PRINT QUALITY ASSESSMENT AND DEFECT DETECTION USING COMPUTER VISION

Thesis Submitted by

JAYEETA SAHA

Doctor of Philosophy (Engineering)

Department of Printing Engineering

Faculty Council of Engineering & Technology

Jadavpur University

Kolkata-700032, India

April, 2025

JADAVPUR UNIVERSITY

FACULTY OF ENGINEERING & TECHNOLOGY

KOLKATA-700032, INDIA

INDEX NO. 239/18/E

1. **Title of the Thesis:** **Print Quality Assessment and Defect Detection using Computer Vision**

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

Dr. Shilpi Naskar

Assistant Professor
Department of Printing Engineering
Jadavpur University, Salt Lake Campus,
Plot-8, Sector-III, Block-LB
Bidhan Nagar, Kolkata-700106
West Bengal, India
Phone: +919874710383

3. **List of Publications:**

Papers Published in International Journals

[1] J.Saha, S.Naskar, S.Maiti, "Showthrough and Strikethrough print defect detection using histogram equalization based computer vision method", Journal of graphic engineering and design, 2023. DOI: 10.24867/JGED-2023-2-015. (Scopus Indexed)

[2] J.Saha, S. Naskar, "An approach towards measurement of color shifting in misregistration print defect using euclidean and manhattan distance metrics", International Journal on Recent and Innovation Trends in Computing and Communication,2023.
DOI 10.17762/ijritcc.v11i9.9590. (Scopus Indexed)

[3] J.Saha, S.Naskar, A. Chatterjee and K.C.Paul, "Evaluation of scumming printing defect by using computer vision based bit plane slicing method",Acta graphica: journal of printing science and graphic communications, Vol.33 No. 2,2025. DOI: https://doi.org/10.25027/ag.33.2.4

[4] J.Saha, S.Naskar, A. Chatterjee, “Prediction of Print Contrast and Density using Machine Learning” (Communicated)

Papers Published in International Conference

[1] J.Saha, S.Naskar, A. Chatterjee and K.C.Paul, “Print Scum Identification using DCT based Computer Vision Method”, 2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks(ICRCICN), kolkata, India, 2018, pp.103-107. DOI: 10.1109/ICRCICN.2018.8718734. (IEEE)

[2] J.Saha, S.Naskar, “An approach toward detection of doubling print defect using SSIM algorithm”, Proceedings of International Conference on Data, Electronics and Computing, Algorithms for Intelligent Systems, 2022. DOI: 10.1007/978-99-1509-5_15. (Springer).

4. **List of Patents:** Nil

5. **List of Presentation in National / International Presentations in National/International/Conferences/Workshops:**

[1] Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN),2018, kolkata, India

[2] International Conference on Data, Electronics and Computing, Algorithms for Intelligent Systems, 2022

Statement of Originality

I, **Jayeeta Saha**, registered on the 24th April, 2018 do hereby declare that, this thesis entitled "**Print Quality Assessment and Defect Detection using Computer Vision**". contains literature survey and original research work done by the undersigned candidate as part of Doctoral studies.

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

I also declare that I have checked this thesis as per the "Policy on Anti Plagiarism, Jadavpur University, 2019", and the level of similarity as checked by iThenticate software is **4%**.

Jayeeta Saha

Jayeeta Saha
Index No. 239/18/E
Date: 16/09/25

Shilpi Naskar

Dr. Shilpi Naskar 16/09/25

Assistant Professor
Department of Printing Engineering
Jadavpur University, Salt Lak Campus,
Plot-8, Sector-III, Block-LB
Bidhan Nagar, Kolkata-700106
West Bengal, India

Dr. Shilpi Naskar
Assistant Professor
Department of Printing Engineering
Jadavpur University, Salt Lake Campus
Kolkata-700106

CERTIFICATE FROM THE SUPERVISOR/S

Date:

This is to certify that the thesis entitled ***"Print Quality Assessment and Defect Detection using Computer Vision"*** *submitted by* ***Jayeeta Saha****, who got her name registered on* ***24th April, 2018*** *for the award of Ph.D. (Engineering) degree of Jadavpur University, is absolutely based upon her own work under the supervision of* ***Dr. Shilpi Naskar*** *and that neither her thesis nor any part of the thesis has been submitted for any degree/diploma or any other academic award anywhere before.*

Shilpi Naskar. 16/09/25

Dr. Shilpi Naskar
Assistant Professor
Department of Printing Engineering
Jadavpur University, Salt Lak Campus,
Plot-8, Sector-III, Block-LB
Bidhan Nagar, Kolkata-700106
West Bengal, India

Dr. Shilpi Naskar
Assistant Professor
Department of Printing Engineering
Jadavpur University, Salt Lake Campus
Kolkata-700106

ACKNOWLEDGEMENT

First of all, I take the privilege to pay homage to my supervisor Dr. Shilpi Naskar, Assistant Professor, Department of Printing Engineering, Jadavpur University for her invaluable guidance, help and support. Her undaunted encouragements and unforgettable inspirations have always motivated me towards completion of this thesis, even during the most challenging moments of my life.

I am deeply indebted to Dr. Kanai Chandra Paul, Professor, Department of Printing Engineering, and Dr. Arpitam Chatterjee, Associate Professor, Head of the Department of Printing Engineering, Jadavpur University, for their invaluable advice and guidance. Their constructive suggestions and unwavering support helped me navigate the challenges encountered throughout this research journey. My sincere appreciation also extends to all the faculty members, as well as the technical and non-technical staff of the Department of Printing Engineering, for their constant help during my research.

Finally, my profound gratitude goes to my parents, my late mother, Mrs. Swagata Saha, and my father, Mr. Pradip Kumar Saha, whose unconditional love, sacrifices, and unwavering belief in me have been the foundation of my strength. I am profoundly grateful to my husband, Mr. Debanjan Baksi, whose kind support and encouragement have been a constant source of motivation and reassurance throughout this journey. I would also like to extend my heartfelt thanks to my entire family for their constant inspiration and belief in me, which has guided me through this endeavor.

Jayeeta Saha

Jayeeta Saha

Department of Printing Engineering
Jadavpur University, Salt Lake Campus
Plot-8, Sector-III, Block-LB
Kolkata-700106

April – 2025

Dedicated to my beloved parents,
Late Mrs. Swagata Saha and Mr. Pradip Kumar Saha
with deepest love and gratitude…

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER - 1

Aim and Scope of the Work

CHAPTER-1

Aim and Scope of the Work

1.1 Introduction

Printing technologies have undergone constant evolution over recent decades, driven in large part by the digital revolution, which has introduced a wide range of solutions for image reproduction [1]. With the rapid advancements in printing technology, the quality of printed product has significantly need to be monitored to meet the expectation of customers. In the printing industry, including sectors such as packaging, publishing, pharmacy, textile and commercial printing, print quality is of paramount importance. Defects in printing can significantly affect the appearance, usability, and functionality of the final product, potentially leading to customer dissatisfaction, financial losses, and harm to a brand's reputation. In critical industries like pharmaceuticals or food packaging, print defects can have even more severe consequences, such as incorrect labelling or faulty barcodes, which may compromise compliance, safety, and traceability.

This thesis arises within a context where industries increasingly demand automated approaches for assessing quality of printed product. Printing quality refers to the overall visual and functional excellence of printed materials, determined by various technical and aesthetic factors. It plays a crucial role in meeting customer expectations, ensuring readability, enhancing visual appeal, and achieving the intended purpose of the printed product i.e. functionality of printed materials across various industries, from packaging and publishing to advertising.

Print defects are imperfections or irregularities that occur during the printing process, resulting in deviations from the intended print quality. These defects can manifest in various forms, such as scumming, misregistration, showthrough, color inconsistencies, or missing elements. The causes of these issues can range from mechanical failures, such as equipment malfunctions, to human errors like incorrect settings, poor material handling, or environmental factors. Moreover, Printing quality is influenced by multiple elements, including the characteristics of the printing substrate, ink formulation, printing process parameters, and environmental conditions [2]. The properties of the substrate, such as smoothness, porosity, gloss, and surface energy, play a significant role in determining how ink adheres and spreads to the substrate [3]. Smoother surfaces typically yield sharper images, while porous substrates may absorb ink excessively, reducing color intensity and uniformity. Similarly, ink characteristics, including viscosity, drying time, pigment concentration, and color stability, are essential for achieving print clarity, vibrancy, and smudge resistance. Ensuring proper ink formulation is crucial for achieving compatibility with the substrate and the selected printing process. Additionally, the settings of the printing process, such as speed, pressure, and temperature, critically impact the precision and consistency of the print. Variations in these parameters can result in defects like scumming, misregistration, showthrough, strikethrough, doubling etc. Environmental conditions, including humidity, temperature, and air circulation, further influence print quality [4]. For instance, high humidity can cause the substrate to absorb moisture, altering its interaction with the ink. Lastly, effective design and prepress preparation, such as high-resolution artwork, accurate color calibration, and proper alignment, are fundamental to achieving the intended print quality. Together, these factors determine the overall success and visual appeal of printed materials.

Even with modern printing technologies, checking print quality and finding defects can still be difficult. Current methods often use expensive tools like Spectrophotometer, Densometer Micrometer [5] [6], and rely on skilled operators to inspect prints manually, which takes time and costs a lot. These methods can also vary in accuracy because manual methods depend on human judgment, especially when defects are judged by eye assumption. As the conventional methods need a huge capital investment and also time constraints, they are unsuitable for high-speed, large-scale production. These challenges show the need for better solutions that are easier to use, more reliable, less costly, and suitable for everyday print production. Given these limitations, there is a clear need for a modernized approach to print quality assessment and defect detection. Such an approach should address the shortcomings of conventional methods

by minimizing manual intervention, reducing costs, and delivering consistent and reliable results. Computer vision techniques offer a promising alternative to overcome these challenges. It involves the use of advanced algorithms to analyse and interpret visual data, providing a powerful tool for automated quality control in print production.

This study investigates the potential of computer vision (CV) techniques as a cost-effective and efficient solution for print quality assessment and defect detection. By utilizing advanced algorithms to analyse and interpret visual data, Computer Vison offers a powerful tool for automated quality control in print production. Computer Vision systems can accurately detect and classify common printing defects with minimal human intervention, reducing variability caused by operator judgment. Unlike traditional qualitative methods, it provides objective and measurable evaluations of defects ensuring consistent and repeatable quality control. Additionally, Computer vision systems can analyse printed samples to identify correlations between defects and substrate or ink properties, offering valuable insights for optimizing the printing process. Once implemented, these systems not only can lower operational costs but also significantly enhance processing speed compared to conventional testing methods, making them an efficient and reliable alternative for the printing industry.

1.2 Aim of the Thesis

The primary goals of this research are focused on improving the efficiency and accuracy of print quality assessment and defect detection through the application of computer vision techniques. By automating the conventional process of defect detection, the research aims to reduce human intervention, making the entire process faster and more reliable. This shift towards automation can significantly enhance productivity while minimizing the chances of human error, offering more consistent and repeatable results. Additionally, the research seeks to develop methods that are both cost-effective and time-efficient, without compromising on the quality or precision of the print assessments. By focusing on techniques that can be implemented with minimal resources, the study aims to make these advanced methods accessible to a wider range of printing operations, contributing to the overall optimization of the industry. Furthermore, the integration of a mobile-based approach allows for more flexible and on-the-go monitoring of print quality, making the system easier to deploy in diverse settings and ensuring that quality control can be performed remotely, offering significant advantages in terms of convenience and scalability.

By addressing these objectives, this research seeks to contribute to the advancement of print production quality control. It aims to demonstrate the feasibility and effectiveness of computer vision technologies as a transformative tool for the printing industry, paving the way for enhanced efficiency, reduced costs, and improved print quality in diverse applications. This study represents a significant step toward modernizing traditional practices, ensuring consistency and reliability in print production, and driving innovation in quality assessment methodologies.

1.3 Scope of the Thesis

In order to achieve a problem-free, high-quality printed output, it is essential to control various parameters throughout the printing process. Numerous technical factors play a crucial role in determining the quality of the final product. These include the properties of the ink, the characteristics of the substrate, the precision of press settings, and the management of environmental conditions such as temperature and humidity. Each of these elements must be carefully monitored and adjusted to minimize defects and ensure consistent results. Such careful control not only enhances the visual and functional appeal of the printed material but also ensures efficiency and cost-effectiveness in the printing process. Each printing defects or quality issues have its own unique characteristics, and vary significantly depending on the printing mechanism in use. Even within a single printing process, achieving consistent and high-quality results poses considerable challenges. This is because factors such as the printing substrate, ink properties, and other process parameters change based on the specific job requirements. To solve a specific printing problem effectively, it is important to control the parameters that directly influence that issue. However, the factors affecting one problem may be different from those impacting another, so each issue requires its own tailored approach. Additionally, early and quantitative detection of print defects can accelerate to resolve issues, allowing production to resume more quickly.

Currently, in presses, the detection of print problems heavily depends on human expertise and efficiency. However, manual detection introduces variability and delays, which leads to increasing waste and higher production costs. If print problems could be detected automatically and in real time, corrective measures could be implemented immediately, making the printing process more cost-effective and efficient. This study aims to detect certain print defects and assessment of quality parameters using computer vision (CV) techniques. Computer vision, an

advanced and rapidly growing field, employs image processing and machine learning algorithms to analyse and interpret visual data. It offers promising applications in automating the detection of print defects and the assessment of print quality. Each print issue or quality parameter is unique in its nature and requires a customized approach. So, there are scope of utilizing different image processing operations namely Histogram Equalization, Discrete cosine Transform, Thresholding, Structural similarity index measurement, Automated bit plane selection etc. for detection of different print defects. Whereas for prediction and classification of quality parameters, CNN, DNN, RNN, Logistic Regression, Support Vector Regression, Random Forest etc. can be employed. For this study, the offset printing process has been selected, given its widespread use and versatility in producing printed materials.

There are many print problems and quality parameters to consider, each with its own unique set of challenges. The study focuses on identifying specific print problems such as scumming, show-through, strike-through, misregistration, and doubling, along with evaluating quality parameters like contrast and density. Each of these issues or parameters has distinct characteristics, necessitating different computational techniques for detection or prediction. To address defects, image processing-based detection algorithms are employed. For quality assessment, machine learning methods are utilized to predict quality metrics. This dual approach highlights that, defect detection and quality assessment are interrelated processes. Minimizing defects directly correlates with improved print quality. In this study, computer vision-based image processing techniques are developed and applied to detect each selected print problem effectively. Similarly, the quality parameters are predicted using machine learning models that are specifically designed for this purpose.

By integrating these technologies, the study demonstrates how automation can enhance the efficiency and accuracy of defect detection and quality assessment in offset printing. The findings emphasize that computer vision and machine learning can be tailored to address specific print issues and quality metrics, enabling a significant advancement in the printing industry. Additionally, the ability to consistently produce high-quality prints while reducing costs and minimizing reliance on manual inspection is a key benefit of computer vision-based methods. By automating defect detection and quality control, these technologies enhance efficiency, ensuring consistent output and faster response times to quality issues. This reduces the need for human intervention, lowers operational costs, and improves overall productivity in the printing process. This approach paves the way for a more efficient and reliable printing process, meeting the growing demands for precision and quality in printed materials.

CHAPTER - 2

Review of Previous Works

CHAPTER-2

Review of Previous Works

2.1 Introduction

This chapter compiles and presents the key studies conducted in this field, as documented in the existing literature till date. This survey provides a concise overview of the most relevant and significant studies for convenient reference. A survey has been conducted in the field related to the presented work. The details of the investigation are outlined as follows.

2.2 Review of Previous Investigations

2.2.1 Review on Print Quality

Some interesting and relevant works conducted in the field of print quality are investigated and described.

J.Heilmann *et al*. [7] in (1999) investigated significance of Paper Properties on Print Quality in CIJ Printing. They examined the relationship between a high-speed continuous inkjet system and the printing substrate. Various dense surfaces with distinct topographies were analyzed to assess the effects of droplet impact and spreading without capillary penetration. Additionally, commercial papers were evaluated to determine the performance potential of existing grades and identify key performance parameters. A laboratory-scale testing setup was utilized for high-speed imaging of inkjet droplets, allowing the observation and analysis of their impact,

spreading, absorption, and drying behaviours over time scales ranging from microseconds to several minutes. An advanced image analysis system, specifically designed to evaluate print quality in non-impact printing techniques, was used to assess the technical performance of the samples.

Another investigation in (2009) had been done by Yankai *et al*. [8] on Print Quality Assessment and Identification. This study focused on feature extraction from printed edges and assesses edge roughness using the Local High-Order Correlation (LHOC) method. A specialized magnified image acquisition system was designed to capture edge roughness with clarity. Correlation spectrums of various orders were utilized to quantify the degree of roughness.

K.H Thung *et al*. [9] in (2010) surveyed image quality a comprehensive literature review of existing subjective and objective image quality measures, categorizing them based on their strategies and techniques.

Yanfang *et al.* [10] in (2011) developed a model to quantitatively evaluate the perceived sharpness of black text in digital prints. three key attributes—contrast, edge raggedness, and line width—were identified as indicators of text sharpness. A linear model was established by linking these attributes with subjective evaluations. Results show that contrast and line width positively correlate with perceived sharpness, while edge raggedness has a weaker negative correlation.

Liu *et al*. [11] in (2012) did a study based on measurement of density by controlling printing ink volume. The findings established a functional relationship that can serve as a mathematical model for automatic ink volume control.

Zhang *et al.* (2012) [12] did research to examine the growing demand for high-quality printing and industry standards like ISO, G7, and SWOP. It explored how printing enterprises can meet these standards while enhancing efficiency and profitability using the PressSIGN quality inspection system for intelligent and cost-effective print standardization.

Baar *et al.* (2013) [13] surveyed on image quality assessment for 2.5D and 3D prints, focusing on the challenges of evaluating relief and glossiness in multi-layered prints. While 2D image quality metrics were well-studied, metrics for relief prints remain undefined. This survey explored existing quality assessment methods for 3D prints, objects, simulations, and television to inform future metrics for relief print evaluation.

R. Kumar *et al.* [14] in (2013) proposed a novel image quality assessment method using phase congruency and gradient magnitude to improve accuracy in evaluating distorted images (e.g., white noise, Gaussian blur, JPEG2000 compression). The method outperformed traditional approaches by better aligning with human visual perception. Additionally, an image information measure was introduced to quantify the reference image's information and its extraction from distorted images.

Zhang *et al*. [15] in (2016) researched to enhance quality detection in the plate-making product line, a Simplified Homocentric Square Filter (SHSF) was proposed and implemented using machine vision technology. A hardware platform comprising a camera, lens, light source, and encoder was first designed to capture high-quality images. The SHSF was specifically developed based on the characteristics of common defects and their backgrounds to enable automatic defect detection from the acquired images. Multiscale analysis was employed to identify defects of varying sizes, and by combining SHSF with multiscale analysis, nine types of defects were successfully detected. Additionally, the Integral Image Technique (IIT) was integrated with the filter to enable real-time processing.

Schirmer *et al.* (2018) [16] presented a novel approach to print quality assessment using image processing and statistical methods for objective evaluation. It introduced techniques for measuring line spacing, width, and outline homogeneity, with validation on screen-printed conductors. The study aimed to develop methods suitable for different materials and future inline quality control.

Yi Wang *et al.* [17] researched on the influence of paper surface in digital printing quality in (2019). The quality of printed matter is significantly influenced by paper performance. This study explored the relationship between paper surface properties and digital printing quality using five types of paper. Key factors such as whiteness, gloss, smoothness, Lab* color gamut, solid density, and relative contrast were measured to analyze their impact on printing quality. A mathematical model was developed to establish the relationship between paper performance and digital printing quality, highlighting the influence of these properties on overall printing outcomes.

Wang *et al*. (2021) [18] researched to explore the integration of GATF and G7 printing standardization processes for improved offset quality control. A test version was redesigned for offset magazines, printed using Heidelberg SM74 with and without compensation. The

measured data were analysed to assess the control effectiveness and characteristics of both processes.

Angelov *et al.* (2022) [19] explored automated print quality control using deep learning for image comparison. A convolutional autoencoder, combined with data augmentation and clustering, was proposed to assess print quality by comparing input and printed images, enhancing fault detection beyond traditional histogram and statistical methods.

S. Kumar *et al.* [20] (2022) did a study on print quality factors like hue error, print contrast in piezoelectric inkjet press on matt coated and gloss coated cellulosic substrates.

Wang *et al.* (2024) [21] presented a research work that explores machine learning techniques to optimize infill quality in 3D printing by improving structural integrity and print performance. By analysing key parameters like speed, acceleration, and material use, the study enhanced quality control, predicts mechanical performance, and reduces calibration time, leading to more efficient and sustainable manufacturing.

2.2.2 Review on Defect Detection

Some interesting and relevant works conducted in the field of defect detection are investigated and described.

Townshend *et al*. [22] in (1992) worked on the impact of misregistration on change detection. This study evaluated the impact of image misregistration on detecting land cover changes using spatially degraded Landsat MSS images and simulated NDVI at MODIS resolutions of 250 and 500 m. Experiments showed that even subpixel misregistrations can cause significant errors, with finer spatial frequencies being more affected.

X.Dai *et al.* [23] in (1998) investigated the impact of image misregistration on the accuracy of remotely sensed change detection using Landsat TM imagery. The study evaluated the effects through two components: (1) analysing the statistical properties of multispectral difference images with semi variograms under progressive misregistration to assess sensitivities to band, temporal, and spatial frequencies, and (2) proposing an ellipsoidal change detection technique based on Mahalanobis distance to detect land cover transitions at each misregistration stage.

Mitropulosa *et al.* [24] in (1999) conducted research on real-time pilot system for detecting and classifying defects in web textile fabric. A comprehensive hardware and software platform was

developed to address this issue, incorporating a novel and effective method for defect detection. The proposed method performed well in identifying low-contrast defects under real industrial conditions, where various types of noise were commonly present. For defect classification, an artificial neural network trained with a back-propagation algorithm was employed. By utilizing a reduced number of defect categories, the system achieved consistent, repeatable results with high processing speed.

Kumar *et al.* [25] in (2002) published an article about the use of morphological (MO) methods for detecting defects on textured surfaces by employing neural networks and support vector machines (SVMs). Each pixel in the inspection image was represented by a feature vector, which quantifies the local homogeneity of the texture. These feature vectors, derived from the gray-level patterns of neighbouring pixels, were transformed into eigenspace using Principal Component Analysis (PCA). A classifier was trained using the transformed features from a predefined set of training images. This trained classifier was then applied to categorize every pixel in the inspection image into two classes: MO-class (defect or no-defect).

H.Wang *et al.*[26] in (2005) had done a work on how Positional errors affect the accuracy of image misregistration error in change measurements. Positional errors affect the accuracy of landscape change measurements using high-resolution imagery (~1 m), influenced by error magnitude, landscape heterogeneity, and detection resolution. This study compared maps from 1 m Ikonos imagery of rural China and 0.3 m aerial photos of suburban U.S., finding that errors increase with higher positional errors, greater heterogeneity, and finer resolutions. Regression analysis identifies the minimum detection window size needed for accurate measurements, highlighting the importance of determining the "optimal change detection resolution" for reliable ecological analysis.

