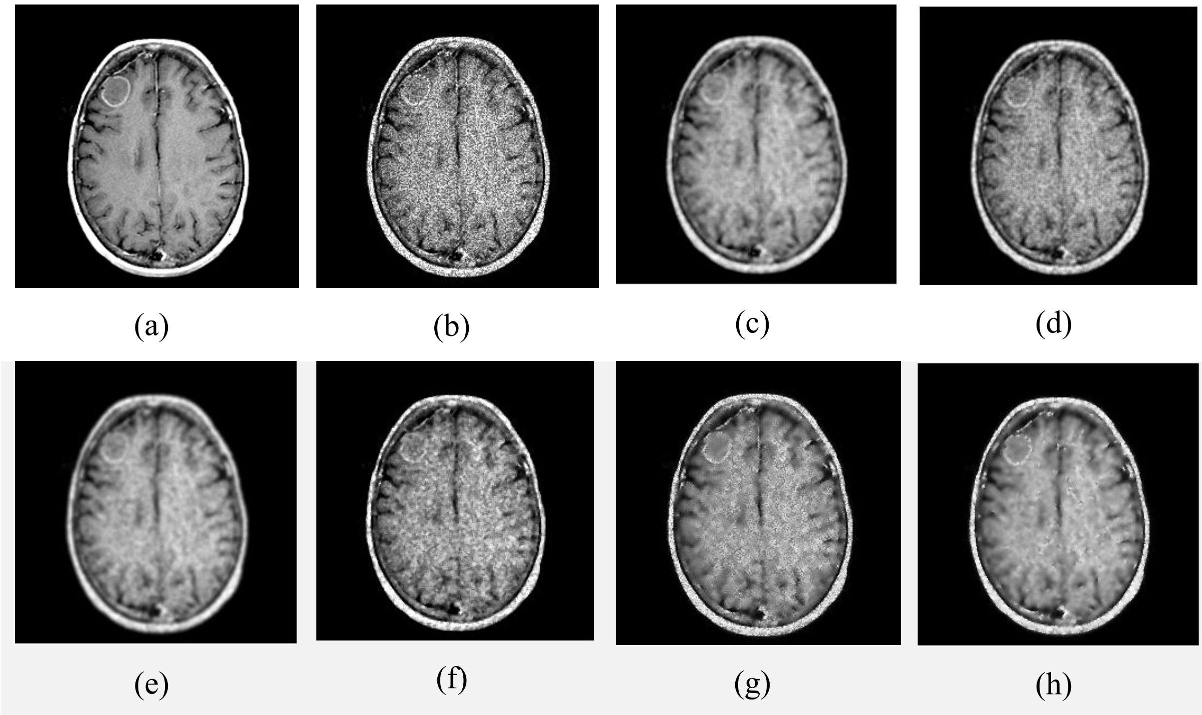


Figure 3.22: Resulting images after Denoising results for speckle noise for DS1 (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

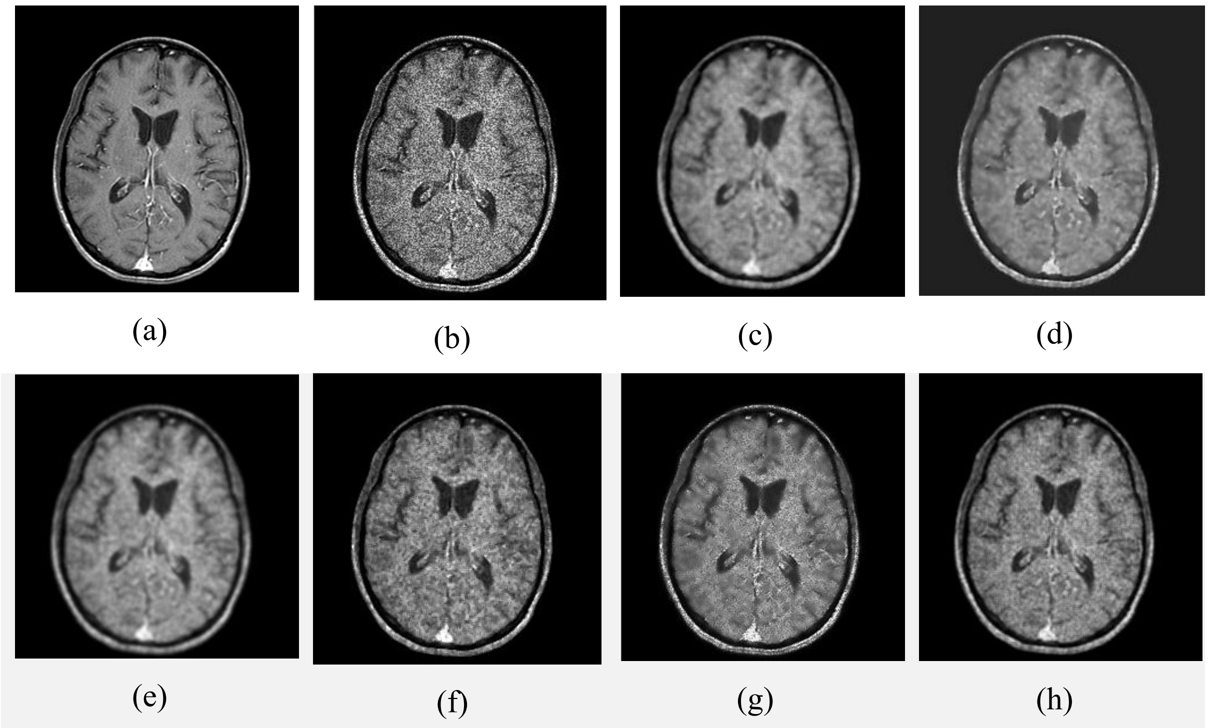


Figure 3.23: Resulting images after Denoising performance for speckle noise for DS2 (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