Oztana *et al.* [27] in (2005) had done quantitative evaluation of misregistration induced color shifts in color halftones. This study explored how average color changes in two-color halftoned images as misregistration distance varies. Using simulations and print measurements with dot-on-dot/dot-off-dot and rotated dot screen configurations based on a spectral Neugebauer model, the findings revealed that dot-on-dot/dot-off-dot patterns show significant color shifts, while rotated dot screens exhibit minimal shifts under ideal conditions.

Niel *et al.* [28] in (2008) did a study on the impact of sub-pixel misregistration on image differences between Shuttle Radar Topography Mission (SRTM) data and other Digital Elevation Models (DEMs). Misregistration was introduced systematically into SRTM data, and

findings showed that: (1) misregistration caused a strong, erroneous correlation between elevation differences and aspect; (2) the misregistration's direction defined the pattern of these errors; (3) for sub-pixel misregistration, errors equalled or exceeded actual elevation differences in large portions of the landscape and (4) steeper terrain amplified these effects. These results indicate that even sub-pixel misregistration introduces significant biases in DEM comparisons, likely common in any DEM analysis.

Fucheng *et al*. [29] in (2009) researched and proposed a machine vision-based method for detecting printing image defects, supported by an experimental system. The method successfully identified common defects such as filar defects, zonal defects, punctuate defects, and block defects, demonstrating its practicality and potential for broader application.

Yong *et al*. in (2009) [30] investigated how spectral variations in illumination affect print defect classification accuracy across different color spaces when a color camera cannot adjust to changing lighting. Simulated changes in correlated color temperature were applied to test images, and percentile features were used to identify defective areas on printed surfaces. Results showed that the RGB color space performs best under varying illumination, followed by the XYZ and CIELuv spaces. Interestingly, some color space features perform worse than those derived from grey-level images.

Liu *et al.* [31] in (2010) proposed a method to reduce registration noise effects in change detection by processing registered images. Using a pixel-level misregistration map and image gradients, a Thin Plate Spline (TPS) transform estimated misregistration at each pixel with tie points. Compensation includes a spatial correction for registration errors and an intensity adjustment based on subpixel misregistration and local gradients. The method's performance was validated with IKONOS panchromatic and multitemporal multispectral images across three change detection techniques: modified image difference, PCA, and image ratio

In (2010) S.H Indera *et al.* [32] introduced a hybrid defect detection and classification system for bare single-layer PCBs by combining methodologies i.e. morphological image segmentation algorithm and basic image processing theories, the system detected defects from grayscale, single-layer, computer-generated PCB images.

In (2011) Bouchot *et al.* [33] presented a novel algorithm for automatic fault detection in textured images. The method addressed the problem of identifying defects in regularly textured patterns using a template matching approach. Specifically, the algorithm registers patches of the input image to a defect-free reference sample through allowable transformations. A key

innovation was the introduction of the discrepancy norm as a fitness function, which exhibits desirable properties such as monotonicity and Lipschitz continuity. The proposed method used only a few parameters, making it highly adaptable for industrial applications while eliminating the need for complex configuration tuning.

Ding *et al*. [34] in (2012) had worked on feature image mosaicing to reduce misregistration. They proposed a method that reduces misregistration by applying geometric constraints known as trilinearity.

Q.C Hsu *et al.* [35] in (2012) proposed Image-Based Inspection System for Detection of Glass Substrate's Edge Defects. They introduced a new, high-efficiency apparatus for panel-edge inspection. Using image subtraction, binary thresholding, and blob-analysis methods, defects were categorized into three types: fractures, cracks, and dark spots. The proposed system enhanced the efficiency of panel-edge inspections for manufacturers, improving production by over 40% and addressing bottlenecks in the panel industry.

Kaur *et al.* [36] in (2014) presented a method for detecting and classifying defects in printed circuit boards (PCBs) using image subtraction, one of the simplest techniques for PCB defect inspection. The study identified and classified defects such as missing holes, over- and under-etching, incorrect hole sizes, missing conductors, and broken lines.

S. Chauhan *et al.* [37] had done a comparative analysis of print defects in news paper printing in (2016). This research focused on identifying printing defects in national Hindi and English newspapers to address this challenge.

S. Langkam *et al.* [38] in (2017) proposed a dual estimation approach for removing the show-through effect in the scanned documents. This paper proposed a state-space-based method to eliminate this common issue in duplex-printed document scans. The approach defined separate state-space representations for signals and parameters and uses a dual state-parameter estimation method. This involved running two Kalman filters simultaneously: a Kalman state filter for estimating states and a Kalman parameter filter for estimating parameters.

Zhou *et al.* [39] in (2017) introduced an automatic inspection system based on computer vision to detect defect of printing images on Cans based on SSIM and Chromatism. A comprehensive printing detection hardware system was developed, incorporating automatic image acquisition and high-precision image registration. The proposed system featured a geometric defect

detection method based on SSIM and a color defect detection algorithm using the CIEDE2000 color difference formula.

Zhang *et al.* [40] in (2017) proposed a One-Class classifier based on a deep convolutional neural network (CNN) for defect detection. The CNN was trained using a specially designed loss function that incorporates a penalty term based on Euclidean distance. A hypersphere was employed as the classification decision surface, with its radius adjusted according to inspection accuracy. Non-defective product images were mapped within the hypersphere in a high-dimensional feature space, while defect images were mapped far from the hypersphere's center. This approach established a One-Class CNN classifier for effective defect detection.

Ghosh *et al.* [41] (2018) presented a transfer learning approach for classifying PCB images into True Defect and Pseudo Defect categories using a pre-trained Inception-V3 model. Mid-level image representations train an adaptation network, with regularization techniques preventing overfitting. Experimental results on real-world PCB images showed improved classification accuracy of 91% over previous methods.

In (2018) Karthik *et al.* [42] studied on printing defect identification in pharmaceutical blisters using image processing. The objective of this study was to detect printing defects on pharmaceutical tablets from the manufacturing line using advanced image processing techniques. The printed labels on the tablets contain critical information such as chemical composition, date of manufacture, date of expiry, and manufacturing location. Images of these labels were captured and processed using specialized image processing algorithms to identify any defects prior to dispatch. The analysis focused on common printing defects, including missing letters, words, lines, and misalignment issues. By employing the Euclidean distance method for comparison, the system achieved a 95% accuracy rate in identifying and removing tablets with printing defects.

S.Langkam *et al.* [43] in (2018) presented a multiple model Kalman filter method for showthrough cancellation. They proposed a dual estimation approach to address show-through by leveraging both the recto and verso sides of the document. The approach assumed that the degradation arises from the linear mixing of the recto image with the horizontally flipped verso image, and vice versa. A state-space model was formulated to represent show-through and estimate the clear recto and verso images. Multiple Kalman filters were employed to perform state estimation and handle unknown parameters in the model.

Yi Wang *et al.* [44] in (2021) explored algorithms for detecting print defects to improve accuracy and efficiency. It addressed image preprocessing by studying denoising and graying techniques to reduce noise and information loss. A cross-correlation algorithm was used for image registration to correct positional mismatches. Defect detection employs image binarization with the maximum cumulative variance threshold method, morphological processing, and the image difference method. For classification, defects were categorized into point, line, and surface defects using a BP neural network. These methods aimed to enhance printing defect detection and classification performance.

Li *et al.* [45] presented a method in (2022) about deep learning-based method for detecting printing defects, capable of classifying defects into five categories. Experimental results demonstrated that the proposed method achieves accuracy, precision, and recall rates exceeding 96%.

J. Li *et al.* [46] (2023) proposed a deep learning-based method for printing defect detection, addressing class imbalance where normal samples and dirty points dominate other defect types. A warp-based bundled homography model was also introduced to enhance registration accuracy, improving defect detection efficiency.

Liu *et al.* [47] in (2023) introduced an improved YOLOv5 method incorporating a Coordinate Attention mechanism into its feature extraction network to detect five types of printing defects. Experimental results showed the method achieves a 91.7% mAP, a 0.9% improvement over the baseline YOLOv5s, meeting industrial production requirements effectively.

Qi *et al.* (2024) [48] had done research on defect detection in printing barcode area based on machine learning. This paper focused on detecting defects in non-flat printed barcode areas on glass bottles, addressing a gap in current research. The barcode area was divided into two parts: the barcode and the characters. For barcode defects, support vector machines (SVM) were used for feature extraction and classification, while character quality was evaluated using a gray projection method to generate quality scores.

2.2.3 Review on Image Processing Techniques

Some interesting and relevant works conducted in the field of Image Processing Techniques are investigated and described.

Lee *et al.* [49] in (1990) had done a comparative performance study of several global thresholding techniques for segmentation. It was conducted on five global thresholding algorithms for image segmentation using a diverse image database. Histogram distributions were varied by adjusting object size and the mean difference between object and background. Performance was evaluated using criteria such as error probability, shape, and uniformity, including tests on noisy images. Results showed that most algorithms perform well on bimodal histograms but fail when the object-to-background pixel ratio becomes extreme. Performance was found to be data-dependent, with analysis provided for each algorithm based on the evaluation measures.

M. Beauchemin *et al.* [50] proposed an adaptive filter to reduce artifacts caused by image registration in (2005). This technique uses a centre-weighted median filter, where the central pixel's weight varies spatially based on local heterogeneity in the original images. The method's performance is demonstrated using a subset of multitemporal Landsat TM images.

Du *et al.* [51] in (2011) introduced a dynamic threshold edge-preserving smoothing (DTEPS) segmentation algorithm based on histogram analysis for anterior chamber OCT images. The algorithm defined two thresholds: the region of interest discrimination threshold (ROIDT) to identify the region of interest (ROI) and the noise threshold (NT) to detect noise within the ROI. ROIDT and NT were determined using the first-order and second-order differences of the smoothed histogram. To refine segmentation, basic mathematical morphology and median filtering were applied for denoising, while the Kirsch edge detection operator was used to identify the edges of the denoised anterior chamber OCT image. The proposed DTEPS algorithm's performance is evaluated against a global maximum variance threshold (GMVT) segmentation algorithm.

T. Kiran *et al.* [52] in (2011) introduced a novel method for segmenting pulmonary parenchyma using the bit-plane slicing technique. In the proposed approach, the original image was binarized through bit-plane slicing, and the optimal binarized image was selected from the resulting set. After binarization, labeling was performed, and the area of each label was calculated to generate the next level of the binarized image. Subsequently, a boundary tracing algorithm was applied to refine the binarized image. The outlined steps enable the automatic extraction of the lung region.

Patel *et al.* [53] in (2016) studied image registration of satellite images with varying illumination level using hog descriptor-based surf. This study addressed this issue by

combining Histogram of Oriented Gradient (HOG) with Speeded-Up Robust Feature (SURF). Illumination variations could cause incorrect matches with SURF alone, reducing registration accuracy. Using HOG as a descriptor in SURF minimizes these incorrect matches. Simulations with satellite images demonstrate improved correct matching rates with this combined approach.

R. Boda *et al.* [54] in (2016) analysed performance of four segmentation method Active Contour Method (ACM), Gray Threshold Mechanism (GTM), Sobel Gradient Method (SGM), and Watershed Transform (WST) for noisy MRI images. Performance metrics such as Mean Square Error, Peak Signal-to-Noise Ratio, Entropy, and processing time were computed and compared.

Y. Cheng *et al.* [55] did a study on image segmentation technology and its application in digital image processing in (2021). This paper explored image segmentation methods, including thresholding, clustering, and edge detection. It focuses on applying these methods to road image extraction, achieving promising results. Additionally, the study highlighted the applications of image segmentation in areas such as automatic license plate recognition, biomedical engineering, remote sensing, and fire detection.

Jianqing *et al*. [56] in (2022) proposed an image segmentation method for banana leaf disease image with complex background. To address the complexity of banana leaf disease images, a segmentation method combining color segmentation, Otsu segmentation, and area thresholding was proposed.

2.2.4 Review on Discrete Cosine Transform (DCT)

Some interesting and relevant work about application of DCT algorithm are investigated and described.

Sharma *et al.* [57] in (2011) proposed image retrieval technology aims to retrieve images based on DCT color coefficients. This approach used the DCT to retrieve similar images efficiently. It employs the YCbCr color model to decouple color components, transforms them into DCT coefficients, and reduced bit usage by dividing each coefficient by a constant quantization matrix and rounding to the nearest number. These processed coefficients were then used for similarity measurement.

N. Frid *et al.* [58] in (2013) had studied on acceleration of DCT transformation in jpeg image conversion. This paper compared alternative implementations of the DCT transformation algorithm, based on the IEEE 1180 standard, on FPGA development boards like Spartan-3E and Virtex-5 with a 32-bit MicroBlaze™ soft-core processor. Three implementation cases were presented: a software-based AAN algorithm, an FPGA IP-core that accelerates the standard DCT algorithm, and an FPGA IP-core implementing the AAN algorithm.

N. A. Abu *et al.* [59] in (2015) proposed a psychovisual threshold-based approach for image compression, optimizing quantization values using large discrete cosine transform (DCT) image blocks. The psychovisual threshold reflects human visual sensitivity to frequency components, guiding the quantization process to reduce encoded transform coefficients while preserving image quality. By automating the generation of quantization tables, this method significantly improves image quality and reduces artifacts.

X. Fu *et al.* [60] in (2015) proposed a method of image rotation using DCT transformation. In this paper, a novel image rotation method was proposed. This method takes out an n × n block around the point which is the nearest integer point around the unknown point and performs 2-dimensions DCT transformation in the block to obtain the frequency domain information, then interpolate the unknown pixel by performing its 2-dimensions IDCT in the n × n block.

Pang *et al*. [61] in (2019) proposed a method for signal and image compression using quantum discrete cosine transform. This paper introduces a quantum DCT (QDCT) algorithm with significantly lower complexities to efficiently compute and identify significant DCT coefficients, improving performance over Grover's original method. The proposed QDCT-based image compression technique demonstrates superior efficiency, enabling its application in quantum signal and image processing tasks.

2.2.5 Review on Histogram Equalization

Some relevant works about histogram equalization application are investigated and described.

In (2012) S.Muniyappan *et al.* [62] presented a novel approach for image enhancement by using contrast limited adaptive histogram equalization method where Contrast Limited Adaptive Histogram Equalization (CLAHE) method was used to improve image contrast, particularly in medical imaging. They proposed a general framework based on adaptive

histogram equalization and demonstrate its effectiveness compared to other contrast enhancement techniques.

Kong *et al*. [63] in (2013) studied and presented a literature review on histogram equalization and its variations for digital image enhancement. This paper provided an overview of several HE-based methods, which can be broadly categorized into three main groups: Mean Brightness Preserving HE (MBPHE), Bin Modified HE (BMHE), and Local HE (LHE).

K.Santhi *et al.* [64] researched on contrast enhancement of bi-histogram equalization with neighborhood metrics in (2013). Contrast enhancement is vital in image preprocessing, with histogram equalization being one of the most widely used methods. However, traditional global histogram equalization can sometimes lead to excessive visual deterioration. To address this, a novel method called Contrast Enhancement of Bi-Histogram Equalization with Neighborhood Metric (CEBHE) was proposed. CEBHE improves upon traditional bi-histogram equalization by dividing large histogram bins (which cause washout artifacts) into sub-bins using neighborhood metrics that consider neighboring pixel information. The original image's histogram was split into two sub-histograms based on its mean, and each sub-histogram undergoes independent equalization using a refined histogram approach, resulting in a flatter histogram. CEBHE effectively enhances local contrast while preserving image brightness.

Patel *et al.* [65] presented a comparative study on histogram equalization technique for image enhancement in (2013). This paper reviewed HE techniques and evaluates them using metrics such as Absolute Mean Brightness Error (AMBE), Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSI), and Entropy.

Senthilkumaran N *et al.* [66] in (2014) presented histogram equalization for image enhancement using MRI brain images. This study compared Global Histogram Equalization (GHE), Local Histogram Equalization (LHE), Brightness Preserving Dynamic Histogram Equalization (BPDHE), and Adaptive Histogram Equalization (AHE) using various objective quality measures to evaluate their effectiveness in enhancing MRI brain images.

Mustafa *et al.* [67] in (2018) reviewed histogram equalization techniques for image enhancement, which is a fundamental process in image analysis, aiming to improve image quality for specific applications. Numerous contrast enhancement techniques had been developed to enhance image quality while minimizing computational complexity and memory usage. This paper focused on image enhancement methods based on Histogram Equalization (HE), providing a comprehensive review of existing studies. It highlighted the advantages and

limitations of each method and suggests future directions for developing advanced enhancement techniques. This work aimed to guide researchers in designing more effective image enhancement approaches.

2.2.6 Review on Structural Similarity Index Measurement (SSIM)

Some relevant work related to the application of SSIM algorithm are investigated and described.

In (2004) Wang *et al*. [68] worked on traditional objective methods for assessing perceptual image quality focus on quantifying the visibility of errors (differences) between a distorted image and a reference image by leveraging known properties of the human visual system. Assuming that human vision was highly adapted to extract structural information from a scene. They proposed an alternative, complementary approach based on the degradation of structural information. As a concrete example, they introduced the Structural Similarity Index (SSIM) and evaluate its effectiveness through intuitive examples and comparisons with subjective ratings and state-of-the-art objective methods.

A.Horé *et al.* [69] (2010) analysed two widely used objective image quality metrics: Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) . It established a simple mathematical relationship between them, applicable to various image degradations such as Gaussian blur, additive Gaussian white noise, and JPEG/JPEG2000 compression. Tests conducted on images from the Kodak database provide insights into the similarities and differences between PSNR and SSIM.

G Dauphinl *et al.* [70] in (2010) presented a new framework to compare three widely used metrics: Peak Signal to Noise Ratio (PSNR), Visible Differences Predictor (VDP), and Mean Structural Similarity index (SSIM). While VDP and SSIM align with Weber's law, studied using uniform color spaces like CIE-Lab* suggested a different photometric invariance law. This paper established this new law and uses numerical simulations to test whether quality metrics comply with it.

Sara *et al*. [71] in (2019) had done image quality assessment through FSIM,SSIM,MSE and PSNR with a comparative study. Experiments on denoising with benchmark images showed that SSIM and FSIM align with human perception better than MSE and PSNR. Unlike MSE

and PSNR, SSIM and FSIM were normalized and provide semantic, saliency-based insights, making them more intuitive for image quality evaluation.

Ahmed *et al.* [72] in (2019) proposed a method for satellite images features extraction using phase congruency model exploring its advantages and limitations. Additionally, two smoothing algorithms were employed to enhance the feature extraction process.

P. Anandababu *et al.* [73] in (2019) presented a paper on structural similarity measurement with metaheuristic algorithm for content-based image retrieval. This paper introduced an efficient feature extraction method, utilizing a feature set containing color signatures that incorporate both shape and color features extracted from the QI. Similarity measurements were performed using a genetic algorithm with iterated local search, and validation is conducted using the Structural Similarity (SSIM) measure to assess differences between the QI and retrieved images.

Xia *et al.* [74] in (2019) proposed SSIM-NET, a novel two-stage defect detection algorithm. First, they utilized the structural similarity index (SSIM) to identify suspicious regions, making the process more robust to factors like illumination changes and camera noise. Second, they employed the lightweight MobileNet-V3 backbone with a binary focal loss to efficiently classify these regions, reducing computational cost. This approach achieved state-of-the-art accuracy and speed, outperforming Faster-RCNN with at least 12× faster performance and no accuracy loss.

2.2.7 Review on Euclidean and Manhattan Distance Measurement

M. D. Malkauthekar [75] in (2013) did an analysis of euclidean distance and Manhattan Distance measure in face recognition. This study used PCA for feature extraction and compares two distance metrics—Manhattan (L1) and Euclidean (L2)—for measuring similarity in image retrieval. Using facial images of three subjects with varying expressions and angles, experiments reveal that the Manhattan distance outperforms the Euclidean distance in classification accuracy.

2.2.8 Review on Machine Learning

Some interesting and relevant works conducted in the field of machine learning are investigated and described.

Chao-ying *et al.* [76] wrote an article in (2002) about the logistic regression analysis techniques. It outlines essential tables, figures, and assumptions for comprehensive result assessment and demonstrates the preferred application pattern with a data set example.

Maher Maalouf [77] in (2009) had given an overview on logistic regression in data analysis. This paper provided an overview of key aspects of LR from an algorithmic and machine learning perspective, emphasizing its application to imbalanced and rare event data in data analysis.

Xia *et al.* [78] in (2017) presented flower classification using the Inception-V3 model with transfer learning on the TensorFlow platform. By overcoming challenges like background complexity and species similarity, the approach significantly improves classification accuracy compared to traditional feature-based methods with 94 and 95% accuracy.

Mittal *et al.* (2018) [79] did the study by employing the Inception-V3 model to break image-based CAPTCHAs, achieving over 91% accuracy on a real Facebook CAPTCHA dataset. The results highlight vulnerabilities in modern CAPTCHA systems, emphasizing the need for more robust human verification methods.

Chen *et al.* [80] in (2019) proposed a CNN-based classifier leveraging image bit-plane slicing to enhance recognition accuracy for breast cancer image classification. Each texture image is decomposed into eight bit-plane images, with each bit-plane capturing different levels of detail and texture features. They evaluate the classification performance of individual bit-planes as well as the fusion of all bit-planes. The CNN classifier is employed for both recognition and classification tasks. Simulation results on breast cancer image datasets demonstrate that the proposed method significantly improves recognition accuracy and enhances classification performance on specific bit-planes.

Li *et al.* (2019) [81] presented a transfer learning approach using the Inception-V3 model for classifying fresh tea leaves. Experiments on a self-created dataset achieved over 95% accuracy, improving to 98% with data augmentation. The method enhanced efficiency and reliability for industrial tea production.

Hsieh *et al.* (2020) [82] proposed an AI-based system to detect and classify breast microcalcification (MC) clusters in mammograms, aiming to reduce invasive biopsies. Using VGG16, Mask R-CNN, and Inception V3, our method achieves 93% accuracy in cluster detection, 95% in MC labeling, and 91% in malignancy classification. This system effectively supports diagnosis and reduces patient burden.

J. Bae *et al.* [83] in (2020) did a study which presents a feature extraction model based on inception v3 model to distinguish normal heart sound from systolic murmur. The model classified normal and abnormal heart sounds, focusing on detecting systolic murmurs. Heart sound data from an electronic stethoscope was converted into mel-spectrogram images, which were used as input for a fine-tuned Inception V3 model. Convolutional layers serve as feature extractors, with effective features selected using point-binary correlation analysis and crystal coefficient values. An artificial neural network classifier was employed, achieving an average accuracy of 87.7%. With 5-fold validation and the top 30 features, accuracy improves to 97.5%, offering valuable support for physicians in identifying systolic murmurs.

Yuan *et al.* [84] in (2020) proposed CNN classifier based on salient features of mammography and bit-plane decomposition to enhance the recognition accuracy of breast cancer images. The proposed approach utilizes bit-plane decomposition, where each texture image is divided into eight bit-plane images, with each plane representing distinct levels of image features. Additionally, 25 feature subgraphs are extracted from the original image, capturing various characteristics of the image features. By combining these two salient feature sets, the method significantly improves recognition accuracy. The integration of features from different bit-plane images has been thoroughly tested. Simulation results on medical image datasets demonstrate that the proposed method increases recognition accuracy from 63.5% to 91.1% for certain bit-planes, achieving a remarkable improvement of 27.6%.

Yin *et al.* [85] in (2020) proposed deep guidance network that incorporates a guided image filter module to recover structural information using a guidance image. The method supports end-to-end training and achieves fast inference for biomedical image segmentation.

Nabil *et al.* (2021) [86] presented a face shape classification approach using the Inception-V3 model, optimized through preprocessing techniques like straightening, cropping, and normalization. Trained on various datasets of female images, the model achieved a 94.3% accuracy

Saeed *et al*. (2021) [87] did a study where they utilized an Inception-V3-based deep learning model with an added dense layer and image augmentation to detect COVID-19, pneumonia, lung opacity, and normal cases from chest X-ray scans. The model achieved a high accuracy of 99.72%, enabling rapid and reliable COVID-19 diagnosis.

Thangaraj *et al*. [88] in (2021) had worked on deep Learning based Real-Time Face Detection and Gender Classification using OpenCV and Inception v3. This study addressed object identification and recognition through CCTV cameras, proposing a motion-detection-based recording system to reduce data storage costs. The method was suitable for monitoring spaces like homes, offices, and gardens to detect unusual activity. For face and gender recognition, the system uses Haar cascades for face detection, cropping the detected faces for gender classification with the Inception V3 model. Trained and tested on the IMDB dataset, the model achieved 97.4% accuracy in real-time gender classification despite challenges like lighting, poses, and expressions.

M. Balipa *et al.* [89] in (2022) detected Alstonia Tree using CNN and Inception V3 Algorithms. As the identification of alstonia tree was challenging due to the diversity of tree species, varying orientations, viewpoints, and background clutter. Accurate tree identification was essential for gathering species-specific information. Using the pretrained Inception V3 model and Convolutional Neural Networks (CNNs) enables effective detection and classification of trees, offering higher accuracy compared to traditional methods.

S. Likhitha *et al.* [90] in (2022) did a study on skin cancer segmentation using R-CNN comparing with Inception V3 for better accuracy. This study compared the performance of R-CNN (Region-based Convolutional Neural Network) and Inception V3 in segmenting skin cancer images. Using a sample size of 20 for each group with a pretest g power of 80%, R-CNN achieved an accuracy of 96.01% and specificity of 86.76, while Inception V3 showed 92.00% accuracy and 89.47 specificity.

G. Priyadharshini *et al.* [91] in (2023) did a study focused on detecting and classifying tomato leaf diseases using methods like CNN, R-CNN, Fast R-CNN, and Faster R-CNN Leaf diseases, such as Mosaic virus, Early Blight, Septoria leaf, Bacterial spot, and healthy leaves, significantly affect crop yield and financial outcomes in agriculture. Early detection is crucial to minimize losses. The proposed approach used deep learning and image processing techniques, employing VGG 16 for feature extraction and classification. Faster R-CNN integrates a regression-based bounding box method for precise disease identification.

Iqbal *et al.* [92] in (2023) presented a paper about COVID-19 classification using CNN, Inception V3 and Transfer Learning. This study proposed a deep learning approach using Convolutional Neural Networks (CNN) and the Inception V3 model, known for extracting detailed features from medical imaging. The CNN-Inception V3 model was evaluated on a comprehensive dataset of chest X-ray and CT images to differentiate COVID-19 from other respiratory conditions like pneumonia, fibrosis, tuberculosis, and normal lung scans. Performance metrics include accuracy, sensitivity, specificity, and AUC-ROC.

Joshi *et al.* [93] in (2023) did a study on multiclassification on using YOLOv5 and Inception V3 model to recognize tea leaves. Tea leaf diseases significantly affect global tea harvest quality and productivity, making early and accurate diagnosis crucial for effective control. This study employed YOLOv5 and Inception V3 models to identify tea leaf conditions. The YOLOv5 model achieved a highest mAP of 0.78 (threshold 0.5), while the Inception V3 model excelled with a 94.3% F1-score in classifying red rust leaves.

Kalpana *et al.* [94] (2023) explored deep learning models for automated cotton leaf disease detection using semantic image segmentation. Convolutional neural networks, including VGG16 and Inception-V3, were applied to classify diseased and healthy leaves, achieving over 96% accuracy.

R. Chavan *et al.* [95] in (2023) had worked on classification of retinal fundus images using vgg16 and inception v3. This study aimed to develop a framework for classifying retinal fundus images and detecting multiple diseases affecting the eye and other organs. Using retinal fundus images and transfer learning models, early disease detection is achievable.

Susanti *et al.* [96] in (2023) did a classification of beef images using the Inception V3, Transfer Learning model. This study used machine learning with the Inception V3 transfer learning architecture to classify beef as fresh, half-fresh, or spoiled.

Verma *et al.* (2023) [97] compares Inception-V3 and VGG-19 models for traffic sign detection using a dataset of 44,028 images. The Inception-V3 model achieved a higher accuracy of 94.38% compared to 90.12% for VGG-19. Results indicated that Inception-V3 was more effective for traffic sign detection.

Kankariya *et al.* (2024) [98] did a study to explore advancements in Sign Language Recognition (SLR) using deep learning, specifically CNN and Inception-V3 models. The

proposed method achieved 98.94% accuracy in recognizing hand gestures, improving communication for the hearing and speech impaired.

Kaur *et al.* (2024) [99] presented a fine-tuned Inception-V3 model for the multiclass classification of groundnut diseases, including Early Leaf Spot, Late Leaf Spot, Rust, and Groundnut Rosette Disease. The model, trained on a comprehensive dataset of groundnut leaf images, achieves high accuracy of 96%, demonstrating the potential of deep learning for efficient and precise agricultural disease diagnosis.

Rani *et al*. (2024) [100] proposed an Inception-V3-based CNN model for malware detection, achieving 90.6% accuracy and 92.6% specificity. By leveraging data preprocessing and visualization techniques, the model enhanced early malware strain identification, improving detection precision and effectiveness.

2.3 Discussion

The motivation behind this research work comes from investigations of previous works where the noticeable lack of work with real life printing problem is identified. In real life, different print problem can occur in a single printed material during production. Though many works have been done with different printing defects but form the literature review it can be seen that most of them are manmade print problems on already printed sheets. There is a significant gap in research that directly tackles the practical print problems encountered in real-world printing. This study seeks to bridge this gap by examining actual offset printing problems, with a cost-effective, automated identification process. By focusing on real-life scenarios, this research aims to contribute valuable insights that can enhance both academic understanding and industrial practices in the field of printing technology.

CHAPTER - 3

Identification and Evaluation of Scumming Printing Defect

CHAPTER-3

Identification and Evaluation of Scumming Printing Defect

3.1 Introduction

Offset lithography [2] is a planographic printing process, meaning the printing plate have image and non-image area on same plane, is a widely used printing technique. The distinction between image and non-image areas is achieved through differences in the plate's surface chemistry, which creates a differential in wetting. The image areas are designed to attract grease (ink receptive) and repel water, while the non-image areas attract water (water receptive) [2]. Both water and oil-based ink are applied to the plate during the process. The water adheres to the non-image areas, preventing the ink from sticking to them. Non-image areas, which are hydrophilic (water-attracting), readily absorb water, while the image areas, being hydrophobic (water-repelling), reject water. The ink adheres only to the hydrophobic image areas and is then transferred to a rubber-coated cylinder. This rubber blanket cylinder subsequently transfers the image to the substrate under high pressure. Unlike other printing methods, lithography uniquely uses two fluids (water and ink) to differentiate between image and non-image areas and employs an intermediate transfer via the rubber blanket rather than directly printing onto the substrate. Figure 3.1 [2] shows a schematic diagram of offset lithography printing process.

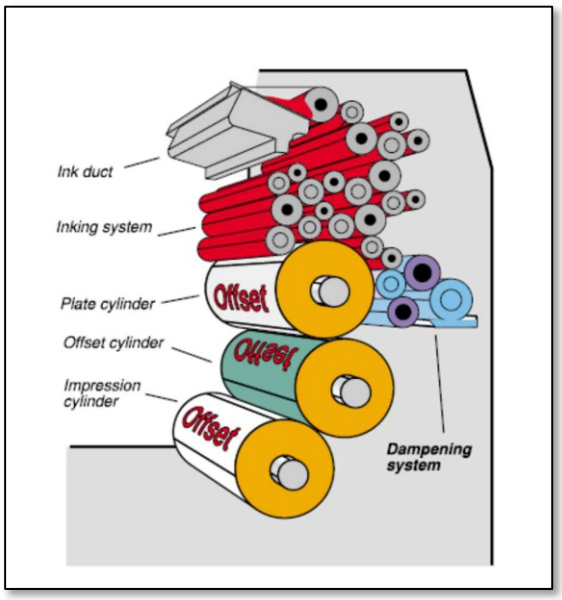

Figure 3.1 Schematic Diagram of Offset Lithography Printing Process

Scumming [3] is a printing defect that occurs in offset lithography when non-image areas of the printing plate, which should repel ink, instead attract it and transfer it to the blanket. Blanket then transfers the ink to the printed surface. This results in unwanted ink marks or patches appearing on the printed substrate. Figure 3.2 depict a real-life sample of scumming printing defects.

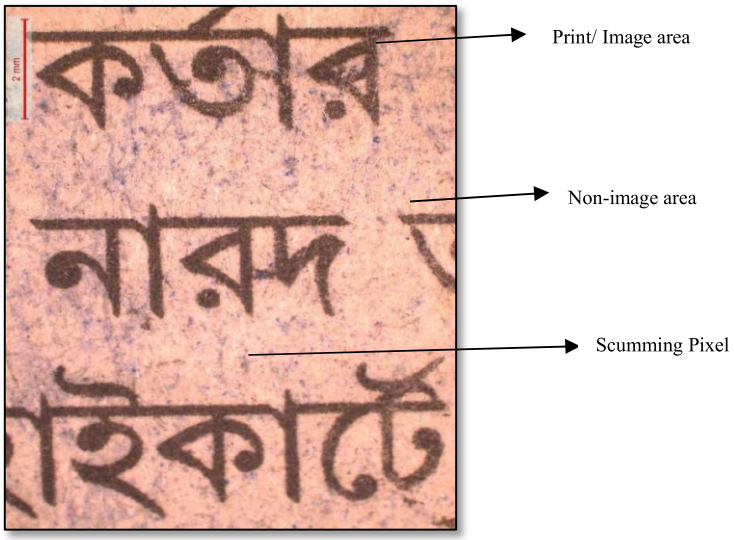

Figure 3.2 Sample Image Showing Image, Non-image Area and Scumming Printing Defect

This phenomenon undermines print quality, increases production costs, and affects operational efficiency. Despite advances in printing technology, scumming remains a persistent issue. The presence of scumming significantly affects the visual quality of printed materials, reducing readability and making text or images less clear. Additionally, during the archival process of documents, scumming can interfere with scanning and Optical Character Recognition (OCR) systems, as the unwanted ink marks may be misinterpreted as characters or distort the intended content, leading to inaccuracies in digital records.

Scumming, a common printing issue, can occur on any color unit during production due to unfavourable press conditions. One major cause is improper ink-water balance [3]. The dampening solution plays a crucial role in offset printing by maintaining the hydrophilic (water-attracting) properties of the non-image areas of the printing plate, preventing ink from adhering to these regions. When there is insufficient dampening water, the non-image areas lose their hydrophilic characteristics, making them prone to ink adhesion [2] [3]. This imbalance allows ink to spread onto unintended areas of the plate, leading to scumming and negatively impacting print quality. Moreover, improper mixing of damping solution water, where the pH is either too acidic or too alkaline. This affects the surface tension of the water, causing non-image areas on the plate to accept ink. Maintaining the damping solution within a pH range of 5.5 to 5.8 [101] and ensuring proper conductivity can mitigate this issue. If the pH is high (pH>7) in the dampening solution it results emulsification between ink and water [4]. Also, improper dampening levels can lead to over-emulsification of the ink, which is another contributing factor to the scumming defect in printing. Over-emulsification occurs when excessive dampening solution mixes with the ink, altering its consistency and stability [2]. This imbalance can cause the ink to lose its adhesion properties and spread uncontrollably onto non-image areas, further increasing the likelihood of scumming and compromising the overall print quality. Excessively low ink viscosity, often caused by high press temperatures, is a common cause of offset printing. Typically, ink exhibits lower viscosity [3] within the 30–40°C operating temperature range of a web press. However, if the temperature rises—often due to mechanical shear—the ink's viscosity decreases further. This can lead to unintended transfer onto non-image areas, resulting in scumming. Maintaining proper ink viscosity and balancing ink and water effectively are essential to minimize such issues. Monitoring press conditions and implementing adequate cooling mechanisms can help prevent excessive temperature rise, preserving print quality. When dampening rollers are contaminated with dried ink, they fail to

properly emulsify the water and ink mixture. This improper emulsification can lead to uneven distribution of the dampening solution on the printing plate. As a result, the printing plate may experience increased friction and wear, particularly in areas where the ink has dried and hardened on the rollers which may cause the scumming. In scumming, the density of halftone areas increases due to the unintended transfer of ink onto the non-image areas of the printing plate. This increase in ink coverage affects the tonal rendition [3] of the printed image, causing the reproduction of tones to deviate from the intended output. The excessive ink density can obscure finer details, particularly in halftone shadow areas, leading to a loss of subtle gradations and shadow details [2]. This degradation in tonal quality impacts the overall visual appeal and accuracy of the print, making the desired output image less detailed than designed. Scumming in offset printing can also result from poor plate development [3]. Poor plate development, often caused by inadequate processing or improper use of chemicals, can leave unprocessed residues on the plate surface. These residues disrupt the hydrophilic nature of the non-image areas, making them susceptible to ink adhesion. Incorrect plate exposure [3], such as underexposure or overexposure during the imaging process, affects the plate's ability to distinguish between image and non-image areas effectively. This compromise can lead to improper ink transfer and increase the likelihood of scumming. If the ink lacks sufficient tackiness and becomes too soft, it may unintentionally transfer over the fountain solution film in the non-image areas of the printing plate, which contributes to scumming. The tackiness of the ink is crucial, as it helps the ink to adhere primarily to the image areas, while the non-image areas—coated with the dampening solution—repel the ink. When the ink is overly soft, it becomes excessively fluid and loses its resistance to spreading [3]. As a result, it can migrate onto the non-image areas, leading to ink deposition where it is not intended. These causes of scumming, which negatively impacts the clarity and overall quality of the print. To avoid this, it is essential to use ink with the right tack and consistency for the specific printing conditions. Regular monitoring of ink properties and adjusting press settings accordingly can help prevent scumming and maintain print quality [2] [3].

The only remedy for preventing scumming caused by overly soft ink is to increase the stiffness of the ink film. This can be achieved by adding gum or water-resistant varnish to the ink [3]. These additives help maintain the ink's tackiness, preventing it from spreading to the non-image areas of the plate. Additionally, if the ink vehicle contains too much fatty acid, it can destroy the desensitizing film on the non-image areas of the plate. This film is critical for repelling ink from the non-image areas, and its degradation can lead to ink transferring to these regions,

resulting in scumming. Therefore, it is essential to maintain a proper balance in the ink formulation to prevent excessive fatty acids and ensure the integrity of the desensitizing film. To prevent scumming, the surface tension of the fount must be reduced to a similar value to the surface tension of ink, approximately 37×10^{-3} $N \cdot m^{-1}$ [3]. By aligning the surface tension of both the ink and the fountain solution, the ink is less likely to transfer to the non-image areas of the plate. Achieving this balance, reduces the risk of scumming and helps maintain the clarity and quality of the print. Properly managing the surface tension is therefore a critical factor in preventing ink transfer to unintended areas. Ensuring plate cleanliness is essential in preventing scumming and maintaining print quality. Regular cleaning of plates helps remove any residual ink, debris, or chemical buildup that could interfere with the proper ink-water balance. By cleaning the plates thoroughly and consistently, it is ensured that the non-image areas remain properly protected and free from ink, allowing the ink to adhere only to the image areas. This practice is crucial for achieving high-quality, consistent prints. Adjusting the composition of the damping solution water to maintain the correct pH and conductivity is crucial. This prevents the erosion of non-image areas on the plate, thereby reducing scumming [101]. In the context of direct plate usage, passing the plate through an activator solution [102] before mounting it on the press can prevent scumming in the image area and blinding in the background area. However, the press life of the direct plate is limited by the wear of the aluminium from the background areas, which can cause scumming over time [102].

This type of printing defect is unacceptable and must be eliminated to prevent interference with archival processes such as scanning and optical character recognition (OCR). In the printing industry, identifying and addressing such defects is critical to ensure high-quality production. Currently, scumming detection in printing presses is predominantly performed manually, resulting in variability due to individual differences in the ability to identify and quantify the defect. Moreover, there is no standardized method for detecting scumming without human intervention. This study introduces a computer vision-based approach employing automatic bit-plane selection to detect scumming defects. The proposed method aims to reduce reliance on manual inspection, improve consistency, and streamline the detection process, making it faster and more efficient. For the purpose of this detection, real press samples are collected from various commercial printing presses. The proposed technique is applied directly to printed images to identify these defects. This method represents a novel approach in the field of printing, introducing an automated defect detection system that significantly reduces human

errors. By leveraging this innovative system, the process becomes more reliable, efficient, and consistent, setting a new standard for identifying printing defects such as scumming.

3.2 Sample Preparation

Samples for the study are collected from various commercial offset printing presses. To digitize these samples, a mobile camera with a 48-megapixel resolution is used under D65 illumination conditions, ensuring cost effective imaging system. The camera setup used for this experiment is illustrated in Figure 3.3. The collected samples are placed in this custom-built image acquisition system, where lights are provided uniformly to mitigate the variation in illumination. Considering future real-time applications, such as mobile apps and handheld instruments, images are captured in JPEG format—a lossy but widely compatible format used across various devices, including mobile phone cameras. An adjustable base is incorporated to modify the distance between the camera lens and the object under study, ensuring optimal focus.

Once digitized, the samples are categorized into three distinct groups based on their tonal range and brightness levels: high-key images (Figure 3.10 (a) & Figure 3.12 (a)), mid-key images (Figure 3.14 (a) & Figure 3.16(a)), and low-key images (Figure 3.18 (a) & Figure 3.20 (a)). These groupings are determined by analysing the histograms of the images, which provided a visual representation of tonal distribution. high key image, characterized by its dominance of lighter tones and minimal shadows. mid key image, featuring a balanced mix of mid-tones, highlights, and shadows. low key image, distinguished by its emphasis on darker tones and shadows [103]. This systematic classification based on image histograms allowed for a structured analysis of scumming defects across various tonal ranges, ensuring that the detection method could be evaluated comprehensively across different types of printed outputs.

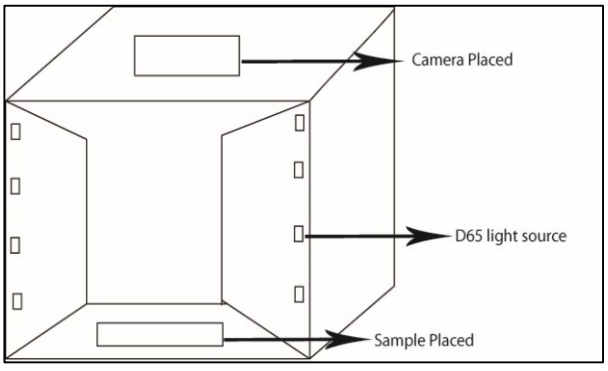

Figure 3.3 Schematic Diagram of the Camera Set-Up

3.3 Problem Statement

In the presented study, a computer vision-based approach has been proposed to detect these print defects such as scumming and quantify their percentage, offering an alternative to traditional manual detection methods. Unlike manual detection, which is subjective and prone to variability, the automated approach aims to provide a more consistent and efficient solution.

One of the key challenges in this study is distinguishing scum pixels from print pixels, as their density differences are often minimal and not easily distinguishable. This subtle difference poses a significant hurdle in accurately identifying scum pixels in the computer vision-based detection process. Furthermore, the intensity of pixels can vary across different types of images, such as high key, mid key, or low key images. This variability complicates the segregation of scum pixels from print pixels, as the intensity ranges may overlap depending on the image's tonal category.

To address these issues, the study proposes using specific intensity ranges for segmenting print pixels and scum pixels, tailored to the sample's category—high key, mid key, or low key. This category-specific segmentation strategy leverages the unique tonal characteristics of each image type to improve the accuracy of defect detection. For instance, high key images, dominated by lighter tones, are assigned different intensity thresholds compared to low key images, which emphasize darker tones and shadows.

This approach significantly enhances segmentation accuracy, yielding better results by minimizing misclassification of images. Without these tailored intensity ranges, there are chances of print pixels being mistakenly identified as scum pixels, particularly in mid key and low key images. By addressing these challenges, the proposed method demonstrates its potential to overcome the limitations of manual detection and offer a reliable solution for identifying and quantifying scumming defects in printed materials.

3.4 Proposed Method

The proposed method has been rigorously tested using a variety of samples collected from offset commercial presses. All the samples used in the study are real-life printed materials, ensuring that the evaluation reflects practical printing conditions and challenges. MATLAB programming software is employed to analyse the samples and detect scumming defects systematically and efficiently.

Figure 3.4 illustrates the flowchart of the proposed method, providing a step-by-step overview of the process. Below is an explanation of each step in the methodology:

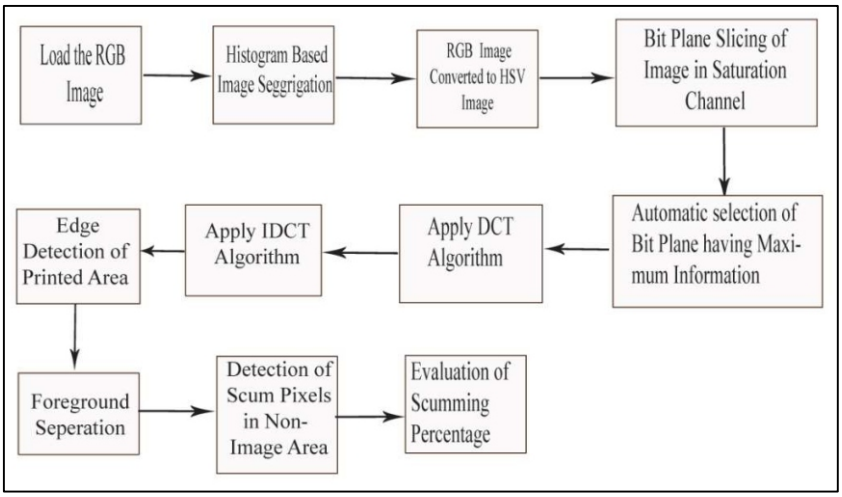

Figure.3.4 Flow Chart of Proposed Method of Scumming Printing Defect Detection

3.4.1 Image Segregation

The first step involves capturing high-resolution digital images of the printed samples using a camera setup shown in Figure 3.3. The imaging process ensures consistency in illumination and resolution for accurate analysis. After digitizing the printed image samples, they are categorized into three groups based on their histograms: high-key images (Figure 3.10 (a) & Figure 3.12 (a)), mid-key images (Figure 3.14(a) & Figure 3.16(a)), and low-key images (Figure 3.18 (a) & Figure 3.20 (a)). The histogram analysis provides a graphical representation of the tonal distribution in each image, enabling a systematic classification:

- **High-Key Images**: These images are characterized by a dominance of lighter tones with minimal shadows, resulting in a bright appearance with high intensities [103].
- **Mid-Key Images**: These images feature a balanced mix of highlights, mid-tones, and shadows, offering a moderate contrast that is neither overly bright nor excessively dark.
- **Low-Key Images**: These images emphasize darker tones and shadows, creating a dramatic and high-contrast effect with minimal highlights and low intensities [103].