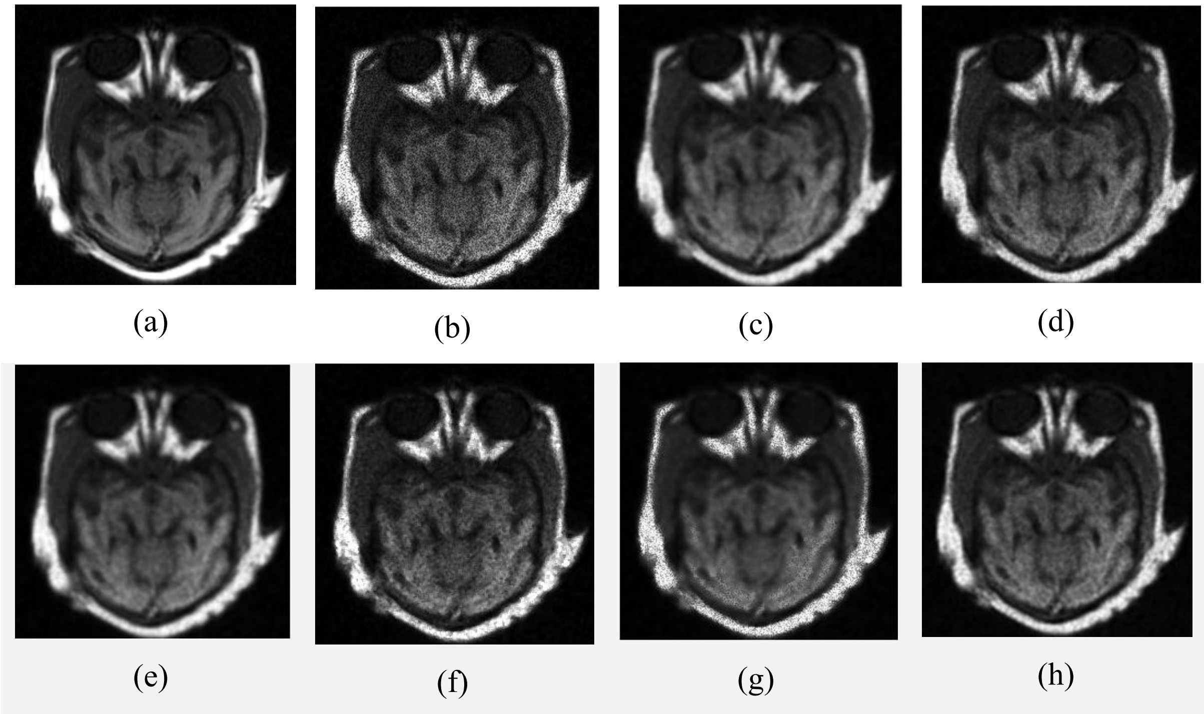


Figure 3.24: Resulting images after Denoising performance for speckle noise for DS3 (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

3.4.3 General Purpose Images (Variance 0.5)

In this experiment a set of test images like Einstein, Lena and cameraman are corrupted with speckle noise with variance 0.5 and then the degraded images are denoised with proposed algorithm and six other existing filters. To compare the results with all other algorithms PSNR, IQI, and SSIM values are measured here. The best results are highlighted with red color and obtained sequence are also highlighted.

Simulated results given in Table 3.9 shows that the proposed algorithm provides better results for all three parameters, compared to six other filters which are used here. The three obtained sequence for Einstein, Lena, and cameraman images are 1-1-1-7, 3-1-4-7, 1-7-12-1. 1-1-1-7 respectively. For instance, 1-1-1-7 indicating a combination of Mean 3×3, Mean 3×3, Mean 3×3, and Wiener 7×7 filters are used sequentially to get the best output for Einstein image. Similarly, 3-1-4-7 including Median 3×3, Mean 3×3, Median 5 ×5, and Wiener 7×7 filters are used sequentially to provide the best output for Lena image, and 1-7-12-1 means Mean 3×3, Wiener 7×7, Unsharp Masking, and Mean 3×3 filters are applied sequentially and produces the best result for cameraman image which is corrupted with speckle noise with variance 0.5. Finally, the resulted images are presented in Figure 3.25, Figure 3.26, and Figure 3.27 respectively.

Table 3.9: PSNR, IQI, and SSIM values after removal of speckle noise with variance 0.5 for cameraman, Lena and Einstein images.

Image	Einstein			Lena			Cameraman		
Method	**PSNR**	**IQI**	**SSIM**	**PSNR**	**IQI**	**SSIM**	**PSNR**	**IQI**	**SSIM**
Noisy	10.7011	0.7148	0.3102	10.0617	0.7083	0.3624	10.1682	0.7613	0.5240
Median	15.6625	0.7788	0.1794	14.7791	0.8110	0.1616	14.1817	0.8350	0.2652
Wiener	19.8323	0.8193	0.3390	18.5632	0.8596	0.3159	17.6314	0.8780	0.3310
Lee	19.5531	0.8382	0.7614	18.4761	0.8781	0.8001	18.0653	0.9073	0.8691
Kuan	21.7713	0.8934	0.8304	20.5765	0.9359	0.8676	19.1128	0.9389	0.8924
Frost	21.7924	0.8789	0.8402	20.4263	0.9224	0.8643	19.7350	0.9373	0.9085
Proposed	**23.3240**	**0.9217**	**0.8750**	**22.5666**	**0.9446**	**0.9196**	**20.2217**	**0.9471**	**0.9155**
Sequence	**1-1-1-7**			**3-1-4-7**			**1-7-12-1**		

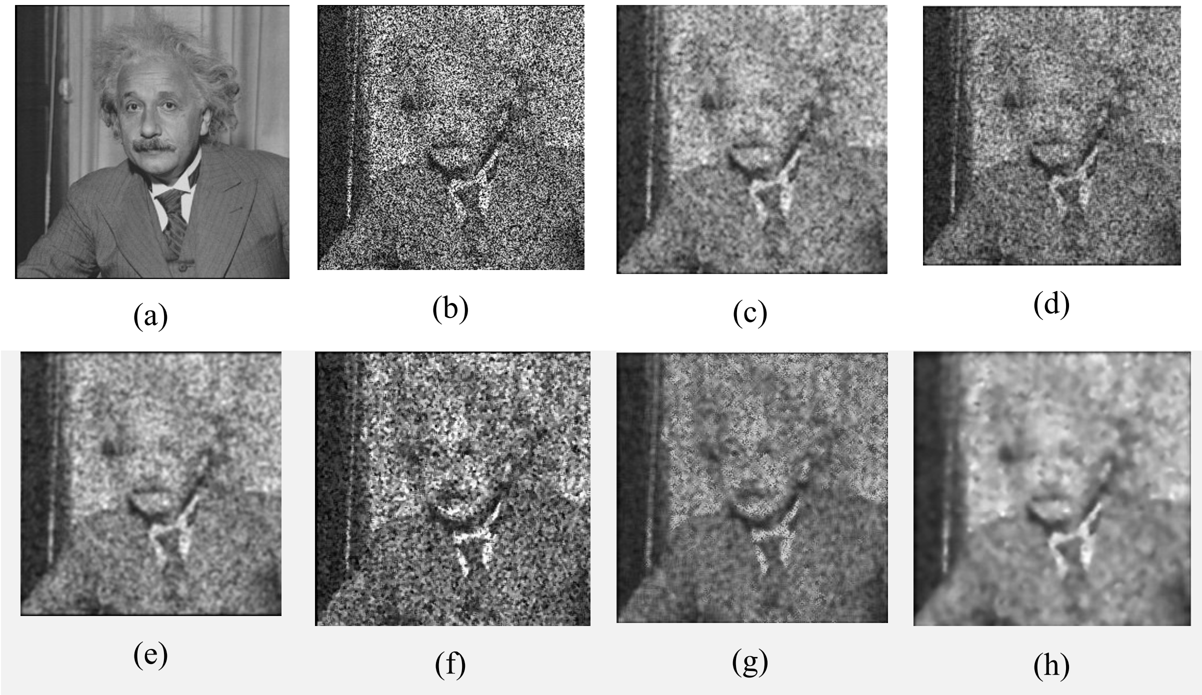


Figure 3.25: Resulting images after Denoising performance for speckle noise for Einstein (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