This categorization is crucial for tailoring the defect detection process, as the tonal characteristics of each group influence the intensity thresholds used for separating print pixels

from scum pixels. By categorizing the images based on their histograms, the method ensures more accurate segmentation and detection of scumming defects across a variety of printed outputs.

3.4.2 Image Conversion

After segregating the images based on their histogram analysis, the RGB images are converted to the HSV color space. Unlike the RGB color space [103], HSV (Hue, Saturation, Value) is channel-independent, meaning the representation of color information remains consistent across devices. This eliminates the variability associated with device-specific RGB interpretations, ensuring more reliable results. Additionally, the HSV color space aligns better with human visual perception, making it more suitable for tasks like image processing and analysis.

In the HSV model, color information is represented in a way that is more intuitive to human vision. It consists of three components:

- Hue (H): Represented as an angle in a circular coordinate system, the hue defines the type or purity of a color, such as pure red, blue, or yellow. The hue value is normalized to a range of 0 to 255 for computational convenience. [103]
- Saturation (S): This component indicates the intensity or vividness of a color, describing how colorful or washed out an area appears in relation to its brightness. Saturation values also range from 0 to 255, with 0 representing a shade of gray (no color) and 255 indicating maximum color intensity.
- Value (V): Also known as brightness, this represents the luminance or color sensation in terms of intensity. The value channel reflects the gray level of the image and ranges from 0 (black) to 255 (full brightness).

For the purpose of this study, the saturation channel is identified as the most informative component for further analysis. The saturation channel provides critical insights into the intensity and vividness of colors, making it particularly useful for image processing tasks like segmentation, smoothing, sharpening, [103] and identifying printing defects. Unlike the hue channel, which primarily deals with color type, and the value channel, which can lose relevance at brightness extremes, the saturation channel captures meaningful variations in color intensity that are vital for detecting defects such as scumming. In Figure 3.5 different channel of

scumming printing defect sample image is shown in HSV color space, where it is clearly shown that Saturation channel image (Figure 3.5(b)) is more appropriate for further operations.

By leveraging the saturation channel, the study is able to proceed with operations that required detailed color intensity information, enhancing the precision and effectiveness of the defect detection methodology. This decision underscores the suitability of HSV over RGB for tasks that demand perceptually meaningful color representation and robust analysis capabilities.

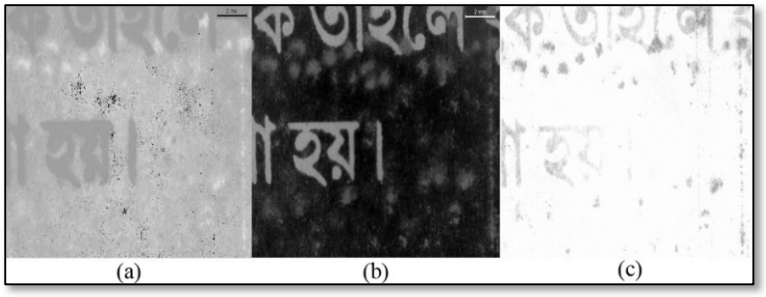

Figure 3.5 Scumming Printing Defect Sample Image in (a) Hue Channel (b) Saturation Channel (c) Value Channel

3.4.3 Automated Bit Plane Selection

After selecting the saturation channel, the next step involved slicing it into different bit planes using the bit-plane slicing technique [80] and selection of the most informative plane automatically. This step is essential for extracting finer details and features from the image by analysing each bit plane individually. Manual selection of bit plane may introduce subjectivity and variability, whereas automation ensures a consistent and objective choice based on predefined criteria. Moreover, automation helps in identifying the most informative plane without manual trial and error. The automated approach is time saving and it also minimizes the risk of human error.

For an 8-bit image, the bit-plane slicing technique allows the image to be divided into eight separate bit planes, ranging from the 0th bit plane (which holds the least significant bit) to the 7th bit plane (which holds the most significant bit) [103]. Each bit plane contains different levels of detail, with the lower bit planes primarily capturing fine details and noise, and the higher bit planes containing the more significant, coarse features of the image.

In the case of scumming print defects, scum pixels can appear both in the foreground and in the background of the image. Therefore, it becomes important to segment the image into bit planes to isolate the raw information at each level. This segmentation allows for a more detailed analysis of the image, as different bit planes emphasize different aspects of the image's structure. It is commonly identified that lower bit planes, which capture finer details, primarily contain foreground pixels, while the upper bit planes, which represent the more significant portions of the image, generally contain background pixels.

To enhance the accuracy of scumming detection, an automated approach is developed to select the bit plane that carries the most useful information related to scumming pixels. This selection process is based on the analysis of the axes properties of the image histogram. Specifically, the histogram of each bit plane is analysed, where the y-axis represents the pixel intensity values, and the x-axis corresponds to the bit planes. The bit plane that contained the highest concentration of higher intensity pixels i.e the higher proportion of active bits ('1's) within a bit plane (those associated with scumming defects) is chosen for further processing. This method allows for the identification of the bit plane most relevant to the detection of scumming, ensuring that the most significant features of the defect are captured.

By selecting the optimal bit plane irrelevant information like noise from other bit planes can be minimized, thus improves the accuracy and efficiency of the defect detection process. This step in the workflow ensures that the analysis focuses on the most important features related to scumming, leading to more effective detection and quantification of the defect.

Bit-plane slicing offers several advantages in digital image processing. It enhances analysis by enabling the study of an image at different intensity levels, allowing finer details and structures to be examined independently. This technique also facilitates a focused approach by isolating and processing the most significant or relevant bit planes, which are critical for specific tasks, such as defect detection or feature extraction [103]. Additionally, by concentrating on individual bit planes, the overall computational complexity is reduced, making it a more efficient method for handling large datasets or performing targeted image manipulations [103]. These benefits make bit-plane slicing a powerful tool for diverse applications in image processing and analysis. Figure 3.6 illustrates the different bit-planes of a sample image, showcasing the decomposition of the image into its binary components.

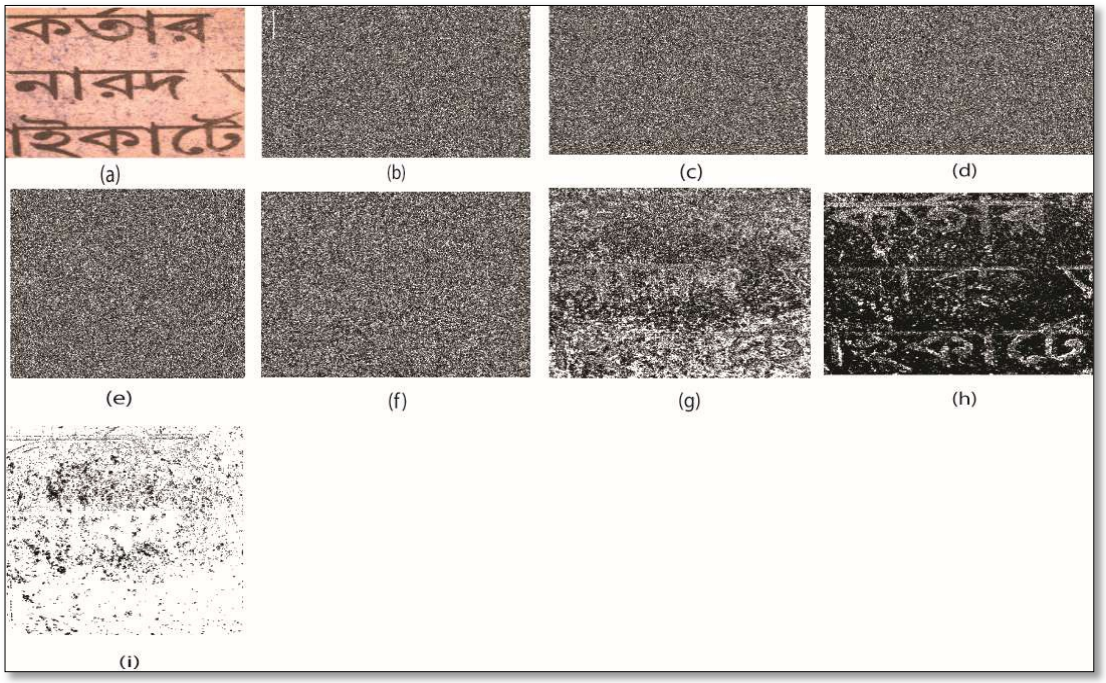

Figure.3.6 The Eight Binary Images after Applying Bit- Plane Slicing Technique (a) Sample Image; (b) Plane Zero (c) Plane One (d) Plane Two (e) Plane Three (f) Plane Four (g) Plane Five (h) Plane Six (i) Plane Seven

3.4.4 Discrete Cosine Transform

The Discrete Cosine Transform (DCT) [58] [59] [60] [61][104] is a widely used mathematical tool in signal and image processing for representing data in terms of frequency components. It transforms spatial domain data into the frequency domain, enabling the analysis of how much energy is concentrated in different frequency ranges. DCT is particularly popular in applications like image compression, such as JPEG, and video compression, like MPEG.

The Discrete Cosine Transform (DCT) is highly efficient in concentrating the majority of a signal's energy into a few low-frequency components. This means that the essential information of the signal is represented compactly, while the higher-frequency components, which often correspond to finer, less noticeable details, can be discarded with minimal impact on quality. This property makes DCT particularly effective for applications such as image compression [103], where data reduction without significant perceptual loss is crucial.

Unlike the Discrete Fourier Transform (DFT), which produces complex values due to its sinusoidal representation, the DCT operates solely with real numbers. This characteristic

simplifies computations and reduces the complexity of processing, making the DCT more efficient and practical for many real-world applications, including digital multimedia processing and feature extraction tasks. 2D-DCT can be expressed as [59] [104]

$$F(u,v) = \left(\frac{2}{N}\right)^{1/2} \left(\frac{2}{M}\right)^{1/2} \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} \alpha(i).\alpha(j).cos\left[\frac{(2i+1)u\pi}{2N}\right] cos\left[\frac{(2j+1)v\pi}{2M}\right].f(i,j)$$

eq. 3.1

where,

The input image is N by M.

f (i, j) = intensity of the pixel in row i and column j.

F(u, v) = DCT coefficient of row and column of the DCT matrix.

After selecting the appropriate bit plane, the Discrete Cosine Transform (DCT) algorithm is applied to filter out unwanted frequency components while retaining the essential ones. This step is crucial for distinguishing the significant low-frequency components, which are generally located near the origin in the frequency domain, from higher-frequency components that often represent noise or irrelevant details. In the context of scumming defect detection, the low-frequency components are more consistent in density and are primarily associated with the intended text or image regions of the print. In this work, 'unwanted frequency' refer to the high frequency DCT Coeffiecients beyond the cumulative energy threshold, which mainly represent noise and scumming and are suppressed using a low-pass mask to retain essential print sturctures. Figure 3.7(b) shows the resulted image after DCT conversion.

The purpose of this frequency reduction is to refine the identification of pixels belonging to the actual print (text or image) by discarding unwanted scum pixels. Once the frequency-domain filtering is completed, the Inverse Discrete Cosine Transform (IDCT) (Figure 3.7(c)) is applied to reconstruct the image from its frequency representation. This combination of DCT and IDCT ensures a clean and accurate reconstruction of the print, enhancing defect detection and maintaining the integrity of the printed content.

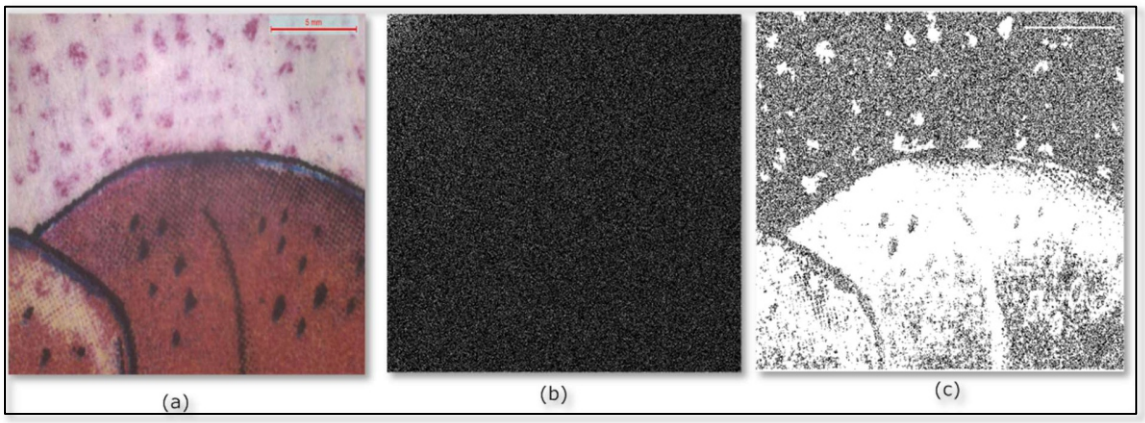

Figure.3.7 (a) Original Image (b)Image after DCT Conversion (C) Image after IDCT Conversion

3.4.5 Detection of Scumming Pixel

After applying the Discrete Cosine Transform (DCT) and reconstructing the image, the contour of the print pixels is identified using the *Prewitt operator*, a gradient-based method for edge detection. This process is essential for distinguishing the boundaries of the print pixels from other elements in the image. The segmentation procedure is carried out in two distinct steps to achieve accurate separation [105]:

I. **Segmentation of Print Pixels from Non-Image Areas:**

In the first step, an adaptive thresholding algorithm [103] is employed to segment the image into two primary regions: the foreground and the background. Figure 3.8(a) shows the thresholded image. The foreground region is expected to contain the print pixels, such as text or images shown in Figure 3.8(b), while the background region included scum pixels and non-image areas shown in Figure 3.8(c). This thresholding method dynamically adjusted the segmentation criteria based on the intensity distribution of the pixels, ensuring better adaptability to varying image conditions.

By intensity distribution it is meant that the histogram based satistical spread of pixel values in the reconstructed image, from which the algorithm adaptively selects a threshlod to seperate foreground print pixel from non image areas.

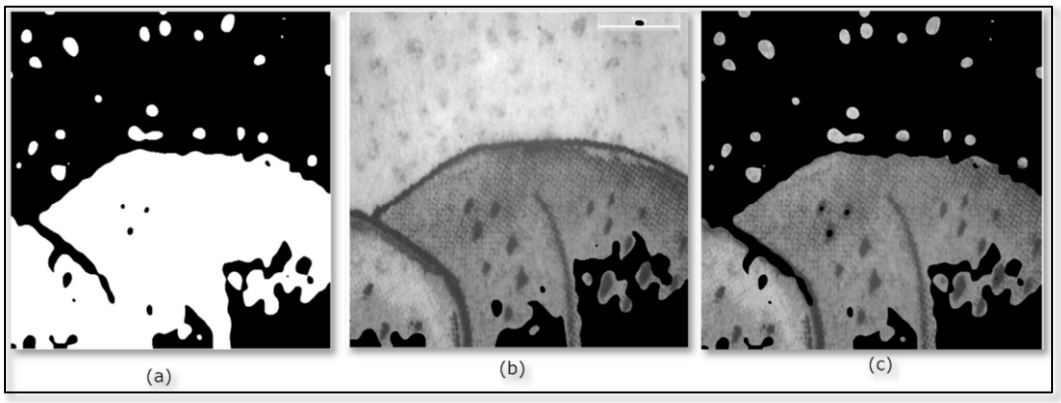

Figure.3.8 (a) Thresholded Image (b)Foreground Image (C) Background Image

II. **Segmentation of Scum Pixels from Non-Image Areas:**

In the second step, segmentation focused specifically on isolating scum pixels from the non-image areas. To achieve this, a range of intensity values is determined by several trials for each image type—high-key, mid-key, and low-key images—based on their unique intensity distributions. These ranges are finalized by testing several samples of each image type to ensure consistent and accurate results.

- **High-Key Images:** For high-key images (Figure 3.10), which are dominated by lighter tones and minimal shadows, pixels with intensity values within the range of 0.65–0.85 are identified as foreground or print pixels. Pixels with intensity values below 0.65 are classified as background or scum pixels.

- **Mid-Key and Low-Key Images:** Mid-key images (Figure 3.14), characterized by a balanced mix of tones, and low-key images (Figure 3.17), defined by darker tones and pronounced shadows, required slightly different intensity ranges due to their inherent tonal differences. For these image types, pixels with intensity values in the range of 0.45–0.75 are identified as print pixels, while pixels with intensity values below 0.45 are classified as background or scum pixels.

This segmentation process relied on the careful selection of intensity ranges for each image category, ensuring that scum pixels are accurately differentiated from non-image areas without compromising the identification of print pixels. The adaptive nature of the thresholding and the specific intensity ranges tailored to image types provided a robust framework for detecting

scumming defects in a variety of printed samples. This multi-step segmentation technique enhanced the precision of defect detection, contributing to better quality control in the printing process.

3.4.6 Percentage of Scumming

After identifying the scum pixels, the percentage of scumming is determined by comparing the number of scum pixels to the total number of pixels in the original image. The process involved the following steps:

- **Calculation of Total Pixels:** The total number of pixels in the original image is calculated. This value represents the complete dataset, including both print pixels and non-image areas.
- **Detection of Scum Pixels:** From the scum pixel-detected image, the total number of pixels classified as scum pixels is determined. These are the unwanted pixels identified in the non-image areas of the print.
- **Percentage Calculation:** The percentage of scumming is computed by comparing the total number of scum pixels to the total number of print pixels.

This approach provided a quantitative measure of scumming in the printed samples, allowing for an objective assessment of the defect's extent. By systematically calculating the percentage of scum pixels, the method ensured consistency and accuracy in defect evaluation, aiding in quality control processes.

3.5 Result and Discussion

The proposed method is tested on a variety of samples obtained from different offset printing presses, with their digital versions captured using a mobile camera with 48 mega pixels. The pictorial results for three distinct sample types: high-key (Figure 3.10 and 3.12), mid-key (Figure 3.14 and 3.16), and low-key (Figure 3.18 and 3.20) images are presented.

3.5.1 For High-key Image:

In Figure 3.10(a), the original high-key image of a sample is presented. The original image is categorized as high-key image by analysing the histogram of image showing in Figure 3.9. By analysing the bit-plane slicing of this image, it is observed that among the seven-bit planes, bit plane-6 contained the majority of the scum pixels. This critical observation formed the basis for subsequent image processing steps, as further operations to identify and isolate scum pixels while retaining the print pixels are focused exclusively on this bit plane. This bit plane is selected automatically for further operations.

The first step involved segmentation to isolate the foreground image. Following the initial segmentation process, the foreground image area is successfully segmented, as illustrated in Figure 3.10(b). This step enabled the separation of the primary printed content from the background, paving the way for a more targeted analysis of scum pixels in the non-image regions.

The second segmentation process is carried out to analyse the non-image areas of the sample. For high-key images, a specific intensity range of 0.65–0.85 is chosen for this purpose after several trial and error. This intensity range is selected based on its effectiveness in distinguishing scum pixels from other non-image background regions. After this second level of segmentation, the scum pixels in the non-image area are successfully identified. These pixels are visually represented in white within the non-image region in Figure 3.10(c), where the contours of the printed text and images are also delineated, showing the scum pixels present in the background.

Finally, the percentage occurrence of scum pixels is calculated, providing a quantitative measure of scumming for the analysed sample. This data is summarized in Table 3.1, where the scum percentage for the high-key image in Figure. 3.10 (a) is reported as 42%. This metric offers valuable insight into the extent of scum occurrence, facilitating comparisons with other sample images and assisting in the evaluation of print quality.

Another high-key sample image is presented in Figure 3.12 (a) where similar image processing techniques are applied to analyse scum pixel occurrence. The histogram of the original image is shown in Figure 3.11 which shows that the image can be categorized into high-key sample group. An initial assessment of the bit-plane information revealed that, as in the earlier example, bit plane-6 contained the majority of the scum pixels for this high-key sample. This

consistent observation across high-key images underscores the importance of bit plane-6 in isolating scum pixels for such samples.

The analysis began with the segregation of the foreground and background regions of the image. Through the first level of segmentation, the foreground area, which includes the printed text and image content, is successfully separated from the non-image background. This step is crucial for ensuring that subsequent processing focuses only on the relevant regions where scum pixels may be present.

Following this, the second level of segmentation is performed to identify scum pixels within the non-image background area. This involved analysing pixel intensity values within a defined range, similar to the previous case. The intensity range considered for identifying scum pixels in this high-key image is again optimized to ensure accurate detection of these artifacts without interfering with the foreground print pixels. After processing, the scum pixels in the non-image background are identified and are clearly shown in Figure 3.12(c). In this figure, the scum pixels are represented in white against the background, with the contours of the printed elements also visible to demonstrate the separation.

Finally, the percentage of scum pixels in the non-image background is calculated for this high-key sample. The calculated scum percentage for the image presented in Figure 3.12 (a) is 16% (Table 3.1).

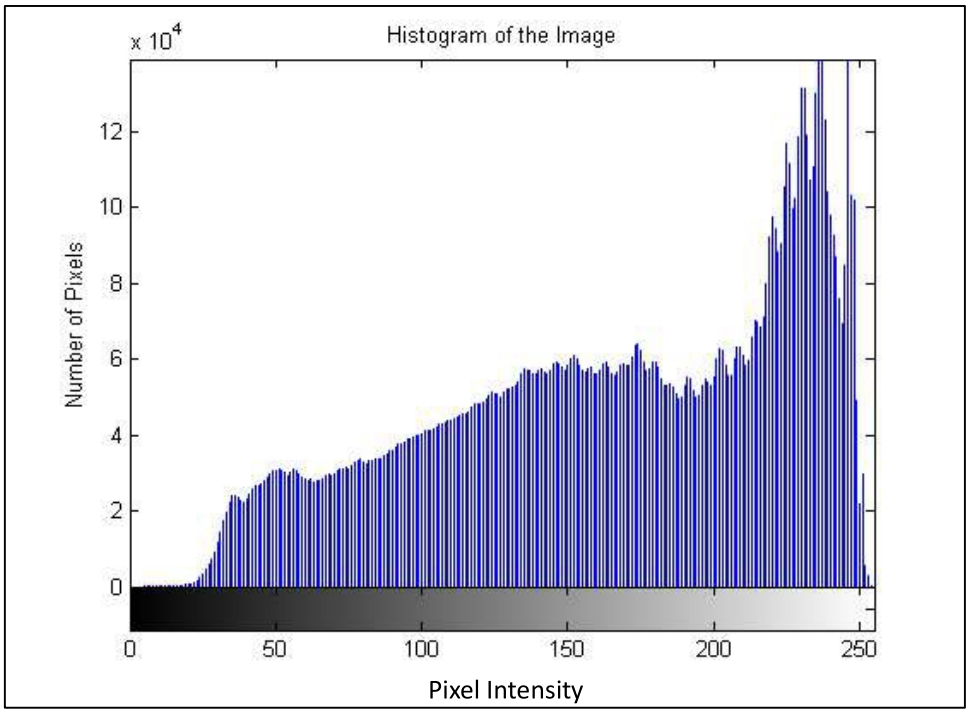

Figure 3.9 Histogram of Original Image (Sample 1) (Figure 3.10 a)

Figure.3.10 Resulted Image of Scumming (Sample 1) (a) Original Image (High Key) (b) Foreground Image (c) Detected Scum Pixels

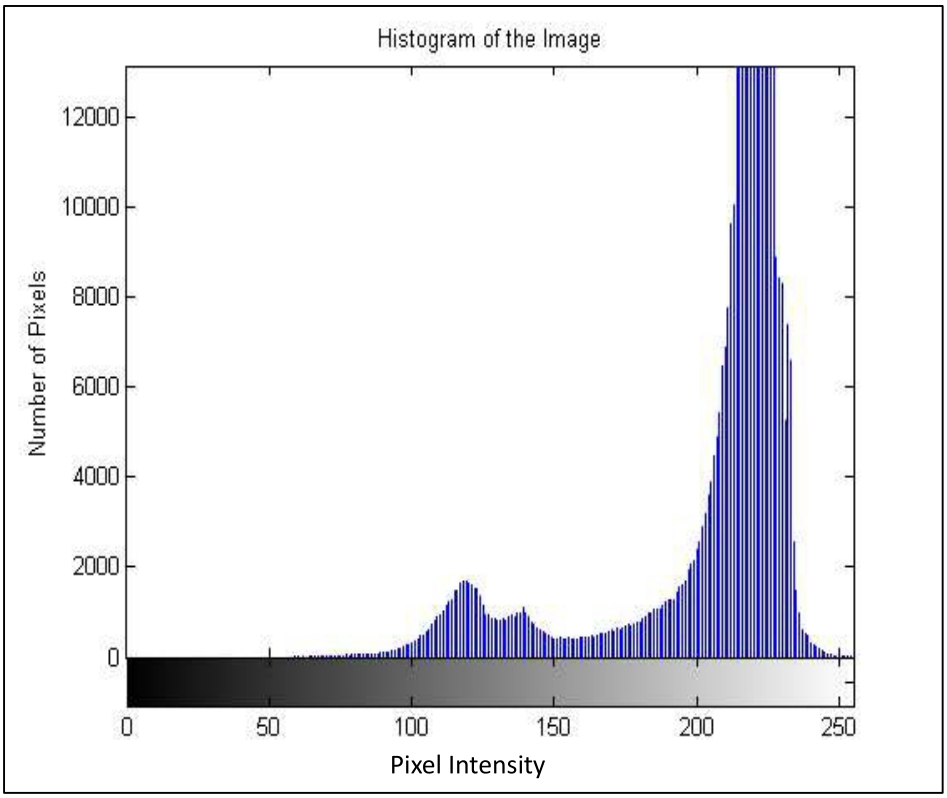

Figure 3.11 Histogram of Original Image (Sample 2) (Figure 3.12 a)

Figure.3.12 Resulted Image of Scumming (Sample 2) (a) Original Image (High Key) (b) Foreground Image (c) Detected Scum Pixels

3.5.2 For Mid-key Image:

In Figure 3.14(a), a mid-key sample image is presented. Histogram of this sample image is depicted in Figure 3.13. A detailed bit-plane analysis of this sample revealed that, bit plane-7 contained the highest concentration of scum pixels for this sample. This significant finding informed the decision to focus all subsequent image processing operations exclusively on bit plane-7 to effectively detect and isolate the scum pixels in this mid-key image. This particular bit plane is selected by an automated approach for further processing.

The same steps for segmenting the foreground and background regions of the image and detecting scum pixels are followed in accordance with the proposed methodology. The result of this segmentation process is displayed in Figure 3.14(b), which clearly highlights the printed image as the foreground region.

Following the foreground segmentation, the second stage of analysis concentrated on the non-image background area to identify and isolate scum pixels. For this mid-key sample, a specific intensity range of 0.45–0.75 is utilized after several trial and error during the second-level segmentation process. This range is carefully selected to enhance the accuracy of scum detection by differentiating scum pixels from other background elements. After applying this segmentation, the scum pixels in the non-image background are identified and are depicted in Figure 3.14(c). In this figure, the scum pixels are represented in white, contrasting against the background, while the contours of the printed image remain distinct to demonstrate the successful separation of print and scum areas.

Finally, the percentage of scum pixels is calculated for this mid-key sample to quantify the extent of scumming in the image. The scum percentage for this sample is determined to be 14% (Table 3.1). These findings provide valuable insights into the behaviour of scumming in mid-key images and validate the segmentation methodology applied to this analysis.

Figure. 3.16(a) presents another mid-key sample image and its histogram in Figure 3.15, which is primarily image-based. The same steps of segmentation, including the separation of foreground and background regions and the detection of scum pixels, are performed using the recommended intensity range. For this sample, the scum pixel percentage is calculated to be 40% (Table 3.1).

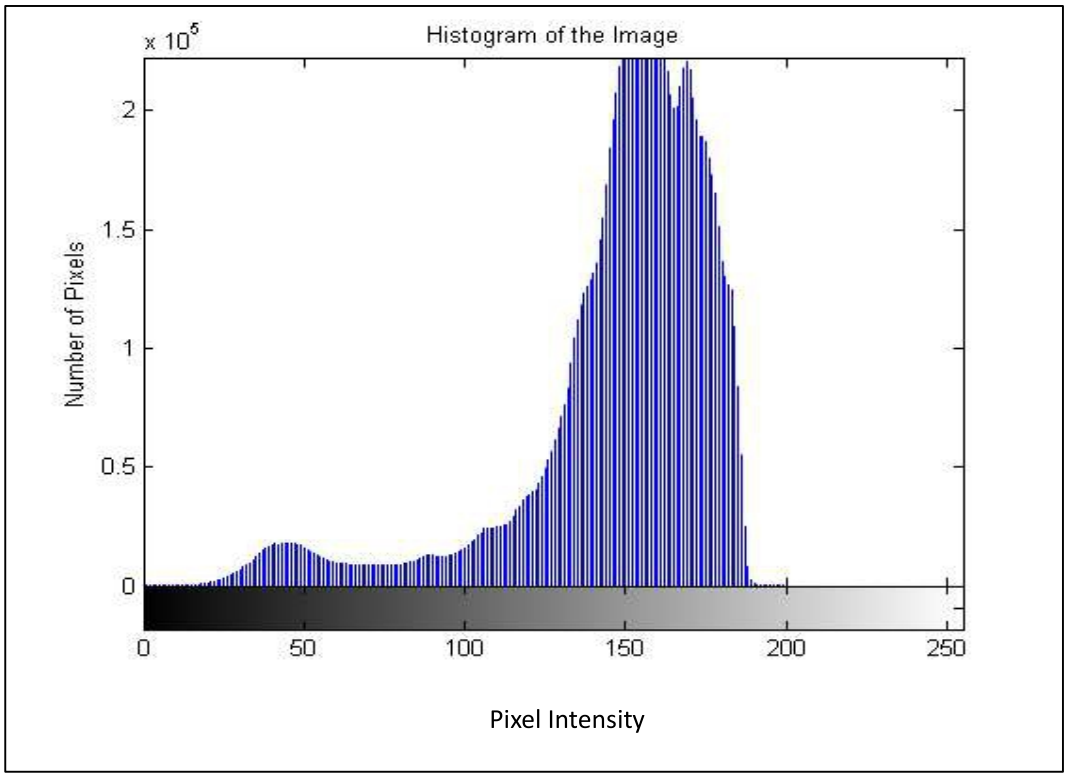

Figure 3.13 Histogram of Original Image (Sample 3) (Figure 3.14 a)

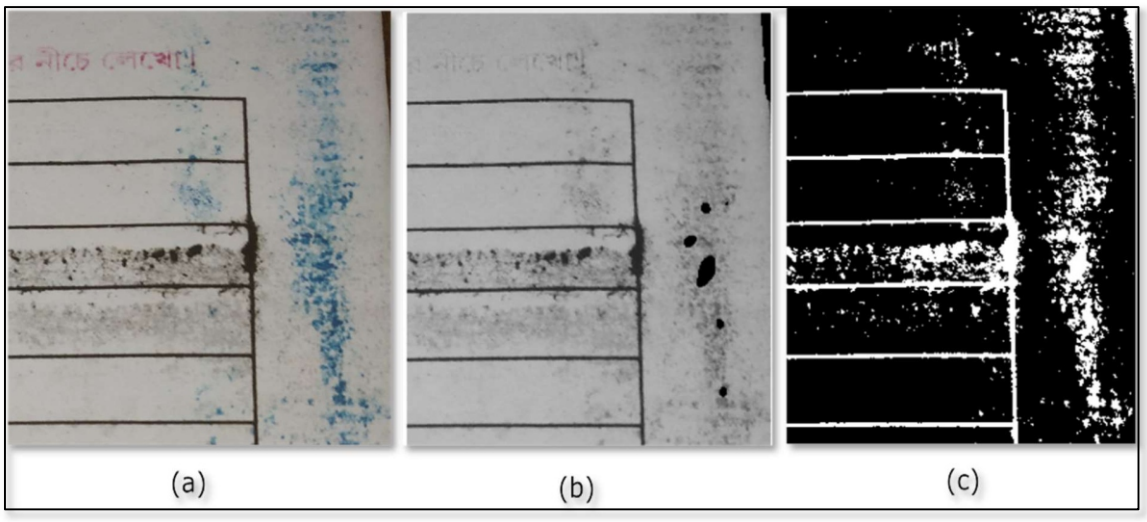

Figure.3.14 Resulted Image of Scumming (Sample 3) (a) Original Image (Mid Key) (b) Foreground Image (c) Detected Scum Pixels

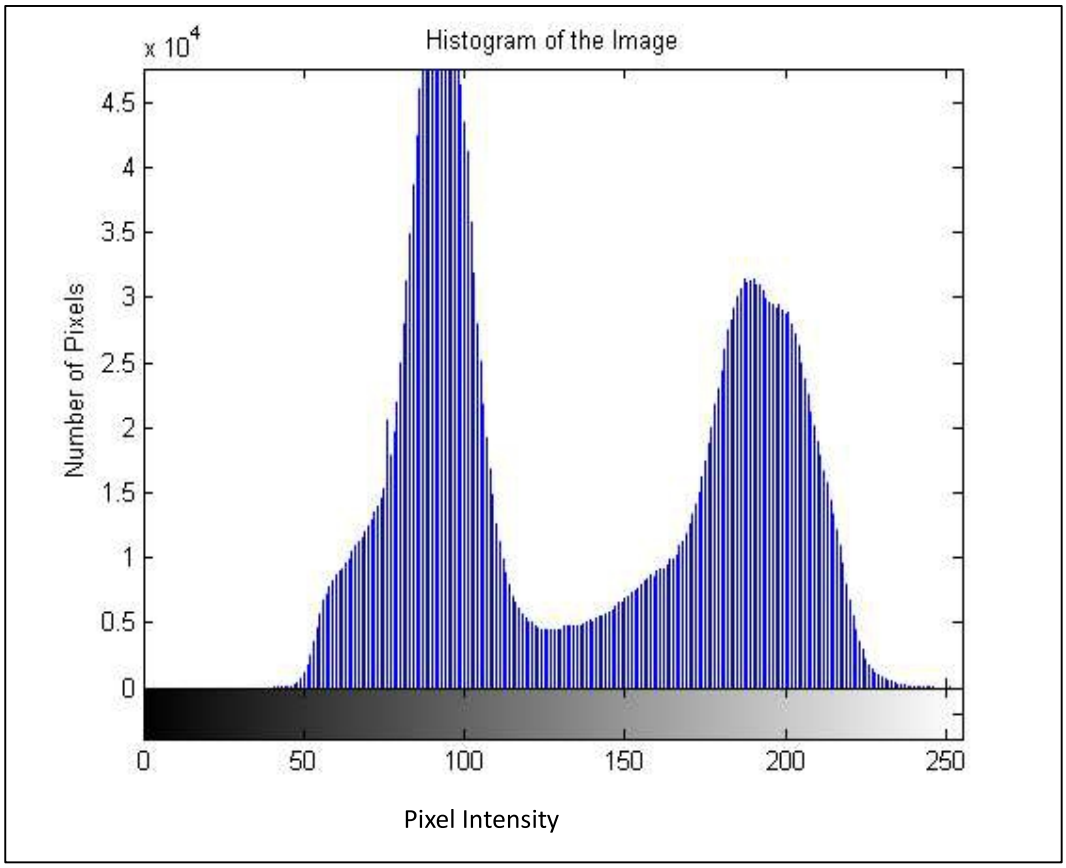

Figure 3.15 Histogram of Original Image (Sample 4) (Figure 3.16 a)

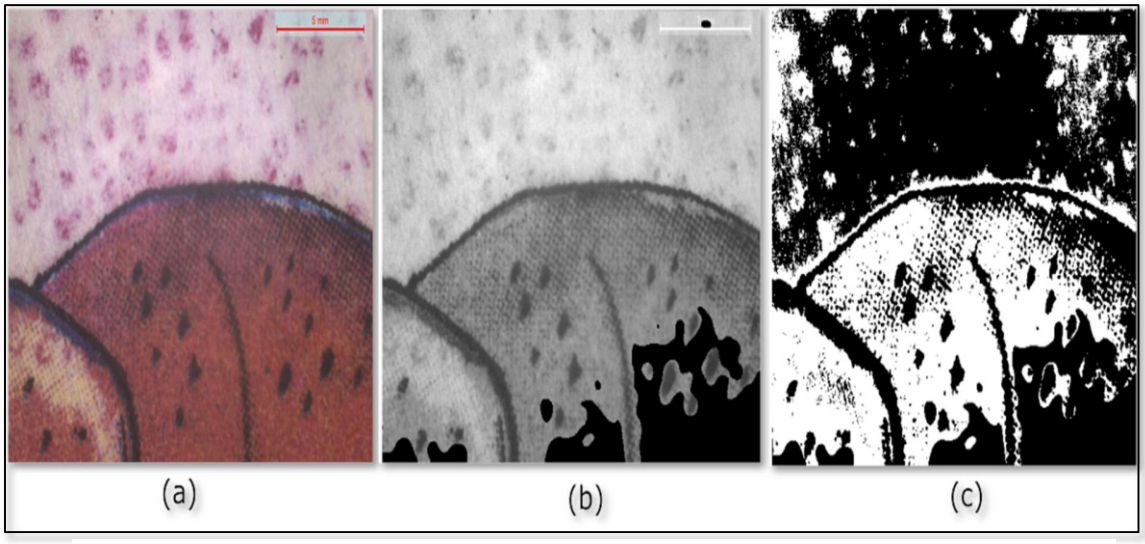

Figure.3.16 Resulted Image of Scumming (Sample 4) (a) Original Image (Mid Key) (b) Foreground Image (c) Detected Scum Pixels

3.5.3 For Low-key Image:

Figure. 3.18(a) illustrates the original image of a low-key sample and the histogram of this image is shown in Figure 3.17. Upon applying the bit-plane slicing technique, it is observed that bit plane-7 contained the majority of the scum pixels. Following this, the printed image area is isolated as the foreground, as shown in Figure. 3.18(b). Similar to the mid-key image analysis, the same intensity (0.45–0.75) is utilized after several trial and error during the second-level segmentation process. After segmentation, the scum pixels are identified and are depicted in Figure. 3.18(c). The scum pixel percentage for this sample is measured to be 42% (Table 3.1).

The same steps are followed for another low-key sample image illustrated in Figure.3.20. Histogram of this sample image is depicted in Figure 3.19. resulting in a calculated scum pixel percentage of 85% (Table 3.1) for this sample.

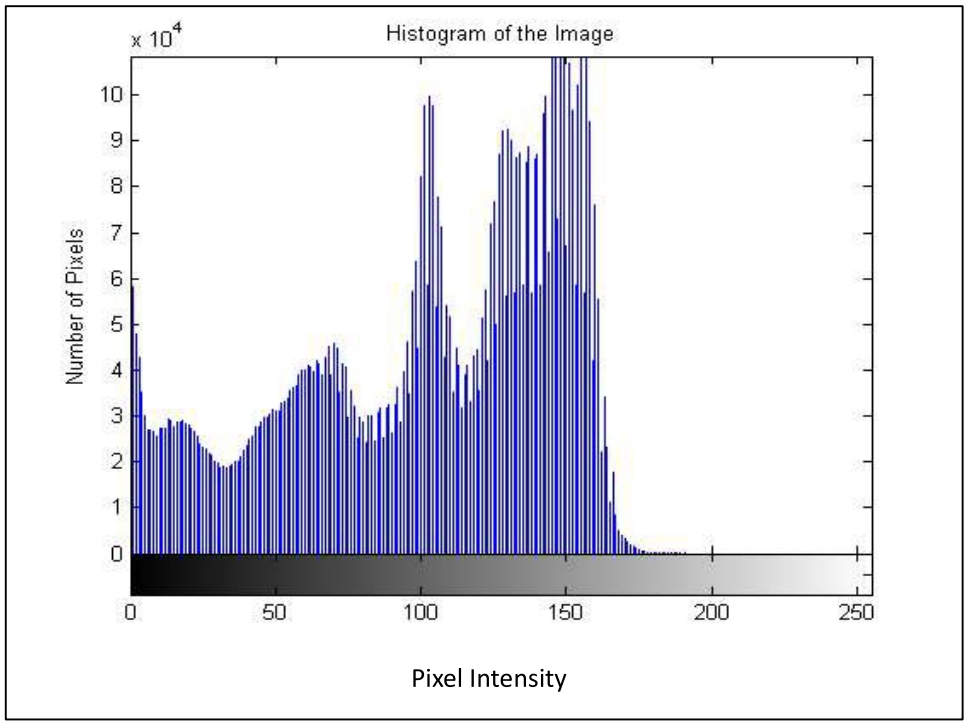

Figure 3.17 Histogram of Original Image (Sample 5) (Figure 3.18 a)

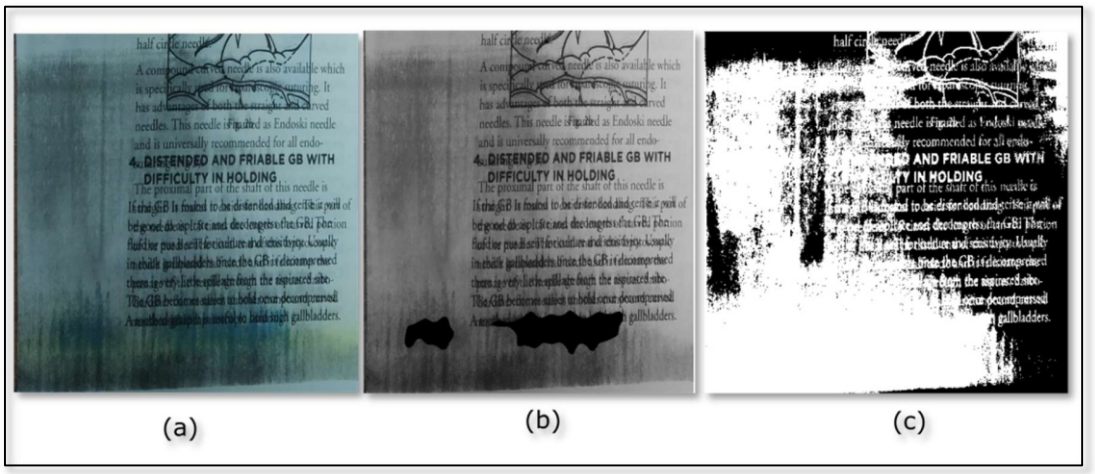

Figure 3.18 Resulted Image of Scumming (Sample 5) (a) Original Image (Low Key) (b) Foreground Image (c) Detected Scum Pixel

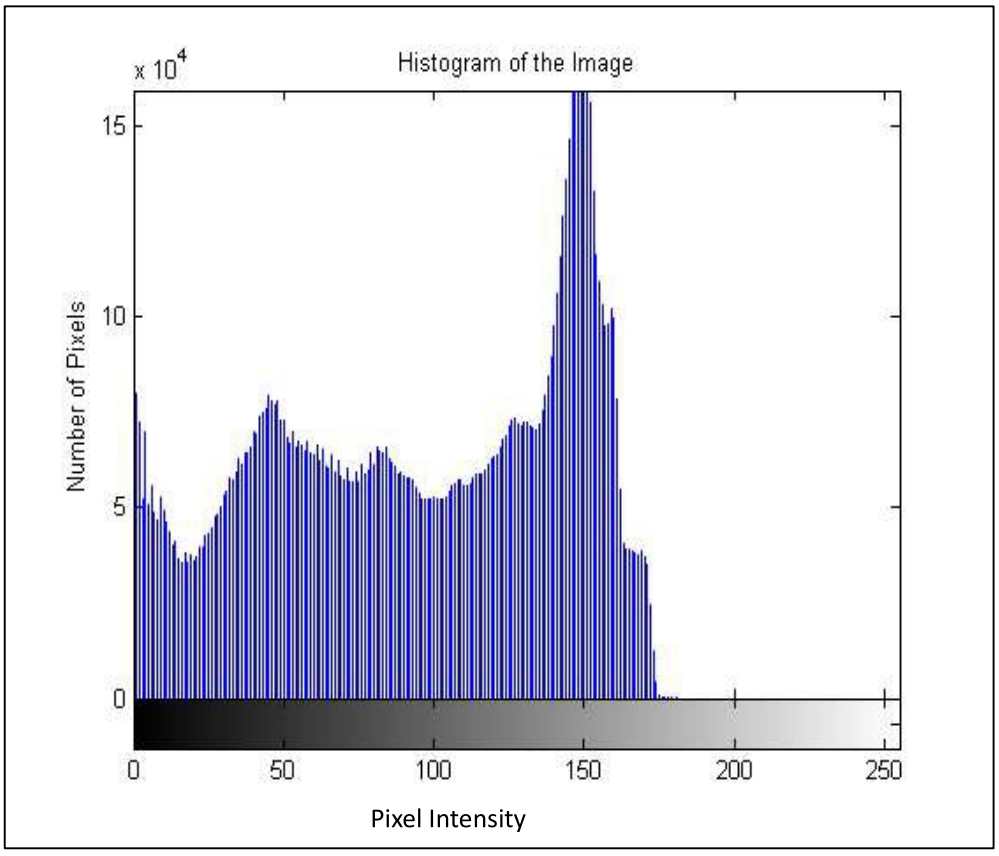

Figure 3.19 Histogram of Original Image (Sample 6) (Figure 3.20 a)

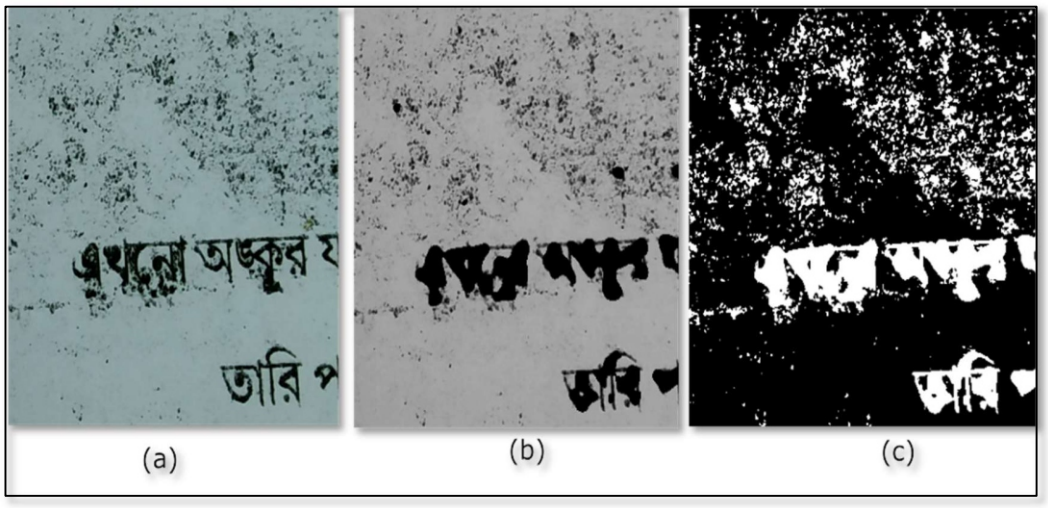

Figure.3.20 Resulted Image of Scumming (Sample 6) (a) Original Image (Low Key) (b) Foreground Image (c) Detected Scum Pixels

Table 3.1: Scumming Percentage of Test Images

Test Image	**Scumming Percentage (%)**
Sample 1 (Figure.3.10)	42
Sample 2 (Figure.3.12)	16
Sample 3 (Figure.3.14)	14
Sample 4 (Figure.3.16)	40
Sample 5 (Figure.3.18)	42
Sample 6 (Figure.3.20)	85

3.6 Conclusion

This study introduces a novel method for identifying and quantifying the scumming printing defect. The proposed technique has been rigorously tested on diverse samples obtained from offset presses, showcasing its robustness and practical applicability. By implementing the detection process entirely within the MATLAB environment, the method minimizes manual intervention, offering a more efficient alternative to traditional subjective measurement

techniques. Key advantages of the approach include automation and enhanced efficiency, which reduce manual effort and enable faster defect detection. Additionally, this technique paves the way for developing a mobile based application, further streamlining the defect detection process, shortening problem identification times, and ensuring consistent production of high-quality prints. The integration of artificial intelligence presents exciting opportunities to enhance defect detection and unlock advanced capabilities, such as automated correction of scumming errors.