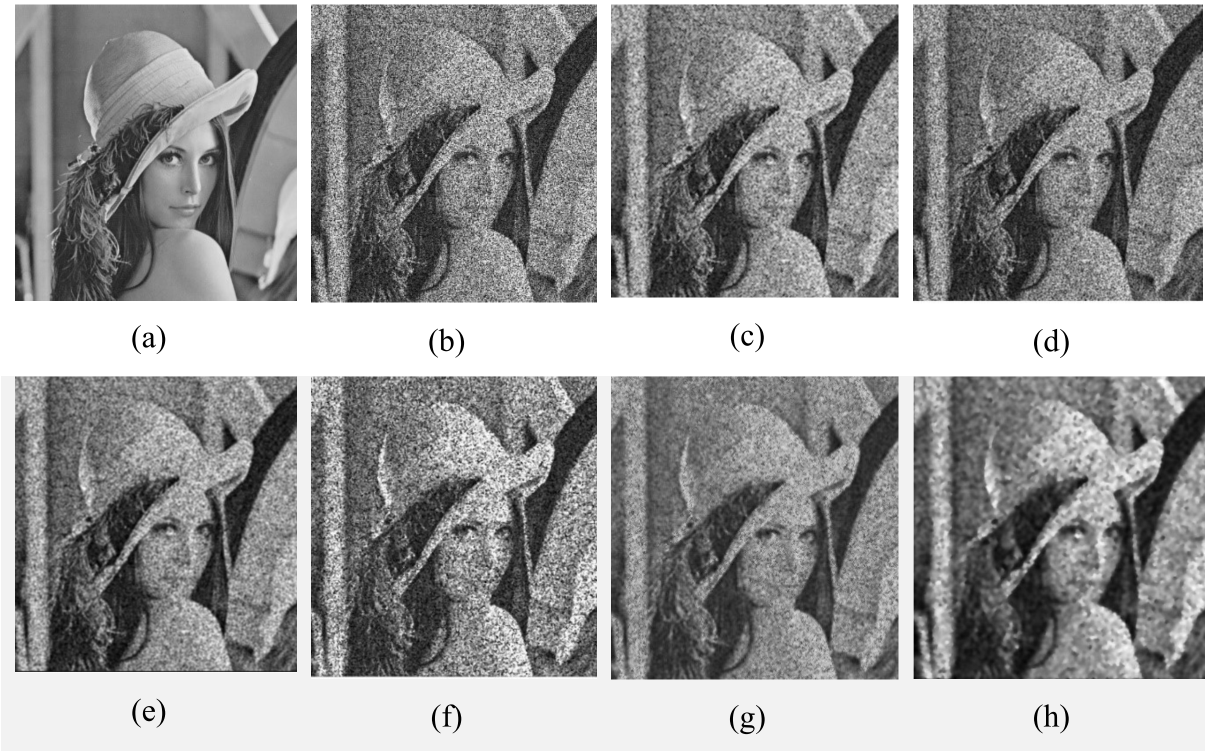


Figure 3.26: Resulting images after Denoising performance for speckle noise for Lena (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

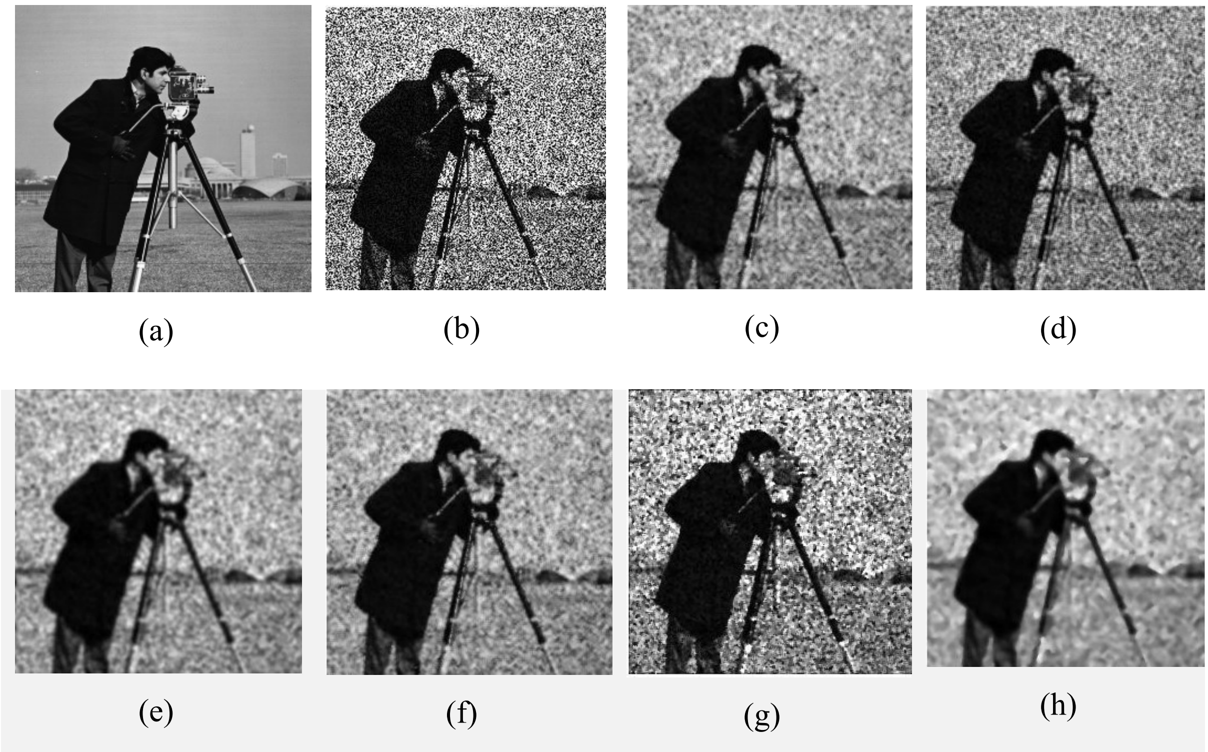


Figure 3.27: Resulting images after Denoising performance for speckle noise for cameraman (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

3.4.4. Medical Images (Variance 0.5)

In this experiment a set of test images like DS3, DS2, DS1 are corrupted with speckle noise with variance 0.5 and then the degraded images are denoised with proposed algorithm and six other existing filters. To compare the results with all other filters PSNR, SSIM, and IQI, these three parameters are used here. The best result is highlighted with red color and obtained sequence is also highlighted.

Simulated results given in Table 3.10 shows that the proposed algorithm gives better results for all three parameters, compared to six other filters which are used here. The three obtained sequence for DS3, DS2, DS1 are 4-1-1-3, 5-4-1-3, 4-3-1-1. 4-1-1-3 means Median 5 ×5, Mean 3×3, Mean 3×3, Median 3×3 filters are used sequentially to get the best output for DS3 image, 5-4-1-3 means Wiener 3×3, Median 5 ×5, Mean 3×3, Median 3×3 filters are used sequentially to provide the best output for DS2 image, and 4-1-1-3 means Median 5 ×5, Mean 3×3, Mean 3×3, Median 3×3 is used sequentially and produces the best result for DS1 image which is corrupted with speckle noise with variance 0.5. Finally, the resulted images are presented in Figure 3.28, Figure 3.29, and Figure 3.30 respectively.

Table 3.10: PSNR, IQI, and SSIM values after removal of speckle noise with variance 0.5 for DS3, DS2, DS1 images.

Image	**DS3**			**DS2**			**DS1**		
Method	**PSNR**	**IQI**	**SSIM**	**PSNR**	**IQI**	**SSIM**	**PSNR**	**IQI**	**SSIM**
Noisy	15.1164	0.7846	0.7391	13.8978	0.8470	0.7343	13.0944	0.8587	0.7635
Median	20.1819	0.9159	0.6428	18.4929	0.9623	0.6724	16.9667	0.9601	0.6930
Wiener	18.1235	0.9213	0.7095	17.8405	0.9625	0.6549	16.4657	0.9615	0.6733
Lee	21.8919	0.9637	0.9295	21.8459	0.9843	0.9427	19.4450	0.9778	0.9293
Kuan	22.8082	0.9726	0.9403	22.6752	0.9847	0.9503	19.4751	0.9704	0.9271
Frost	22.7319	0.9727	0.9401	23.0627	0.9849	0.9554	19.9862	0.9770	0.9357
Proposed	**24.0001**	**0.9718**	**0.9576**	**23.2782**	**0.9853**	**0.9564**	**20.4271**	**0.9796**	**0.9421**
Sequence	**4-1-1-3**			**5-4-1-3**			**4-3-1-1**		

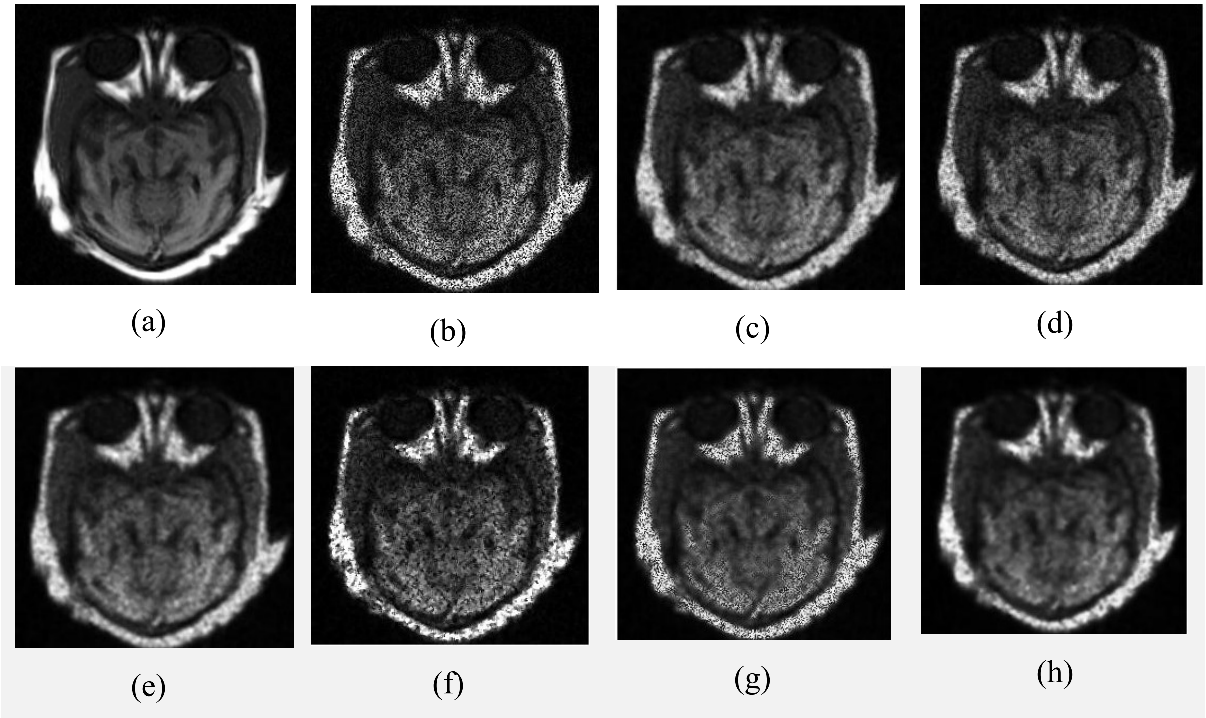


Figure 3.28: Resulting images after Denoising for speckle noise for DS3 image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

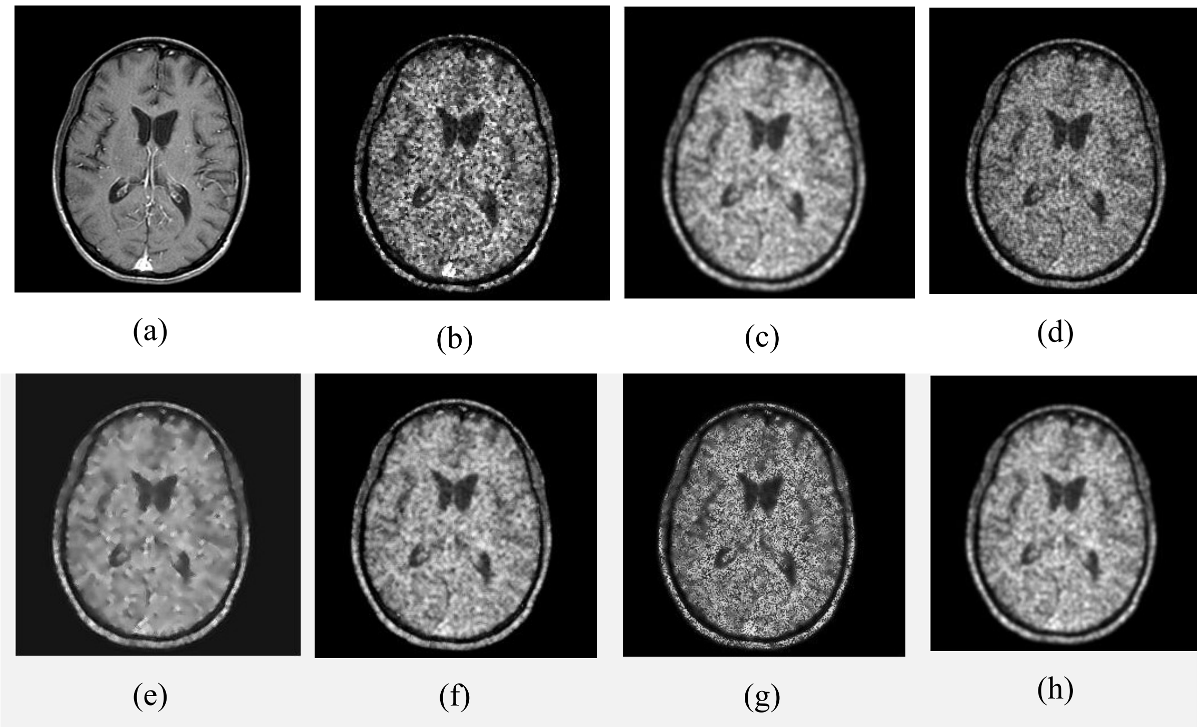


Figure 3.29: Resulting images after Denoising for speckle noise for DS2 image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