The primary challenge arises when real-world printing samples exhibit additional defects, which can obstruct the accurate identification of scum pixels. Furthermore, the minimal density difference between print and scum pixels makes it difficult to distinguish between the two, posing a significant challenge in calculating the scum percentage with precision. A key area for future work is developing a mobile-based approach for detecting and removing scum pixels during the image reconstruction process, particularly for the archival preservation of printed documents.

CHAPTER - 4

Identification of Showthrough and Strikethrough Printing Defect

CHAPTER- 4

Identification of Showthrough and Strikethrough Printing Defect

4.1 Introduction

In the offset lithography printing process (Figure 3.1), showthrough and strikethrough are common issues. This planographic technique differentiates the image and non-image areas based on the surface chemistry of the printing image carrier [2]. Substrate and ink play an important role in this printing process. Showthrough print defect occurs when the printed material on one side of a substrate, such as paper, is visible from the reverse side [3]. This defect is particularly noticeable in thin or low-opacity paper [2] and can compromise the quality and readability of the print. Whereas strikethrough is somewhat different from showthrough caused by ink penetration through substrate [3]. Figure 4.1 and 4.2 depicts printed image of showthrough and strikethrough real life samples of offset process respectively.

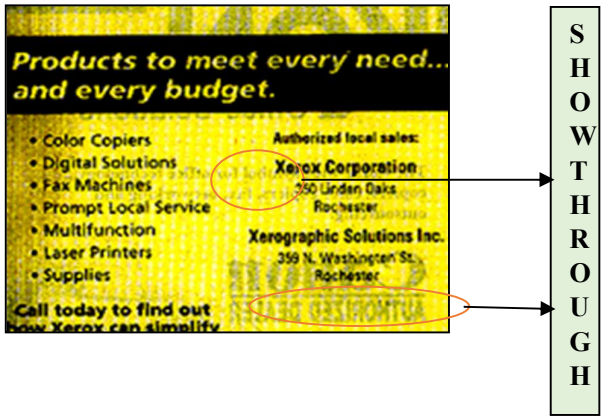

Figure 4.1 Sample Image of Showthrough Printing Defect

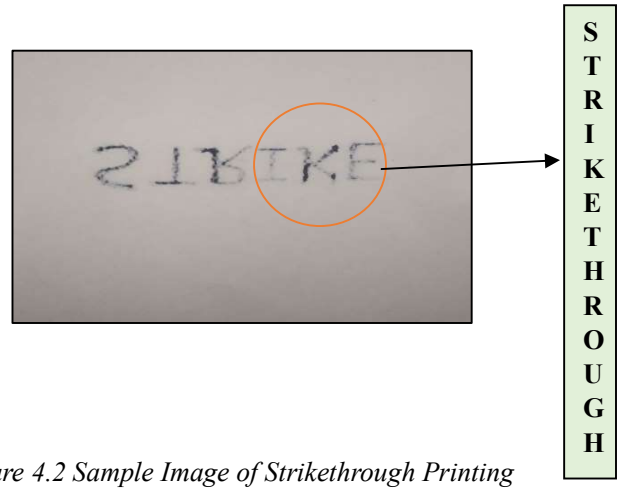

Figure 4.2 Sample Image of Strikethrough Printing Defect

The phenomenon of show-through in printing is closely related to the optical properties of light, including transmission, absorption, reflection, and scattering and determined by the opacity of paper [106]. When light strikes the surface of ordinary paper, it undergoes three interactions: one portion is reflected, another portion is absorbed, and the remainder is transmitted through the paper. Light transmission through paper occurs in two ways: as parallel rays that travel without scattering, and as scattered or diffused rays. The total light transmittance, which is the sum of parallel and diffuse transmission, determines the paper's degree of opacity [106]. Pure cellulose fibers are naturally transparent, while printing papers are typically somewhat translucent. Opacity in paper is enhanced by groundwood and unbleached fibers due to their higher light absorption. Papers with fillers exhibit high opacity because of the numerous light-scattering interfaces between fibers and air, air and fillers, and fibers and fillers. The effectiveness of fillers in scattering light and increasing opacity varies. Titanium dioxide stands

out as the most efficient opacifying agent, significantly boosting brightness. Tinting dyes used in blue-white papers [106] are more efficient at scattering light than those used in yellow or neutral-white papers, giving blue-white papers higher opacity than their yellow-tinted counterparts [106]. Similarly, yellow papers have lower opacity than white papers. Colors such as blue, green, and gray exhibit higher opacity than white due to their greater light absorption. Showthrough of images from the reverse side of a sheet reduces print contrast and affects the visual quality of the image. The degree of opacity depends on the filler composition, paper coating, contrast between the printed image and the paper etc. [106]. Achieving sufficient opacity and minimizing objectionable showthrough can often be challenging and expensive, especially with high-brightness white papers.

Showthrough is caused by the differences in total paper reflectance, on the observer viewing side (back side of the printed sheet), resulted from printed and non-printed areas.[106]. While the printing opacity method provides a more accurate measurement of the reflectance differences perceived as showthrough, the contrast ratio method is commonly used because of its speed and convenience. Moreover, Showthrough can be caused by ink with a vehicle system that is too fluid. The vehicle is absorbed too much by the substrate, making the paper become translucent [107].

Strike-through is not commonly encountered, but when it does occur, it is usually associated with the printing of absorptive-drying inks onto newsprint [107]. It happens when some of the printed ink is absorbed so deeply into the paper that it becomes visible on the other side. While it can interfere with readability, tends to make the paper more transparent [106] increasing the visibility on the reverse side of the paper, it is generally less problematic than showthrough. This issue has been linked to the use of oil-soluble blue toners in low-viscosity oil-based inks. The solution of this kind of problem is to use a different ink or a less absorbent substrate [107].

Porosity is another important property of paper which plays important role in order to avoid some printing defect like showthrough and strikethrough. It determines the degree to which a paper or substrate is porous. The substrate's porosity and the coating's affinity for ink solvents significantly influence the drying speed of the inking system [3]. Water, though relatively non-volatile and unlikely to cause issues due to evaporation from the cell, can pose challenges during the final drying stage, depending on the substrate's porosity [3]. Special attention must be given to selecting paper substrates that are not prone to cockling, as water-based systems exacerbate this issue, making it the most problematic scenario [3]. If the porosity is too high, the adhesive

and coatings may strike through the paper, or their solids may penetrate excessively, resulting in a weak adhesive bond or an insufficient ink film thickness on the paper surface [106]. Fiber refining, mechanical pulp addition, increasing short-fibre pulp can help to reduce the fibrous structure of the stock and also reducing the porosity of substrate [106]. Generally, in porous or uncoated paper showthrough and strikethrough print defect arises because of the excessive absorbent nature of paper which allows the ink vehicle to penetrate. Moreover, it can also occur when the refractive index of the ink vehicle is close to that of cellulose, making the printed area more transparent [3]. Dimensional change in paper is also affected by porosity. For an example, papers which are highly porous, minimize their dimensional change for making continuous form [106].

In showthrough printing defect, dried printed ink produces unwanted color on reverse side. Whereas if ink is too soft it can cause strike through. By maintaining the ink viscosity, keeping it high, these problems can be avoided [3]. Mostly in lithography the interaction between paper, ink and press condition aggravated printing problems. Ink too much tacky or soften can cause different print problems. Therefore, to avoid such kind of printing problems handling of ink is very much necessary step to take. Press operator should be careful and take every precaution to avoid contamination. Fountain solution and ink should be prevented from mixing. While scrapping the remaining ink, operator should be careful to scrap off evenly. Operator should make sure that press must be cleaned properly before inking up otherwise remaining dried ink can cause print problems like showthrough.

The amount and type of ink that is transferred is determined by the type of printing process. Excessive ink transfer may cause printing defects like showthrough and srtikethrough. The most important factor which affects ink receptivity and ink absorption nature among many is the percentage of binder in the coating. Higher ink receptivity and absorption will occur where binder pigment ratio is low [108].

Ink receptivity and ink absorption of the coated surface affected by the percentage of binder in the coating [108]. Another significant factor that influences ink receptivity is pigment type. Pigments which produce high pigment volume add to ink receptivity. Precipitated calcium carbonate, amorphous silicates, thermally treated clays are such pigments which helps to increased ink receptivity [108]. Ink receptivity reduces where pigments orient and pack together, particularly in the case of delaminated clays [108]. It is true that the coated surface should be ink receptive and absorption of ink is required for ink drying. But excessive ink absorption can

affect adversely and can be responsible for degradation of print quality as showthrough and strikethrough may occur.

Fillers of various types and quantities are added to the papermaking furnish to improve opacity, brightness, and smoothness while reducing ink strike-through [106]. Surface sizing enhances the surface strength of uncoated papers, providing the necessary strength for offset printing as it fills the voids among fibres [106]. It also increases ink holdout and reduces opacity and porosity which subsequently helps to reduce the chances of showthrough and strikethrough [106] The only remedy to avoid strikethrough is to use less absorbent substrate or non-porous paper, usage of quick set ink and maintain the printing pressure [3]. The ink setting speed is very important criterion to avoid strikethrough as well as showthrough. Moreover, excessive printing impression pressure causes bleeding of the printed image through the paper [2]. The choice of resin and oil determines the wetting characteristics of ink vehicle. In newspaper printing an asphaltum complex solution in mineral oil with the correct addition (2or 3%) is used which helps to avoid excessive strikethrough in printing [3]. To increase paper opacity, brightness, whiteness and smoothness, fillers are used to control ink absorption and reduce the chances of strikethrough and showthrough. The most common fillers used for this purpose are calcium carbonate, titanium dioxide etc. [4]. Paper sizing is one of the most common methods to control the penetration of ink through the paper. Rosin is common sizing agent that provides significant resistance to wetting paper by ink [4].

4.2 Sample Preparation

Samples are collected from different commercial offset presses. Ink used for the sample of showthrough is paste ink whereas for strikethrough sample liquid ink is used. At the very first step substrate properties like thickness and porosity of the samples are measured. Thickness of the substrate is measured using Micrometer (make: S.C.Dey & co.) (Figure 4.4). Micrometers measure the thickness with additional precision of even single sheets of paper or any substrate. Therefore, for maximum accuracy Micrometer is the good choice and should be considered. Porosity of the samples is measured by Gurley Densometer [6] (Figure 4.5) based on air leak method. Densometer helps to measure the porosity of materials such as papers, plastic and membranes etc. After measuring the substrate properties samples are placed into a handmade camera setup environment (shown Figure 3.3) to take digital copies of the samples. Here for this experiment mobile camera with 48-megapixel resolution and D65 illumination is used.

4.3 Problem Statement

The primary objective of this study is to develop an easier automated method for identifying showthrough and strikethrough print defects with reduced computational load. In the printing industry, these defects are typically identified by operators by eye assumption, who take corrective actions based on their ability to detect the issues. However, this process is subjective and varies from person to person. An automated approach could replace the traditional human perception-based method, providing a more consistent and reliable solution. Moreover, for a better-quality print, these defects must be avoided as it can hinder readability.

To implement an image processing or computer vision-based approach, a digital image is required, as it differs from the printed image in terms of data representation, color representation, manipulability etc. Digital images consist of pixel information, with features such as contrast and intensity being utilized in various applications of image processing. The proposed method aims to segment the printed image from the pixels associated with showthrough and strikethrough defects.

In the case of showthrough, identifying and segmenting pixels with lower intensity from the background presents a challenge. For strikethrough, ink penetration causes the defect pixels to overlap with the print pixels in the foreground. The challenge arises when pixels of similar intensity are present in both the foreground and background, complicating the identification of strikethrough defects. Additionally, noise from other printing defects, such as scumming and tinting, can interfere with the accurate detection of showthrough and strikethrough pixels.

Achieving the goal of minimizing manual intervention through automation, precise pixel segregation is crucial. Without accurate segmentation, print pixels could be mistakenly identified as showthrough or strikethrough defects.

4.4 Proposed Method

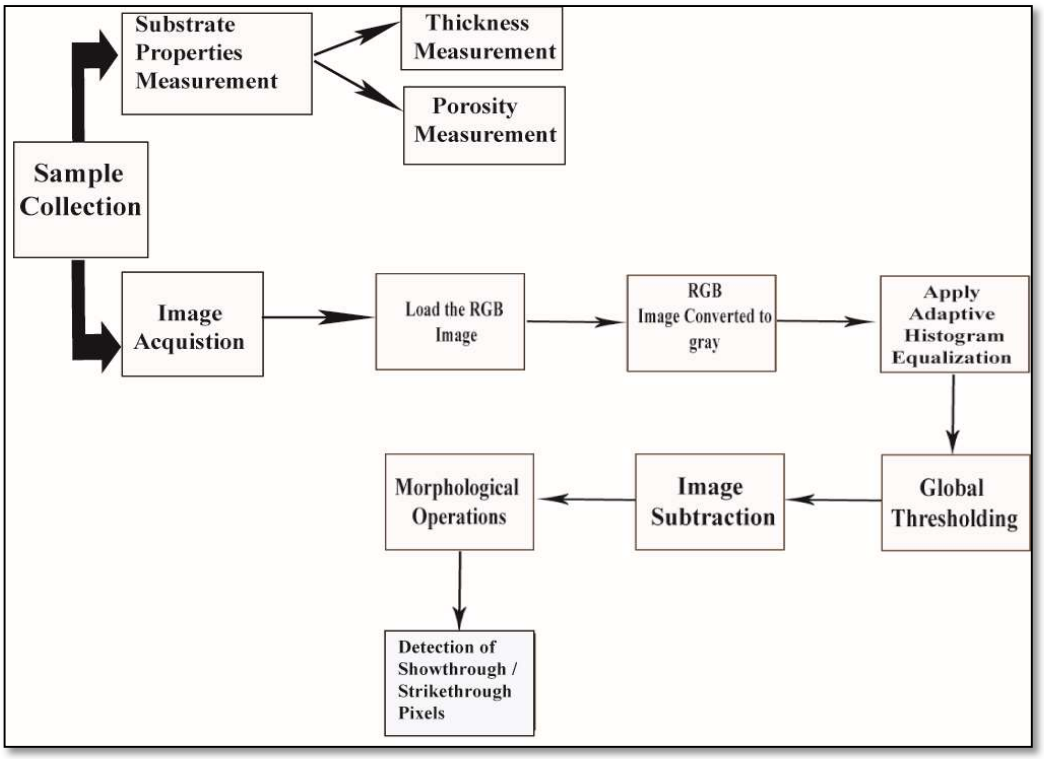

Figure 4.3 Flowchart of Proposed method for Detection of Showthrough and Strikethrough Printing Defect

The steps involved in the proposed method [109] for detecting showthrough and strikethrough print defects are illustrated in the flowchart in Figure 4.3. For this experiment, all samples are collected from various commercial offset presses. The necessary steps for identifying showthrough and strikethrough pixels are briefly described. It is important to note that the same procedure and steps are applied to both defects. Since these two print defects share similarities in their causes and characteristics, the same algorithm is used for their detection.

4.4.1 Substrate Properties Measurement

Substrate properties like thickness, porosity, is measured at the very first step of this detection process. This step is very much necessary as because showthrough and strikethrough print defects depends on the substrate characteristics. Understanding the nature of the substrate helps

to prevent these defects. Here the properties are measured and the procedure of the measurement is given below.

4.4.1.1 Thickness Measurement

Substrate thickness matters in case of printing as because thicker substrate absorbs ink differently than thinner paper. Here for this measurement procedure printed samples are collected from offset presses. At first thickness of these samples is measured using Dial gauge Micrometer (make: S.C.Dey & co.) (Figure 4.4). A dial gauge micrometer measures paper thickness using mechanical amplification and a screw gauge mechanism for high precision. The paper is placed between the fixed anvil and the movable spindle connected to a precision screw. Turning the ratchet knob or thimble moves the spindle toward the anvil, applying gentle pressure on paper. The spindle movement is transferred to a rack-and-pinion or lever system, amplifying small variations, and the dial indicator provides a resolution of 0.001 mm (1 μm). A spring-loaded spindle ensures consistent pressure, preventing excessive compression for accurate readings The result of the thickness measurement for the tested sample is shown in Table 4.1.

Figure 4.4 Micrometre for Measuring the Thickness of Substrate

4.4.1.2 Porosity Measurement

Porosity is an important substrate property which matters in printing and plays an important role to avoid printing defect like showthrough and strikethrough. The Gurley Densometer

(Figure 4.5) based on air leak method, is commonly used for measuring porosity in substrate. It measures the time required for air (100 cc) to pass through a given area of a material under standard conditions. This method is suitable for porous materials like paper or nonwoven fabrics. To measure porosity using a Gurley densometer, first cut a sample of the material, ensuring it fits the size requirements of the densometer. The sample must be in a controlled environment to standardize its moisture content. Ensure the densometer is clean and calibrated. Set the instrument to the specified test conditions, including temperature and pressure. Place the sample onto the instrument, making sure it covers the testing area completely. Sample material is held between clamping plates which has circular orifice area. Measure the time it takes for a certain volume of air generally 100cc, to pass through the sample. Record the time taken for air to pass through the substrate using stopwatch usually in seconds. The air pressure is produced by a cylinder of a specific diameter floating freely inside the oil filled outer cylinder. Repeat the steps for accurate results. The obtained results from this porosity measurement for tested samples are given in Table 4.1.

Figure 4.5 Densometer for Measuring the Porosity of Substrate

4.4.2 Image Conversion

The collected offset real-life defected printed samples are first acquisitioned using a mobile camera in a handmade set-up (shown in Figure 3.3) environment with D65 illumination. Onwards all the necessary step for the identification of showthrough and strikethrough printing defect has been done in MATLAB programming software. After capturing the image of defected printed samples, it is converted into gray color space from RGB color space. The choice between using grayscale or RGB color space for image processing depends on the specific requirements of the task. Here for this detection purpose gray color space is chosen as because it has only one channel representing intensity or brightness which makes easier to perform any image processing operations [103] and also reduce computational load. In gray color space intensity value of pixel ranges from 0(black) to 1(white) and any fractional values in between. Moreover, grayscale images can be more suitable for the tasks that rely on intensity information. Therefore, for this particular printing defect, gray scale is more suitable than RGB color space. Moreover, the RGB color space is device dependent; [110] as a result pixel information varies from device to device.

Converting an image from the RGB color space to gray color space can be achieved using various formulas. Generally, a simple conversion with the weights 0.299, 0.587, 0.114are commonly used and derived from the luminance values associated with each color channel [110]. The simple and widely used approach of the conversion is:

$$Gray = 0.299*Red + 0.587*Green + 0.114*Blue \qquad \text{eq. 4.1}$$

4.4.3 Histogram Equalization

Histogram equalization [63] [66] [67] is one of the most important operations in image processing and computer vision. The intensity value of showthrough pixels is very much low and strikethrough pixels often get merged with print pixel, it is necessary to enhance the contrast of the image to detect the pixels properly. Therefore, the histogram equalization technique is applied to enhance the image intensity while preserving the overall details of the image. Histogram of an image is the basis for various spatial domain processing techniques in image processing whereas manipulation of histogram is very much effective for image enhancement [63] [103]. There are several steps for histogram equalization for image contrast enhancement as follows [103] [111]:

I. First read the image pixel by pixel

II. Get histogram of original image where the frequency of each pixel's intensity values is shown.

III. Find the intensity value represented on the horizontal axis of the histogram.

IV. Compute the probability density function (PDF) for each intensity value by dividing each bin count by the total no. of pixel in image. Probability density function is a way to represent the distribution of pixel intensities of the image.