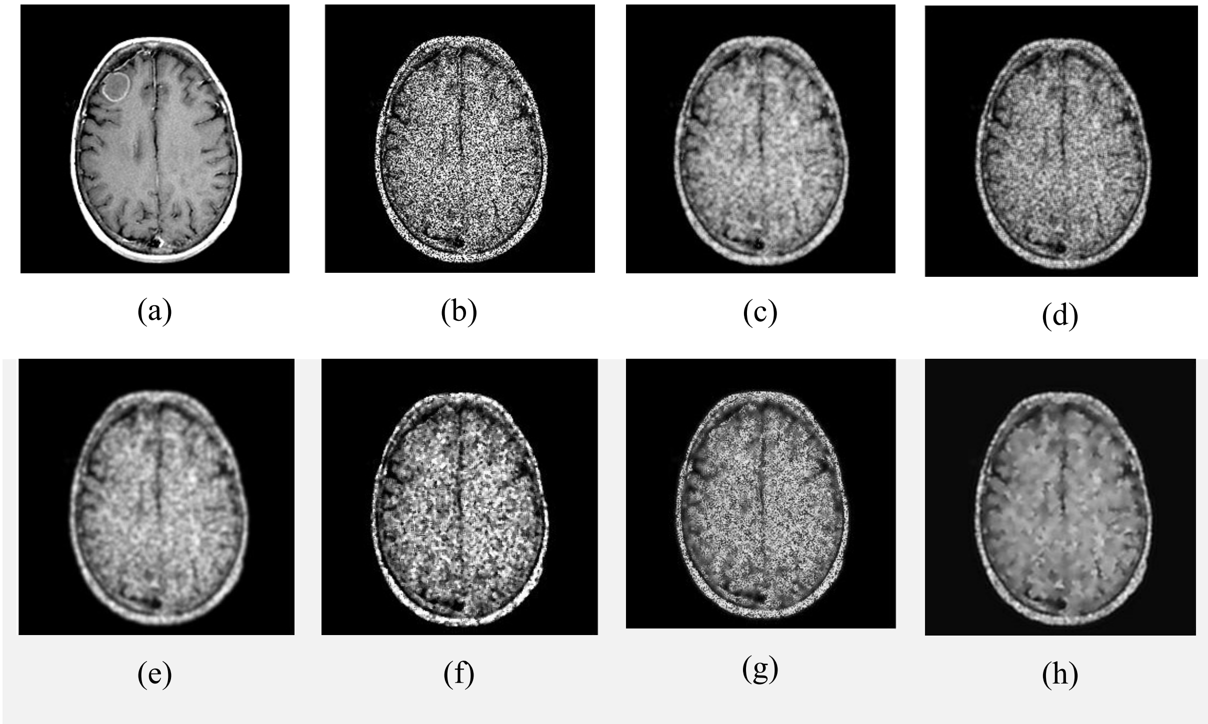


Figure 3.30: Resulting images after Denoising for speckle noise for DS1 image (a) Original, (b) Noisy, (c) Frost, (d) Lee, (e) Kuan, (f) Median, (g) Wiener, (h) Proposed

3.5 Summary

In this chapter total 10 experiments have been performed to verify the effectiveness of the proposed SMA algorithm. To check the performance of proposed algorithm, different standard images are taken with different degradation values and PSNR, SSIM, and IQI parameters are measured. The results obtained from proposed method are compared with standard filtering methods such as Median, Wiener, Frost, Lee, Kuan etc. After performing experiments with 40 trials, it can be concluded that the proposed algorithm provides better results than the compared methods.

CHAPTER 4

CONCLUSION & FUTURE SCOPE

4.1 Conclusion

In this work an improved image denoising technique is proposed with the help of Slime Mould Algorithm. The proposed SMA algorithm is adopted for the optimal design of a cascaded filter in image denoising. This denoising approach is compared to some other well-known filtering techniques. PSNR, IQI and SSIM values are calculated for each filtering method with various types of noises with some degradation levels. From the simulation results discussed in chapter 3, it can be easily seen that the proposed algorithm provides superior results than the other compared filters. The proposed image de-noising technique provides the best values for salt & pepper noise, gaussian noise and also for speckle noise. Hence it can be considered as a best performing method for designing cascaded filter in image de-noising.

4.2 Future Scope

In most of the experiments the proposed approach not only gives better qualitative analysis but also produces outstanding reconstructed visual image. Hence the proposed algorithm may be used as an alternative method for the denoising of images like medical, satellite, and PAN images etc. The application of proposed algorithm for image segmentation, image fusion and image registration can be extended as future work.

The findings of this Thesis work have been communicated for CALCON 2022 as per the details given below:

A. Debnath, S. Dhabal, and P. Venkateswaran, “An Improved Image Denoising Technique by designing cascaded filter Using Slime Mould Algorithm” CALCON 2022, KOLKATA

The following conference (CALCON 2022) has been communicated

2022 IEEE Calcutta Conference (CALCON): Submission (61) has been created.

Inbox

Microsoft CMT **<email@msr-cmt.org>**

Jun 22, 2022, 4:39 PM

to me

Hello,

The following submission has been created.

Track Name: CALCON2022

Paper Title: An Improved Image Denoising Technique by designing cascaded filter Using Slime Mould Algorithm

Abstract:
Abstract— This paper presents an improved image denoising method based on the order of cascaded filters. The amalgamation of different filters is acquired by using the Slime Mould Algorithm (SMA) and cascading four filters among twelve filters which provide improved denoising performance. SMA has a new and individual mathematical model that employs adaptive measures to stimulate the affair of fabricating positive and negative feedback of the propagation wave of Slime Mould and the entire model is supported by a bio-oscillator to build the optical path for bridging food with excellent exploitation propensity and exploratory potentiality. Most of the existing denoising techniques are suitable for either Salt & pepper, Gaussian, or Speckle noise but our proposed algorithm is helpful in removing all the three noises and denoised images are far superior in terms of visual and quantitative analysis. The proposed algorithm also provides a significant improvement compared to other algorithms.

Created on: Sat, 18 Jun 2022 11:32:39 GMT

Thanks,
CMT team.

REFERENCES

[1] J. Zhang, G. Lin, L. Wu, Y. Cheng, "Speckle filtering of medical ultrasonic images using wave-let and Guided filter" ultrasonics, Vol. 65, pp. 177-193, 2016.

[2] V. S. Frost, J. A. Stiles, K. S. Shanmugan, J. C. Holtzman, "A model for radar images and its application to adaptive digital filtering of multiplicative noise" IEEE Trans Pattern Anal Mach Intell, Vol. 4, no. 2, pp. 157-166, 1982.

[3] V. Gupta, C. C. Chan, P. T. Sian, "A Differential evolution approach to pet image denoising" 29th Annual International Conference of IEEE Engineering in Medicine and Biology Society, pp. 4173-4176, 2007.

[4] A. K. Bhandari, A. Kumar, G. K. Singh, V. Soni, "Performance study of evolutionary algorithm for different wavelet filters for satellite image denoising using sub-band adaptive threshold" J Exp Theror Artif Intell, Vol. 28, no. 1-2, pp. 71-95, 2016.

[5] H. Zeng, Y. Z. Liu, Y. M. Fan, X. Tang, "An improved algorithm for impulse noise by median filter" AASRI Procedure, no. 1, pp. 68-73, 2012.

[6] T. D. Pham, "Estimating parameters of optimal average and adaptive wiener filters for image restoration with sequential gaussian simulation" IEEE Signal Process Lett, Vol. 22, no. 11, pp. 1950-1954, 2015.

[7] M. Malik, F. Ahsan, S. Mohsin, "Adaptive image denoising using cuckoo algorithm" Soft Computer, Vol. 20, no. 3, pp. 925-938, 2016.

[8] D. L. Donoho, I. M. Jhonstone, "Adapting to unknown smoothness via wave-let shrinkage" J Am Stat Assoc, Vol. 90, no. 432, pp. 1200-1224, 1995.

[9] M. N. Do, M. Vetterli, "The contourlet transform: an efficient directional multiresolution image representation" IEEE Trans Image Process, Vol. 14, no. 12, pp. 2091-2106, 2005.

[10] J. L. Stark, E. J. Candes, D. L. Donoho, "The curvelet transform for image denoising" IEEE Trans Image Process" Vol. 11, no. 6, pp. 638-684, 2002.

[11] F. Luisier, T. Blu, "Sure-let multichannel image denoising with multichannel image denoising: inter scale orthonormal wavelet thresholding" IEEE Trans Image Process, Vol. 17, no. 4, pp. 482-492, 2008.

[12] H. Y. Yang, X. Y. Wang, P. P. Niu, Y. C. Liu, "Image denoising using non-subsampled shear-let transform and twin support vector machines, Neural Netw, Vol. 57, pp. 157-165, 2014.

[13] U. Erkan, L. Gokrem, S. Enginoglu, "Different applied median filter in salt & pepper noise" Computer Electrical Engineering, Vol. 70, pp. 789-798, 2018.

[14] S. V. Kumar, C. Nagaraju, "Cluster based fuzzy firefly bayes filter for noise identification and removal from grayscale images" Cluster Computing, 2018.

[15] K. N. Chaudhury, S. D. Dabhade, "Fast and Probably accurate bilateral filtering" IEEE Trans Image Processing, Vol. 25, no. 6, pp. 2519-2528, 2016.

[16] K. N. Chaudhury and K. Rithwik, "Image denoising using optimally weighted bilateral filters: A sure and fast approach," 2015 IEEE International Conference on Image Processing (ICIP), pp. 108-112, 2015.

[17] K. He, J. Sun, X. Tang, "Guided image filtering", Computer Vision ECCV, pp. 1-14, 2010.

[18] Z. Youlian, H. Cheng, "Image Denoising algorithm based on PSO optimizing structuring element" 24th Chinese Control and Decision Conference (CCDC), pp. 2404-2408, 2012.

[19] S. Kockanat, N. Karabuga, T. Koza, "Image denoising with 2d fir filter by artificial bee colony algorithm" In international Symposium on innovations in Intelligent system and applications, pp. 1-4, 2012.

[20] C.F.M. Toledo, L. D. Oliveira, R. D. D. Silva, H. Pedrini "Image denoising based on genetic algorithm" IEEE Congress on Evolutionary Computation, pp.1294-1301, 2013.

[21] J. Hua, W. Kuang, Z. Gao, L. Meng, Z. Xu, "Image denoising using 2d fir filters designed with depso" Multimed Tools Appl, Vol. 69, no. 1, pp. 157-169, 2014.

[22] S. Mirjalili, S. M. Mirjalili, A. Hatamlou, "Multi-verse optimizer: a nature inspired algorithm for global optimization", Neural Computation Application, Vol. 27, no. 2, pp. 495-513, 2016.

[23] E. Rashedi, H. Nezamabadi-pour, S. Saryazadi, "GSA: AGravitational Search Algorithm" Vol. 179, pp. 2232-2248, 2009.

[24] R. V. Rao, V. Savsani, D. P. Vakharia, "Teaching-Learning-Based optimization: An optimization method for continuous Non-linear Large-Scale problems" Vol. 183, pp. 1-15, 2012.

[25] S. Mirjalili, "SCA: a sine cosine algorithm for solving optimization problem" Knowledge based System, Vol. 96, pp. 120-133, 2016.

[26] R. Formato, "Central Force Optimization: A new metaheuristic with applications in Applied Electro magnetics" Vol. 77, pp. 425-491, 2007.

[27] D. Wolpart, W. Macready, "No free Lunch Theorems for Optimization" Vol. 1, pp. 67-82, 1997.

[28] D. Straszak, N. Vishnoi "IRLS and Slime Mold: Equivalence and convergence",2016.

[29] M. Premkumar, P. Jangir, R. Sowmya, H. H. Alhelou, A. A. Heidari, and H. Chew, "MOSMA: Multi-objective Slime Mold Algorithm based on Elitist Non-Dominated Sorting", in IEEE Access, Vol.9, pp. 3229-3248, 2021.

[30] L. Abualigah, A. Diabat & M. A. Elaziz "Improved Slime Mold Algorithm by opposition-based learning and Levy Flight Distribution for global optimization & advances in real-world engineering problems." J Ambient Intell Numan Computer, 2021.

[31] M. liu, Y. Li, A. Li, Q. Huo, N. Zhang, N. Qu, M. Zhu, L. Chen "A Slime Mold-Ant Colony Fusion Algorithm for solving travelling salesman problem" in IEEE Access, Vol.8, pp. 2508-2521,2020.

[32] K. Yu, L. Liu, Z. chen, "An Improved Slime Mold Algorithm for Demand Estimation of Urban Water Resources", Mathematics, Vol.9, pp-1-26, No.12, 2021.