V. After getting the PDF, compute cumulative density function (CDF) for each pixel intensity's frequency. For a discrete image with intensity levels 0,1,2……L-1, where L is the no. of intensity levels. The CDF will be

$$CDF(i) = \sum_{j=0}^{i} PDF(j) \qquad \text{eq. 4.2}$$

Where,

i= intensity level

PDF(j)= Probability Density Function at intensity level j

CDF(i)= cumulative probability up to intensity level i

VI. The range of CDF value is 0-1. Therefore, CDF value is multiplied by largest value of intensity i.e. 255

VII. Apply the histogram equalization transformation function to each pixel in the original image using CDF. In this step the intensity values are redistributed. The histogram equalization transformation function T is defined as:

$$T(r) = \frac{L-1}{N} \sum_{j=0}^{r} PDF(j) \qquad \text{eq. 4.3}$$

Where,

r = original intensity value

N= total no. of pixels in the image

VII. Compute the equalized image histogram and normalize the histogram to get the equalized PDF

Thus, the contrast of the output image will be enhanced with a more uniform PDF after histogram equalization. Now the enhanced image without lack of any information is better for further image processing applications. This method applies histogram equalization technique globally. This histogram equalization technique is also called Adaptive histogram equalization and it is more effective than simple histogram equalization techniques.

4.4.4 Segmentation using Global Thresholding Algorithm

Thresholding is the most important part in image processing. It helps to segment the image into parts. There are several thresholding algorithms exist, each with its own approach. Global thresholding [49] [103] can be done using methods like Otsu's method or choosing a threshold value based on histogram analysis. Here in this presented method, Otsu's thresholding algorithm [103] is applied to segment the image into foreground and background. As it is assumed that all the showthrough pixels and strikethrough pixels are in background of the image and printed pixels are in foreground. Therefore, before any further operation segmentation between foreground and background of the image is a necessary step to take. Now in this global thresholding algorithm the whole segmentation process has done based on the intensity of the pixel in image. The algorithm gives a single intensity threshold value that separates the image background from foreground. In this thresholding method the grayscale histogram of an image is used to select an optimal threshold value. This threshold value then separates the image into two regions with maximum inter-class variance. The steps of segmentation are like:

I. First, compute the grayscale histogram of the sample image. The histogram can be obtained by counting the number of pixels at each intensity level. The grayscale histogram $H(i)$ of an image I is defined by the following equation:

$$H(i) = \sum_{x=1}^{M} \sum_{y=1}^{N} [I(x, y) = i] \qquad \text{eq. 4.4}$$

Where,

M= height of the Image

N= Width of the Image

II. Next compute the Cumulative Distribution Function (CDF) which represents the probability that the intensity value of a pixel in image is less than or equal to a particular level. The mathematical equation of calculating CDF is:

$$C(i) = \sum_{j=0}^{i} \frac{H(j)}{M \times N} \qquad \text{eq. 4.5}$$

III. The mean grayscale intensity value of the image is computed in next step. In Otsu's global thresholding method, the mean of grayscale value is used to compute the class variance. The mathematical equation of calculating mean grayscale intensity value is as follows:

$$\mu = \frac{1}{M \times N} \sum_{x=1}^{M} \sum_{y=1}^{N} I(x, y) \qquad \text{eq. 4.6}$$

Where,

$\mu =$ Image's mean gray scale intensity value.

IV. The between class variance is computed in next step where class variance is the product of the square of the difference between the image's mean gray scale intensity values and the probability of background and foreground regions. For the threshold value T, class variance between background and foreground is as follows:

$$var(T) = P_0(T).P_1(T).(m_0(T) - m_1(T)) \qquad \text{eq. 4.7}$$

Where,

$P_0(T) =$ Probabilities of the background

$P_1(T) =$ Probabilities of the foreground

$m_0(T)$= Mean grayscale intensity value of the background

$m_1(T)$= Mean grayscale intensity value of the foreground

The mean grayscale intensity values, $m_0(T)$ and $m_1(T)$ can be calculated as:

$$m_0(T) = \frac{\sum_{i=0}^{T-1} i.H(i)}{P_0(T).MN} \qquad \text{eq. 4.8}$$

$$m_1(T) = \frac{\sum_{i=T}^{255} i.H(i)}{P_1(T).MN} \qquad \text{eq. 4.9}$$

V. Finally calculate the optimal threshold value for image segmentation. Therefore, to obtain optimal threshold value first between class variance for each possible threshold value need to be calculated. The optimal threshold value can be mathematically expressed as follows:

$$T_{opt} = argmax_T(var(T)) \qquad \text{eq. 4.10}$$

Now this optimal threshold value is considered and applied to the gray scale image to obtain binary image. In this binary image the pixels having intensity values above the optimal threshold value are considered to be the foreground image pixel and the pixels having lower intensity values than optimal threshold value is considered as background pixel of image.

Otsu's global thresholding method has many advantages as well as disadvantages. As because it is simple and less time taking comparing other thresholding method this thresholding algorithm used widely in image processing and computer vision. Moreover, this algorithm does not need any prior knowledge about the image to separate its foreground from background. However, the limitation of Otsu's algorithm should be taken into consideration. The result of this algorithm may not be accurate with improper illumination. This algorithm is not robust to noise. Images with multiple histograms may not give proper result with this algorithm. To overcome this problem pre- processing of images like, division of image into smaller parts to apply thresholding method locally or application of Gaussian or median filter to reduce noise can be helpful making the thresholding method more reliable.

4.4.5 Image Subtraction

In case of showthrough and strikethrough defect detection thresholding is not enough to segregate the pixels from foreground. Therefore, image subtraction has been done. Image subtraction [103] computed pixel-wise difference between corresponding pixels of two images. Before applying subtraction, it should be ensured that both images have the same dimensions. This process is used usually in image processing and computer vision for various applications like changes in between consecutive frames in video, motion detection, and change analysis. The general equation for image subtraction is: Result= (Image_1- Image_2).

In this proposed method, the thresholded image is subtracted from the histogram-equalized image to obtain the background image containing show-through pixels. Since an 8-bit image has pixel values ranging from 0 to 255, the resulting difference image can have values between

-255 and 255 [103]. Therefore, proper scaling is required. A fast and simple scaling method involves adding 255 to each pixel value and dividing by 2, ensuring all pixel values fall within the 0–255 range. However, this method has limitations and may lead to a loss of accuracy. A more precise approach involves first determining the minimum pixel value in the difference image. The absolute value of this minimum is then added to all pixels, effectively shifting the range to start from zero. Finally, the image is scaled to the 0–255 range by multiplying each pixel by 255 / (maximum value in the adjusted difference image). This ensures that the full intensity range is utilized while preserving details more accurately [103].

4.4.6 Morphological Operations

Manipulation of the shape or structure of objects in image is termed as morphological operations [103] [111]. This particular operation is mainly applied to binary or grayscale image. In image processing and computer vision, application of morphological operations is very much useful in noise reduction, segmentation, and feature extraction. The fundamental morphological operations include Dilation and Erosion operations. Here in this presented method morphological operations are done to extract the showthrough and strikethrough pixel without losing any information. In this presented method subtracted image is first erode and then dilated with structural element. This step is necessary to segregate and detect showthrough and strikethrough pixel in image. The details of these two operations; Dilation and Erosion, is described below:

4.4.6.1 Dilation

The morphological operation dilation [103] expands the boundaries of objects in image. It is performed by a predefined structuring element which moves over the image. In dilation operation the value of each pixel in the image is set to 1 if any pixel in structuring element is 1. Let's assume A and B are two sets in Z2. Then the set A is dilated by the structuring element B can be expressed as:

$$A \oplus B = \{z|(\hat{B}) \cap A \neq \phi\} \qquad \text{eq. 4.11}$$

This equation is obtained by moving the structuring element into B and its reflection of origin. The dilation of A by the structuring element B is performed when at least one element of A and B is overlapped. Therefore, the above equation can be interpreted as:

$$A \oplus B = \{z|[(\hat{B}) \cap A]\} \subseteq A \qquad \text{eq. 4.12}$$

4.4.6.2 Erosion

Unlike dilation [103] which expands the boundary, erosion shrinks the boundaries of objects in image. Here also predefined structuring element moves all over the pixels to perform the operation. Let's assume A and B are two sets in Z2. Then the set A is dilated by the structuring element B can be expressed as [103]:

$$A \ominus B = \{z|(\hat{B}) \subseteq A \qquad \text{eq. 4.13}$$

This equation is obtained by moving the structuring element into B and its reflection of origin. The erosion of A by the structuring element B is performed when at least one element of A and B is overlapped.

4.5 Result and Discussion

The presented method [109] is tested with numerous samples collected from commercial offset presses. Mainly all these samples are real-life offset printed. A mobile camera with 48 megapixel and D65 illumination in a handmade experimental set up (shown in Figure 3.3) is used to capture the digital form of these samples. The main purpose behind using mobile camera is to make the procedure of detection cost effective with an automated approach. Here some of the samples result and discussion are given below.

4.5.1 Result and Discussion for Showthrough

Figure 4.6, 4.7, 4.8 and 4.9 depicted some of the tested image samples of showthrough with their resulted image.

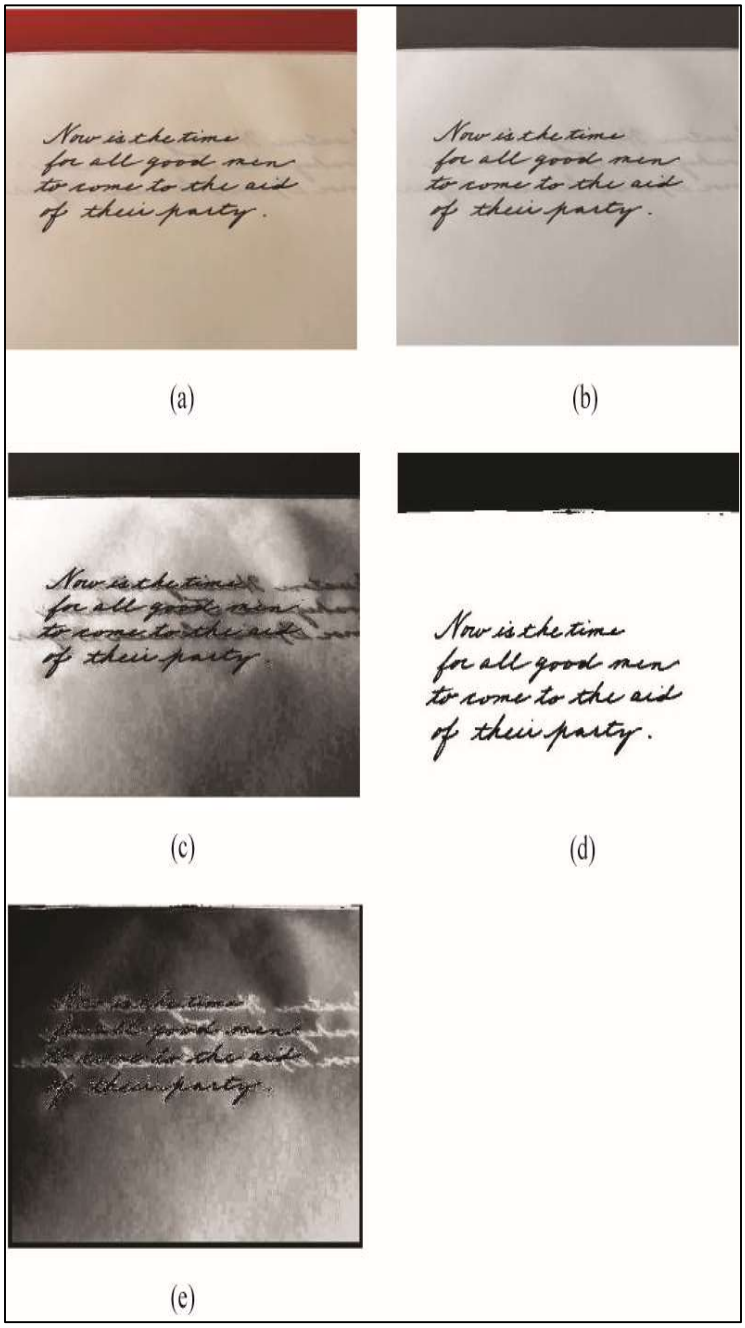

Figure 4.6 Resulted Images of Showthrough (Sample 1) (a) Original image (b) RGB to Gray Converted image (c) Histogram Equalized Image (d) Foreground Image (e) Detected Showthrough Pixels.

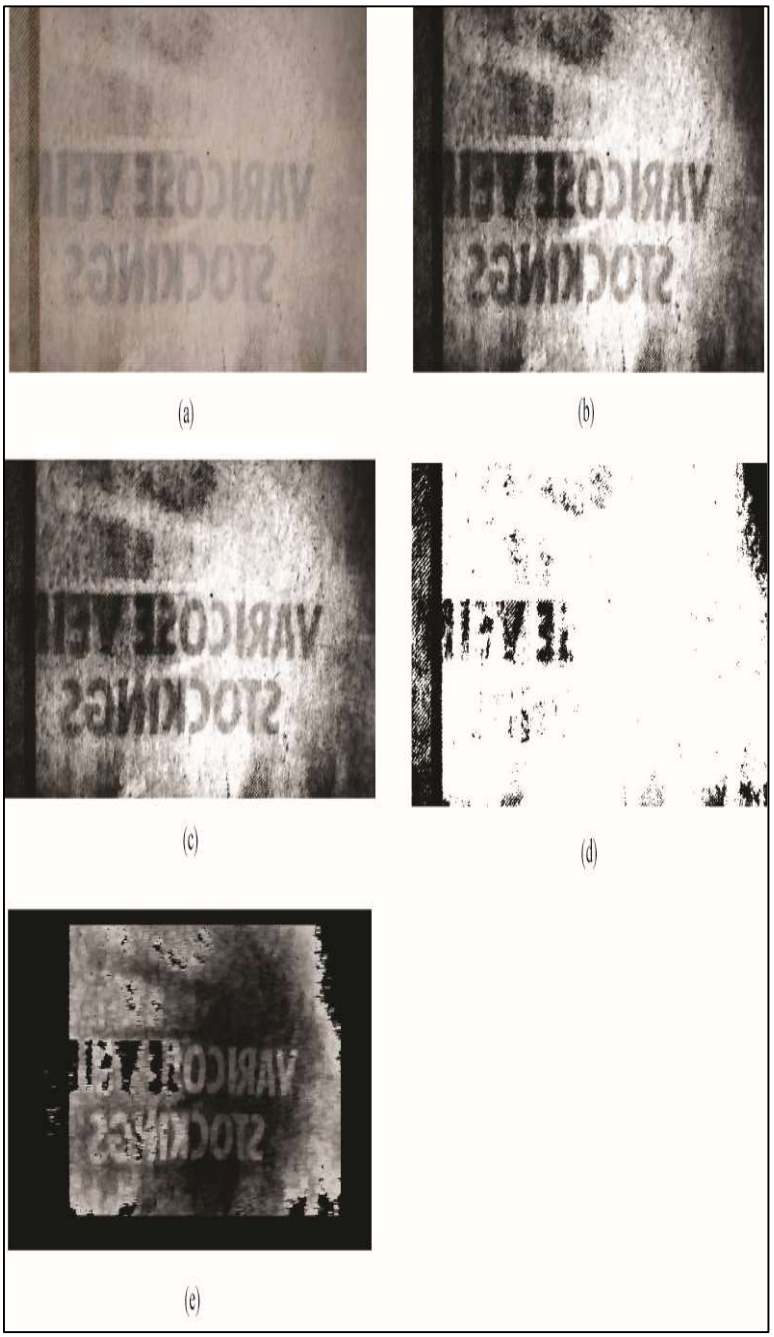

Figure 4.7 Resulted Images of Showthrough (Sample 2) (a) Original image (b) RGB to Gray Converted image (c) Histogram Equalized Image (d) Foreground Image (e) Detected Showthrough Pixels.

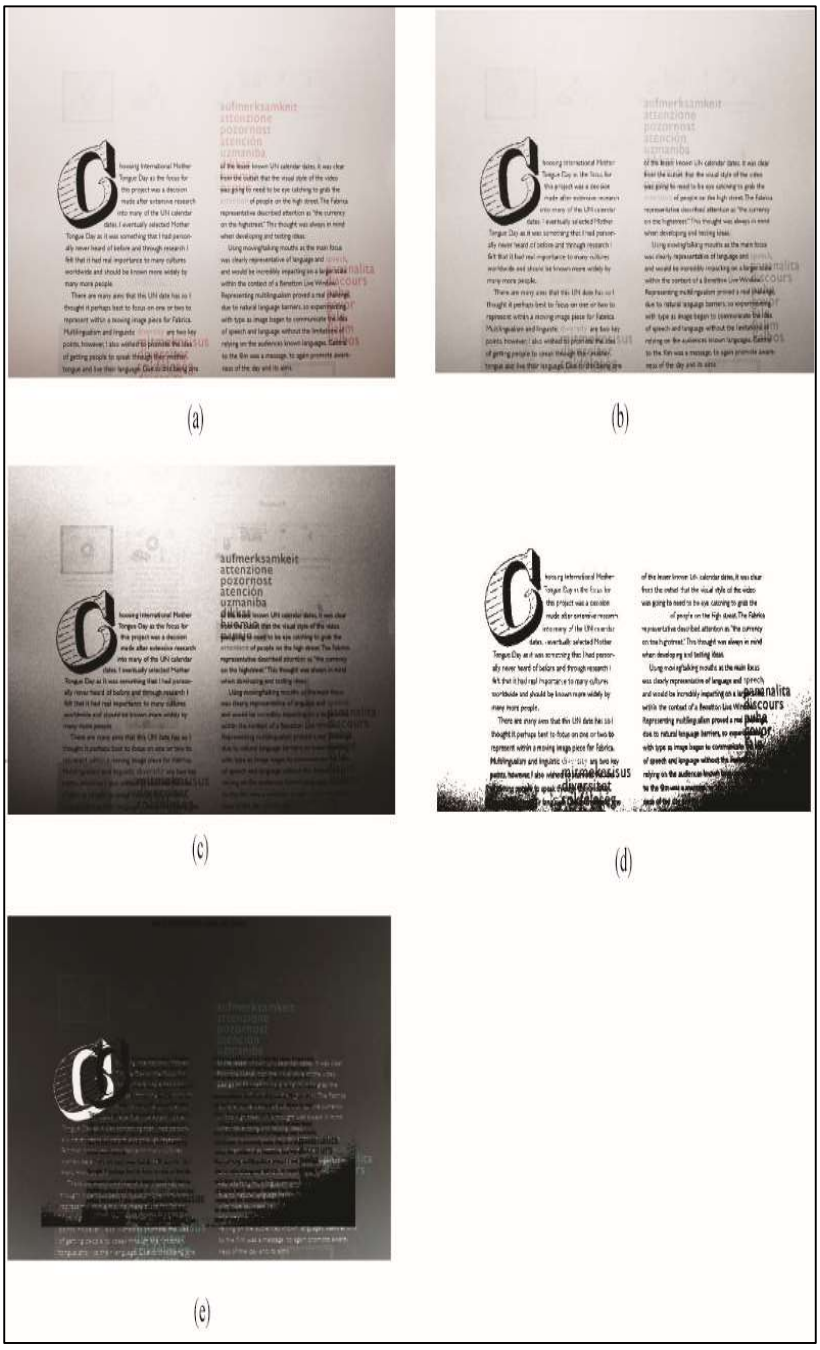

Figure 4.8 Resulted Images of Showthrough (Sample 3) (a) Original image (b) RGB to Gray Converted image (c) Histogram Equalized Image (d) Foreground Image (e) Detected Showthrough Pixels.

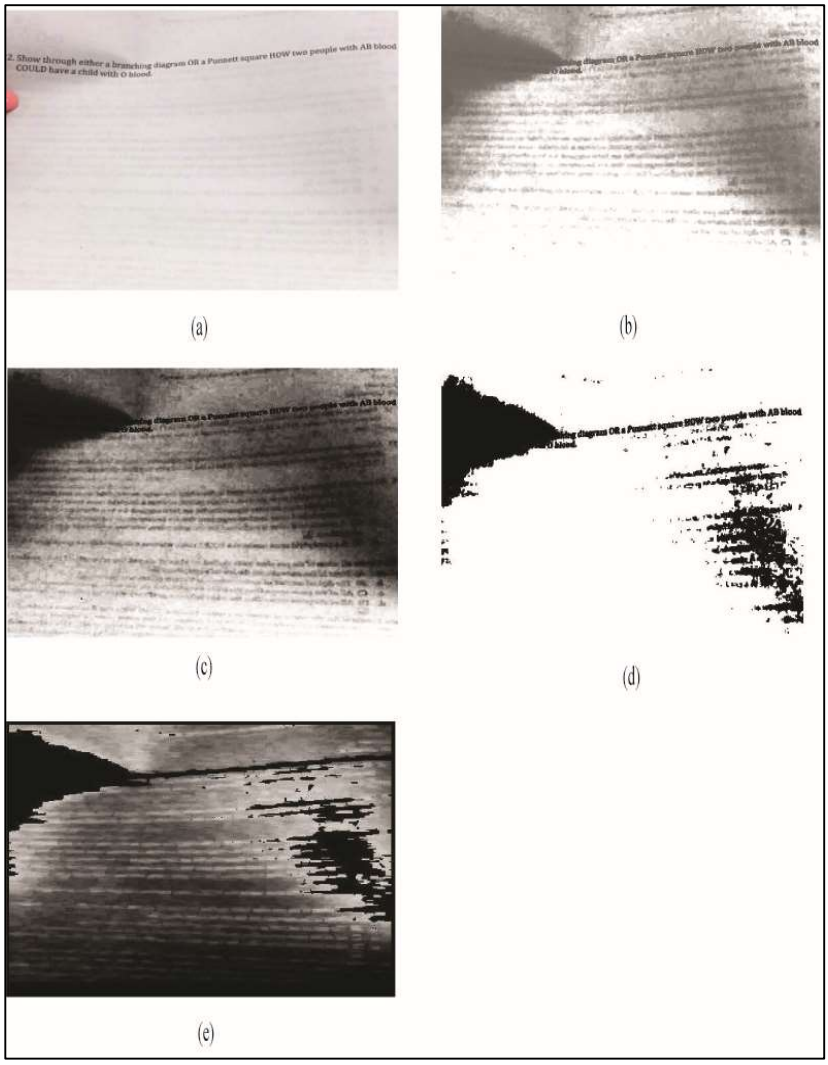

Figure 4.9 Resulted Images of Showthrough (Sample 4) (a) Original image (b) RGB to Gray Converted image (c) Histogram Equalized Image (d) Foreground Image (e) Detected Showthrough Pixels.