[33] Z. Zhang, H. Gao, J, Ma, S. Wang and H. Sun, "Blind Source Separation Based on Quantum Slime Mould Algorithm in Impulse Noise" Vibroengineering PROCEDIA, Vol.7, PP.82-87,2016.

[34] X. Liang, D. Wu, Y. Liu, M. He & L. sun, "An Enhanced Slime Mold Algorithm and Its Application for Digital IIR Filter Design", Discrete Dynamics in Nature and Society, Vol.2021, pp.1-23, 2021.

[35] R. Precup, R. David, R. Roman, E. Petrice, A. SZedlk-Stinean, "Slime Mold Algorithm-based tuning of cost-effective fuzzy controllers for servo systems", Vol.14, pp.1042-1052, 2021.

[36] E. Kropat and S. Meyer-Nieberg, "Slime Mold Inspired Evolving Networks under Uncertainty (SLIMO)," *2014 47th Hawaii International Conference on System Sciences,* pp. 1153-1161, 2014.

[37] T. Schmickland, K. Crailsheim, "A Navigation algorithm for Swarm Robotics inspired by Slime mold aggregation" Vol. 4433, pp. 1-13, 2006.

[38] D. Monismith, B. Mayfield, "Slime mold as a model for numerical optimization" pp. 1-8, 2008.

[39] T. Nakagaki, H. Yamada, T. Ueda, "Interaction between cell shape and Contraction pattern in the physarum plasmodium, Vol. 84, pp. 195-204, 2000.

[40] N. Kamiya, "The Control of Protoplasmic Streaming" Vol. 92, pp. 462-463, 1940.

[41] M. Premkumar, P. Jangir, R. Sowmya, H. H. Alhelou, A. A. Heidari, and H. Chew. "MOSMA: Multi-objective Slime Mold Algorithm based on Elitist Non-Dominated Sorting" in IEEE Access, Vol.9, pp. 3229-3248, 2021.

[42] L. Abualigah, A. Diabat & M. A. Elaziz "Improved Slime Mold Algorithm by oppositionbased learning and Levy Flight Distribution for global optimization & advances in real-world engineering problems." J Ambient Intell Numan Computer, 2021.

[43] M. liu, Y. Li, A. Li, Q. Huo, N. Zhang, N. Qu, M. Zhu, L. Chen "A Slime Mold-Ant Colony Fusion Algorithm for solving travelling salesman problem" in IEEE Access, Vol.8, pp. 202508-202521,2020.

[44] K. Yu, L. Liu, Z. chen, "An Improved Slime Mold Algorithm for Demand Estimation of Urban Water Resources" Mathematics, Vol.9, no.12, 2021.

[45] Z. Zhang, H. Gao, J, Ma, S. Wang and H. Sun, "Blind Source Separation Based on Quantum Slime Mold Algorithm in Impulse Noise" Vibro engineering PROCEDIA, Vol.7, pp.82-87,2016.

[46] D. Straszak, N. Vishnoi "IRLS and Slime Mold: Equivalence and convergence",2016.

[47] X. Liang, D. Wu, Y. Liu, M. He & L. sun, “An Enhanced Slime Mold Algorithm and Its Application for Digital IIR Filter Design”, Discrete Dynamics in Nature and Society, Vol.2021, pp.1-23, 2021.

[48] R. Precup, R. David, R. Roman, E. Petrice, A. SZedlk-Stinean, “Slime Mold Algorithm-based tuning of cost-effective fuzzy controllers for servo systems”, Vol.14, pp.1042-1052, 2021.

[49] E. Kropat and S. Meyer-Nieberg, "Slime Mold Inspired Evolving Networks under Uncertainty (SLIMO)," *2014 47th Hawaii International Conference on System Sciences*, 2014, pp. 1153-1161,

[50] F. Zhang, N. Cai, J. Wu, G. Leu, H. Wang, X. Chen, “Image denoising method based on a deep convolution neutral network” IET Image Process, Vol.12, pp.485-493.

[51] Z. Liu, W. Q. Yan and M. L. Yang, "Image denoising based on a CNN model," 2018 4th International Conference on Control, Automation and Robotics (ICCAR), 2018, pp. 389-393.

[52] J. Guo, Q. Chen, “Image denoising based on nonconvex anisotropic total-variation regularization” Signal Processing, Vol.186, pp.0165-1684, 2021.

[53] M. M. Dhas, N. S. Singh “Blood cell Image Denoising Based on Tunicate Rat Swarm optimization with Median Filter” Vol.116, 2022.

[54] X. Chen, L. Liu, J. Zhang, W. Shao “Infrared Image denoising based on the variation stabilizing transform and the dual-domain filter” Digital Signal Processing, Vol. 113, pp.1051-2004, 2021.

[55] F. Wang, Z. Xu, W. Ni, J. Chen & Z. Pan “An Adaptive learning Image Denoising Algorithm Based on Eigen Value Extraction and GAN Model” Vol.2022, pages. 10, 2022.

[56] S. Kollem, K. R. Reddy, D. S. Rao, C. R. Prasad, V. Malathy, J. Ajayan, D. Muchahary “Image Denoising for Magnetic Resonance Imaging Medical Images using improved generalized cross-validation based on the diffusivity function” pp. 1-23, 2021.

[57] A. Tavakoli, A. pourmohammad “Image Denoising Based on Compressed Sensing” international Journal of Computer Theory and Engineering” Vol.4, 2012.

[58] A. A. Yahya, J. Tan, B. Su, M. Hu, Y. Wang, K. Liu, A. N. Hadi “BM3D Image Denoising Algorithm Based on Adaptive Filtering” Vol.79, pp. 20391-20427, 2020.

[59] C. Wang, M. Li, R. Wang, H. Yu, S. Wang “An Image Denoising Method Based on BP Neutral Network Optimized by Improved Whale Optimization Algorithm” Vol.2021, 2021.

[60] J. V. Manjon, P. Coupe “MRI Denoising using Deep Leaning” Patch Based Technologies in Medical Imaging, pp. 12-19, 2018.

[61] J. Mohan, V. Kishnaveni, Y. Guo, “A Survey on the Magnetic Resonance Image Denoising Methods” Signal Processing, pp. 56-69, 2014.

[62] H. K. Aggarwal, A. Majumder "Mixed Gaussian and Impulse Denoising of Hyperspectral Images." IEEE International Geoscience and Remote Sensing Symposium (IGARSS), pp. 429-432, 2015.

[63] A. S. Ashour, S. Beagum, N. Dey, D. S. Pistolla, G. N. Nguyen, D. N. Le, F. Shi, "Light Microscopy Image Denoising using optimized lpa-ici filter" Neutral Comput Appl 29(12):1517-1533.

[64] T. Blu, F. Luisier, "The sure-let approach to Image Denoising" IEEE Trans Image Process, Vol.16, pp.2778-2786, 2007.

[65] A. Chandra, S. Chattopadhyay "A new strategy of image denoising using multiplier-less fir filter designed with the aid of differential evolution algorithm" Multimed Tools Appl, Vol.75, pp.1079-1098, 2016.

[66] K. N. Chaudhury and K. Rithwik, "Image denoising using optimally weighted bilateral filters: A sure and fast approach," 2015 IEEE International Conference on Image Processing (ICIP), pp. 108-112, 2015.

[67] S. Lahmiri "An Iterative Denoising System based on Wiener filtering with application to biomedical images" Optics Laser Technol, Vol.90, pp.128-132. 2017.

[68] S. Mishra, R. Bisoi, "Image denoising using neural network based accelerated particle swarm algorithm" IEEE Power, Communication and Information Technology Conference (PCITC), pp.901-904.

[69] S. A. Shanthi, C. H. Sulochana, T. Latha "Image denoising in hybrid wavelet and quincunx diamond filter bank domain based on gaussian scale mixture model" Comput Electr Eng, Vol. 46, pp. 384-393, 2015.

[70] C.F.M. Toledo, L. D. Oliveira, R. D. D. Silva, H. Pedrini "Image denoising based on genetic algorithm" IEEE Congress on Evolutionary Computation, pp.1294-1301, 2013.

[71] H. P. Yan and Y. H. Wu, "Filtering image impulse noise by using a PCNN image noise reduction technique", CAAI Transactions on Intelligent Systems, Vol. 12, no. 2, pp. 272-278, 2017.

[72] Y. Ming and S. Li-hua, "The application of an improved fast algorithm of median filter on removing image noise", Engineering of Surveying and Mapping, Vol. 20, no. 3, pp. 65-69, 2011.

[73] S. He, X. L. Pan and Y. M. Li, "Optimization algorithm for average filtering", *Information Technology*, Vol. 3, pp. 41, 2012

[74] P. Milanfar, "A tour of modern image filtering: New insights and methods both practical and theoretical", *IEEE Signal Processing (Magazine)*, Vol. 30, no. 1, pp. 106-128, 2013.

[75] 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 noises," IEE transaction on Pattern analysis and Machine Intelligence, Vol. 4, no. 2, pp.157-166, 1982.

[76] S. Gu, L. Zhang, W. Zuo, and X. Feng, “Weighted nuclear norm minimization with application to image denoising” In proceeding of the IEEE conference on computer Vision & Pattern recognition” Vol. 2014, pp.2862-2869, 2014.

[77] T. D. Pham, “Estimating parameters of optimal average and adaptive wiener filters for image restoration with sequential gaussian simulation” IEEE Signal processing letters, Vol.22, no.11, pp. 1950-1954, 2015.

[78] J. Liu, Y. Wang, K. Su, and W. He “Image denoising with multidirectional shrinkage in directionlet domain,” Signal Processing, Vol. 125, pp. 64-78, 2016.

[79] T. K. Abhiraj and P. Aravindhababu, “Dragonfly Optimization based reconfiguration for voltage profile enhancement in distribution system,” International Journal of Computer Application, Vol. 158, no. 3, pp. 1-4, 2017.

[80] V. Gupta, C. C. Chan, and P.T. Sian, “A differential evolution approach to PET Image denoising” IEEE Engineering in Medicine and Biology Society, Vol. 2007, pp. 4173-4176, 2007.

[81] S. Mirjalili, “Dragonfly Algorithm a new metaheuristic optimization technique for solving single objective, discrete, and multi-objective problem” Neural Computing and Applications, Vol. 27, pp.1053-1073, 2016.

[82] J. L. De, Paiva, C. F. M. Toledo, and H. Pedrini, “An approach based on Hybrid genetic algorithm applied to image de-noising problem,” Applied soft Computing, Vol.46, pp.778-791, 2016.

[83] T. Huang, G. Yang and G. Tang, “A fast two-dimensional median filtering algorithm”, IEEE Transactions on Acoustics Speech and Signal Processing, Vol. 27, no. 1, pp. 13-18, 1979.

[84] D. L. Donoho, “De-noising by Soft-thresholding” IEEE Trans. Inform. Theory, Vol.41, 1995.

[85] B. Ahu & N. I. Cho, “Block-matching convolutional neutral network for image denoising” arxiv preprint arxiv, 2017.

[86] S. Bako, T. Vogels, B. McWilliams, M. Mayer, J. Novak, A. Harvill, P. Sen, T. Derose, & F. Rouselle, “Kernel-predicting convolution networks for denoising monte carlo renderings” ACM Trans Graph.

[87] H. Hassan, A. Saparon, “Still Image denoising based on discrete wavelet transform” In IEEE International conference on system Engineering and Technology, pp. 188-191, 2011.

[88] J. Hua, W. Kuang, Z. Gao, L. Meng, Z. Xu, “Image denoising using 2d fir filters designed with depso” Multimed Tools Appl, Vol. 69, no. 1, pp. 157-169, 2014.

[89] L. Kaur, S. Gupta, R. C. Chauhan, “Image denoising using wavelet thresholding” In Indian Conference on Computer Vision, Graphics and Image processing, 2002.

[90] S. Mirjalili, “SCA: a sine cosine algorithm for solving optimization problem” Knowledge based System, Vol. 96, pp. 120-133, 2016.

[91] S. Kockanat, N. Karabuga, T. Koza, "Image denoising with 2d fir filter by artificial bee colony algorithm" In international Symposium on innovations in intelligent system and applications, pp. 1-4, 2012.

[92] S. Mirjalili, A. Lewis, "The whale optimization algorithm" Advance Engineering Software, Vol. 95, pp. 51-67, 2016.

[93] H. Zeng, Y. Z. Liu, Y. M. Fan, X. Tang, "An improved for impulse noise by median filter" AASRI Procedure, no. 1, pp. 68-73, 2012.

[94] J. Zhang, G. Lin, L. Wu, Y. Cheng, "Speckle filtering of medical ultrasonic images using wave-let and Guided filter" ultrasonics, Vol. 65, pp. 177-193, 2016.

[95] S. Suresh, S. Lal, C. Chen, T. Celik, "Multispectral satellite image denoising via adaptive cuckoo search-based wiener filter" IEEE Transmission Geoscience Remote sensing, Vol. 56, no. 8, pp. 4334-4335, 2018.

[96] Y. Zhou, M. Liu, S. Xu, H. Zang, H. He, Q. Li, J. Guo, "An image denoising algorithm for mixed noise combining nonlocal means filter and sparse representation technique" Vol. 41, pp. 74-86, 2016.

[97] S. Mirjalili, S. M. Mirjalili, A. Lewis, "Grey wolf optimizer" Advance Engineering Software, Vol.69, pp. 46-61, 2014.

[98] T. Blu and F. luisier, "The SURE-LET approach to image denoising" IEEE transaction on image processing, Vol. 16, pp.2778-2786, 2007.

[99] H. Krishna, and C. S. seelamautula, "Sure-fast bilateral filters," IEEE international conference on acoustics, speech and signal processing, pp.1129-1132, 2012.