In Figure 4.6(a) original digital image of sample1 is shown. In this image, showthrough is clearly visible in the background and is overlapped with the foreground text which degrades the quality of printed document. Moreover, it becomes difficult to identify and segregate the showthrough pixels from background as because the showthrough pixels are having very much low intensity. Before doing any further operations according to the proposed method, at first the original RGB image is converted into gray color space. The gray scale image is shown in Figure 4.6(b). Now histogram equalization for contrast enhancement has been applied to the

gray color image. Histogram equalized image is shown in Figure 4.6(c). Contrast enhancement helps to increase the intensity level of showthrough in background. Thus, the pixel of showthrough becomes so prominent. Hence it helps to segregate showthrough pixels from print pixel of foreground. Now this histogram equalized image is considered for further operation. Adaptive global thresholding algorithm is applied to the histogram equalized image. Pixel values greater than optimal threshold value is considered as foreground pixel whereas pixel value less than optimal threshold value is considered as background pixel. And here the defected pixel that is showthrough pixels are considered to be in background. In Figure 4.6(d) the segmentation of foreground image is shown where it can be seen that only the text to be printed coming as absolute black. Now after segmentation of foreground and background of the image subtraction algorithm is applied to get the showthrough pixels in background. The subtraction is applied between histogram equalized image and thresholded image. After subtraction morphological operation like erosion and dilation operation is performed on subtracted image to extract the showthrough pixels properly. Figure 4.6(e) shows the showthrough pixels in background image appears in white.

In Figure 4.7(a) original digital image of Sample 2 is shown. In this sample not only, showthrough pixels are present but scumming pixels are also there in foreground image. These scum pixels can hinder the process of detection of showthrough pixels. As the intensity of scum pixel is almost same as print pixel, sometimes it can be predicted as print pixel after segmentation. Therefore, to overcome this problem at first scum pixels should be detected prior the detection of showthrough pixels. Moreover, post processing operations can be helpful to identify and ignore the scum pixels. The same steps are here also followed for this sample to segregate between print pixel and showthrough pixel. In Figure 4.7(e) the showthrough pixels in white is detected.

Original images with resulted image of Sample 3 are depicted in Figure 4.8. The main difficulties for this sample arise because of the presence of shadow part. The final output image in Figure 4.8(e) shows the detected showthrough pixels in white. However, the appearance of shadow cannot be ignored and can possibly remove by post processing operations. In this context it must be kept in mind that the real-life samples must contain other print defects or any other difficulties may arise because of environmental condition. Those unavoidable conditions must be considered and can be removed by other applications. The final output containing showthrough pixels is shown in Figure 4.8 (e).

Sample 4 in Figure 4.9 is also degraded by shadow part and the showthrough pixels as well. Showthrough pixels have very much low intensity in this sample which makes the detection procedure more difficult. Contrast enhancement helps to improve overall intensity of the image and makes easier to identify the showthrough pixels. After histogram equalization of the gray scale image, the background pixels are clearly visible in Figure 4.9(c). In Figure 4.9(e) the white pixels in background are identified as showthrough pixels and the appearance of shadow can be removed by post image processing operations.

As the showthrough printing defects depends on the substrate properties, porosity and thickness of sample paper is measured. As the sample substrate is different in terms of quality, as because it is collected from various offset presses, the result shows how the properties of substrate affect the showthrough print defect. In Table 4.1 the values of porosity and thickness of the above considered sample is given. According to the measurement sample 2 has the maximum porosity whereas sample 1 has the lowest porosity. The order of porosity (high to low) can be arranged as- Sample 2> Sample 4>Sample 3> Sample 1. Showthrough print defect in Sample 2 is more than other samples as the porosity is high. Sample 1 is less porous and the identification of showthrough pixels in sample 1 is little bit difficult as it is having lower intensity value. According to the intensity of the showthrough pixel it is clearly visible that Sample 2> Sample 3> Sample 1> Sample 4. It is also observed that though the porosity is less in case of Sample 3, but as the thickness of substrate is lowest, showthrough is more as compared to Sample 1 and Sample 4.

Table 4.1: Porosity and Thickness of Showthrough Substrates

Sample	***Thickness (mm)***	***Porosity (sec)***
Sample1(Figure 4.6)	*0.058*	*52*
Sample 2(Figure 4.7)	*0.088*	*40*
Sample 3(Figure 4.8)	*0.055*	*47*
Sample 4(Figure 4.9)	*0.084*	*46*

4.5.2 Result and Discussion for Strikethrough

Like showthrough print defect detection, in strikethrough the same procedure is followed to identify the strikethrough pixels. In case of Strikethrough print defect identification, the substrates used for all the samples are same and the liquid ink is used. The porosity of strikethrough sample is 6 sec and the thickness is 0.059mm.

In Figure 4.10, 4.11, 4.12 and 4.13 are some of the defected samples with their pictorial representation of defect detection is depicted. The results show the strength of proposed method of strikethrough defect detection.

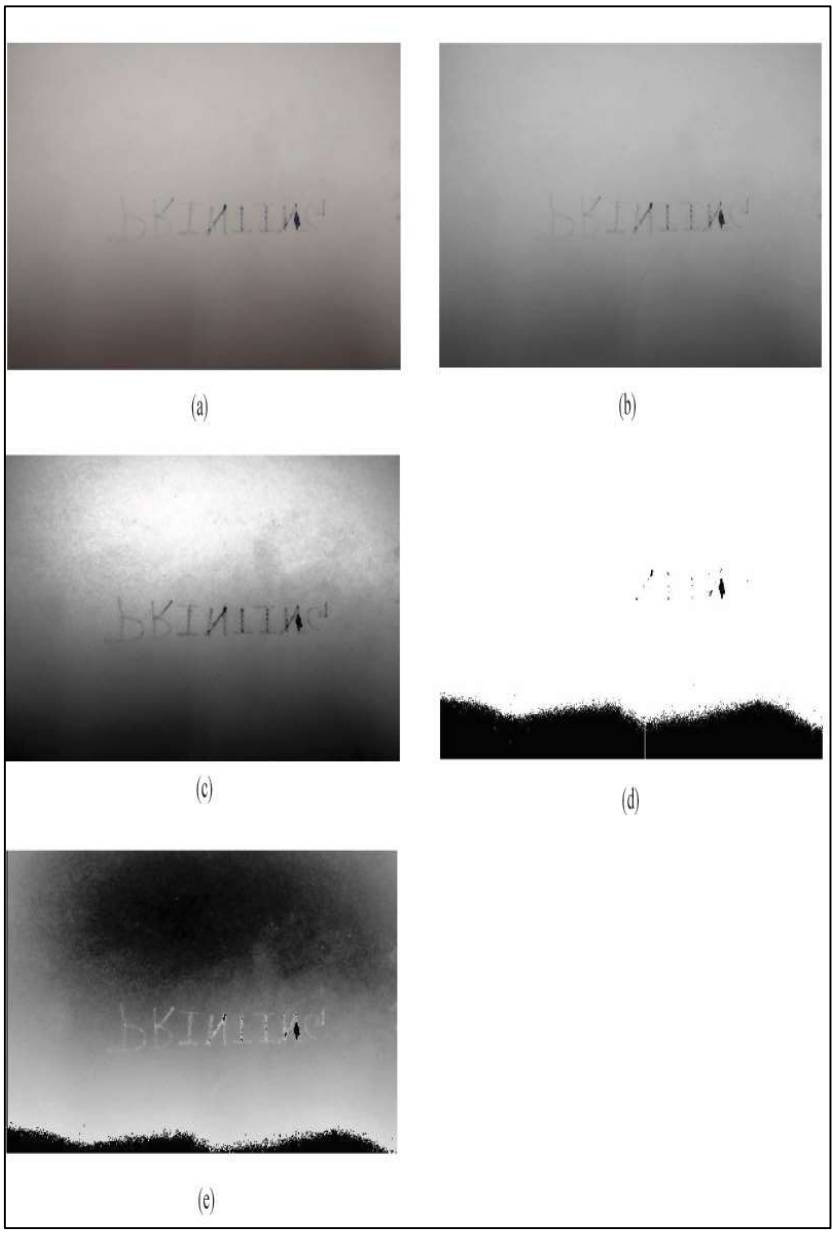

Figure 4.10 Resulted Images of Strikethrough (Sample 1) (a) Original image (b) RGB to Gray Converted image (c) Histogram Equalized Image (d) Foreground Image (e) Detected Strikethrough Pixels.

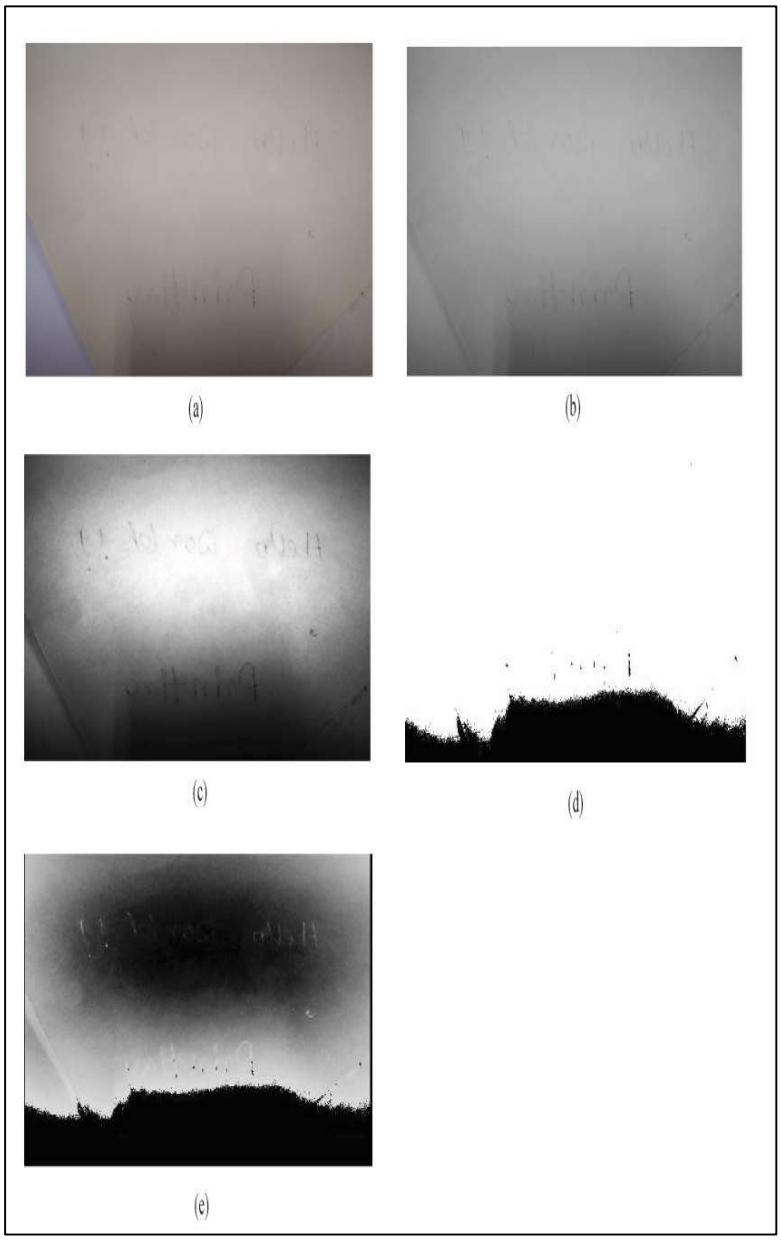

Figure 4.11 Resulted Images of Strikethrough (Sample 2) (a) Original image (b) RGB to Gray Converted image (c) Histogram Equalized Image (d) Foreground Image (e) Detected Strikethrough Pixels.

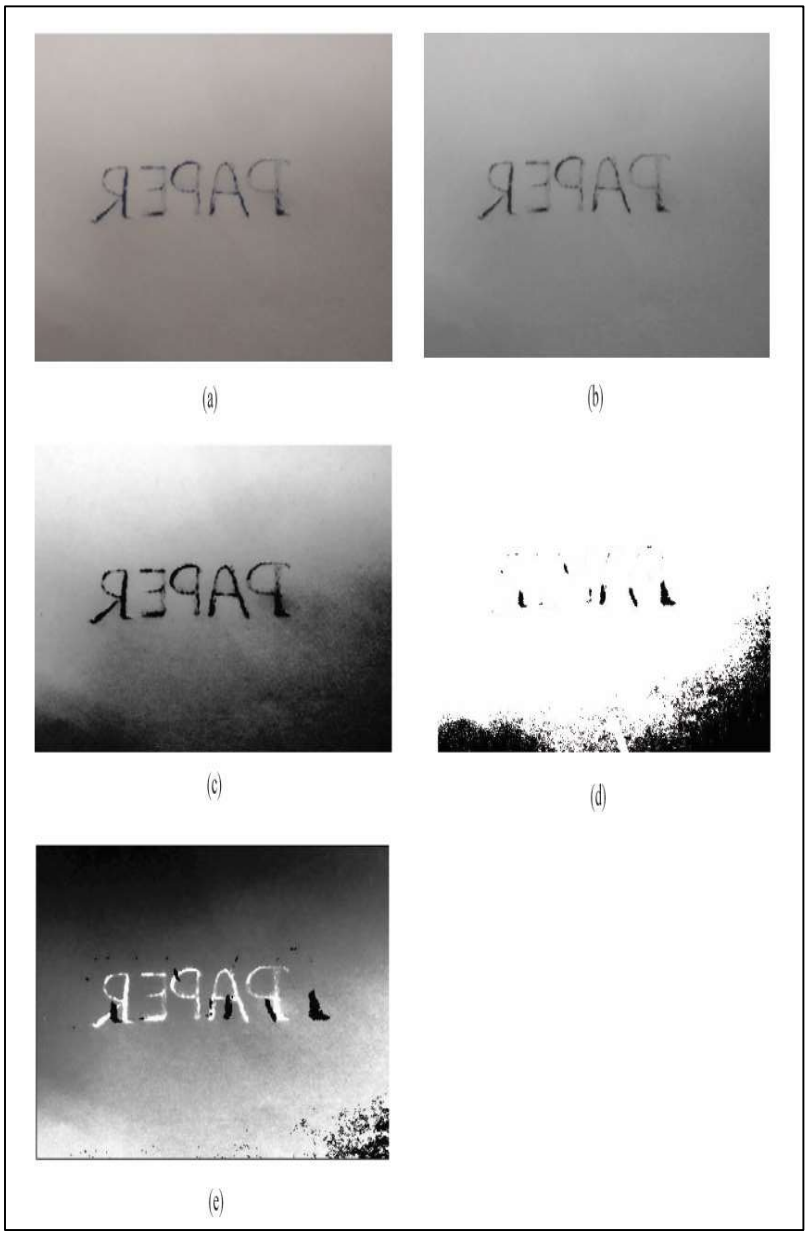

Figure 4.12 Resulted Images of Strikethrough (Sample 3) (a) Original image (b) RGB to Gray Converted image (c) Histogram Equalized Image (d) Foreground Image (e) Detected Strikethrough Pixels.

Figure 4.13 Resulted Images of Strikethrough (Sample 4) (a) Original image (b) RGB to Gray Converted image (c) Histogram Equalized Image (d) Foreground Image (e) Detected Strikethrough Pixels.

In Figure 4.10 (a) the original digital image of strikethrough sample 1 is shown and it is captured using mobile camera only. Here the ink penetrated through the paper fibre and became visible in the foreground of backside image. The intensity of the strike through pixels is more compare to showthrough pixels which facilitates the detection procedure. However, the

presence of other unwanted printing problems cannot be ignored and must be removed by post processing operations. In this sample presence of shadow made the detection of strikethrough pixels a little bit difficult. In Figure 4.10(d) in foreground image, the portion of ink that bleeds through the paper is detected. Finally in Figure 4.10(e) the strikethrough pixels in background are shown in white is identified.

The next sample in Figure 4.11 (a) (sample 2), because of illumination condition it is very difficult to identify strike through pixels. Here histogram equalization is required to increase the contrast of background strikethrough pixels for proper detection. The histogram equalized image is shown in Figure 4.11(c). Here also the shadow part degrades the image and can be removed by further image processing operations. Figure 4.11(d) shows the foreground image and Figure 4.11(e) shows the strikethrough pixels detected in background.

The sample in Figure 4.12 (a) (sample 3) shows that the strikethrough pixels are clearly visible. The foreground image in Figure 4.12(a) shows the penetrated strike through pixels from reverse side of the paper. To segregate the strike through pixels from the background of image, a proper choice of threshold value is very important otherwise foreground pixels can be identified as strikethrough for those areas where intensity values of print pixel will be nearly similar to strikethrough. The threshold value is selected by maximizing the class variance of the pixel of the entire image. Pixels with a gray level greater than the global optimal threshold value are considered as foreground pixels. The final result in Figure 4.12(e) shows the detected strikethrough pixels in background only and depicts the strength of identification of strikethrough pixels.

The sample 4 in Figure 4.13 (a), ink mostly penetrates from the reverse side and is detected in the foreground image in Figure 4.13 (d) after thresholding operation. Ideally the foreground image should be blanked but here the ink penetrated through the fibre and appears in foreground. The strikethrough pixels detected in white are shown in Figure 4.13 (e).

4.6 Conclusion

Quality of image can be degraded for many reasons. Printing defect is one of the most important among them and should be taken care of. Showthrough and strikethrough printing defects are very much common in offset presses. There are several reasons behind this printing defects. Substrate properties and ink characteristics are most important factor in this context.

Sometimes the desired product outcome might affect as showthrough and strikethrough pixels merged with the background and as well as foreground image. Therefore, it is very much important to detect and take the necessary step to prevent these printing defects. Generally, the detection procedure is done manually in presses which make the job dependent on the man's ability of detection. Also, manual process is so much time consuming. Therefore, it becomes a popular research direction to detect these defects with the help of image processing tool that can be less manual and less time consuming. A common method of detection is applied for both these two printing defects as the nature of these defects are almost same. The presented work is comparatively simple in terms of computational load and cost effective too. Therefore, this automated approach is much better than manual intervention to get the better-quality print job and helps to save the production time. For this experimental work, mobile camera is used which is now-a –days very much available to everyone and handy as well. This makes the procedure less expensive in terms of production cost. A major limitation of this study is appearance of scummimg, tinting and other printing defects that can hinder the process and make the detection a little bit difficult. Post processing operations can be a solution to overcome this problem. Exploring presented work in different device independent color space can be further extended study of this thesis work. Apart from contrast enhancement histogram equalization, other histogram equalization technique can be applied to get better result. This study focused only to detect the showthrough and strikethrough pixels while reconstruction of the image removing the showthrough and strikethrough pixels is probably most important follow up work in future.

CHAPTER - 5

Identification and Evaluation of Doubling Printing Defect using Image Quality Metrics

CHAPTER- 5

Identification and Evaluation of Doubling Printing Defect using Image Quality Metrics

5.1 Introduction

Doubling is one of the most frequently encountered defects in printing processes and is a critical quality issue that significantly impacts the visual integrity and functionality of printed materials. This defect is usually visualized as a secondary, slightly displaced same image with lower density [2] or shadow appearing adjacent to the actual printed image. The visual result is a blurred [3] or ghosted impression compromising the sharpness and clarity, directly impacting the quality of the final printed product. Figure 5.1 and 5.2 depicted two such printed sample image with doubling print defect.

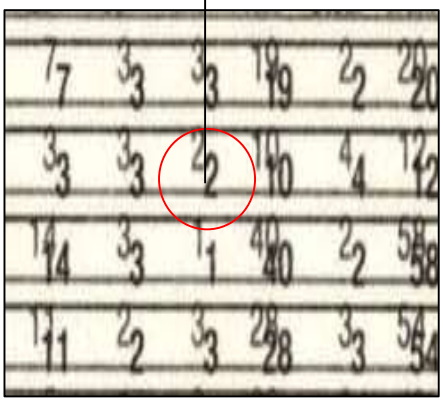

Figure 5.1 Sample Image of Doubling Printing Defect

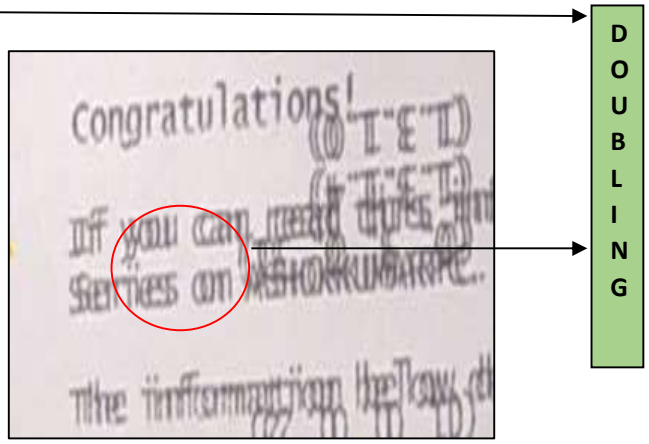

Figure5.2 Sample Image of Doubling Printing Defect

This defect is not confined to any single printing method. It can occur across a variety of processes, including offset printing, flexographic printing, and even digital printing, making it a universal challenge in the field. When a faint duplicate impression occurs with actual printed image in doubling it appears as excessive dot gain [3]. In offset printing process, some of the ink from dots printed by previous unit is transferred from paper to blanket of the next printing unit and consequently transferred onto the succeeding press sheets [106]. This transferred ink is not in register with the actual true full impression dots [106]. The consequences of this phenomenon extend beyond visual imperfections. It often results in the rejection of defective prints, causing material wastage and increasing production costs. Furthermore, for commercial printers and manufacturers, persistent quality issues like doubling can erode customer trust and damage brand reputation.

In doubling dot area increases linearly with elongation [112] and a non-directional extra image is created [113]. If seen under magnification doubling can be identified as the same image printed with lower density with a displacement from the actual printed image [2].

In offset printing, doubling defects can occur due to a range of mechanical issues, including improper roller settings, worn-out gears [106], vibrations in the press, or misaligned and damaged impression cylinders, lack of paper flatness in paper surface [106]. These defects result from the premature or incorrect contact of the substrate with the printing blanket or cylinders, leading to faint duplicate impressions that appear out of register with the intended image. Moreover, Variability in the position of substrate as it passes through the next -in-line printing unit [112] is another cause for this type of phenomenon. Pre- kissing in Offset printing

happens when the paper prematurely contacts the printing blanket, often due to static electricity or deformed paper caused double impression [2]. Environmental factors, such as humidity and temperature, also play a role. For example, excess humidity may cause ink smudging or paper deformation. Dimension instability of paper due to absorption of moisture content can lead to this type of defect [2]. Using low-quality or inappropriate paper can enhance the issue. Doubling can also occur when substrate comes into contact with the blanket twice, creating a double impression. When the blanket pulls part of an already printed image off the sheet and then transfer this image slightly out of register onto the next sheet passing under the blanket, doubling problem occurs. However, this kind of defect is undesirable and can be avoided by the regular maintenance of rollers, blanket, and drums. Replace worn-out components as needed.

5.2 Sample Preparation

For this experimental work the printed samples are collected from different offset printing presses. All sample used for this study is real-life sample. Image acquisition of these samples is done by using mobile camera with 48-megapixel resolution and in D65 illumination. The whole camera setup is made with a simple handmade approach shown in Figure 3.3, where camera is placed in the top and the sample is placed right under the camera. The distance between camera and samples are same for each tested sample.

5.3 Problem Statement

In the printing industry, the identification of doubling print defects has traditionally been a manual process. This reliance on human judgment makes the procedure subjective and prone to variability, as it depends heavily on the operator's ability to recognize the defect. The identification process can differ significantly from one individual to another, leading to inconsistent evaluations. Furthermore, there is no standardized methodology to quantify the degree of similarity except eye assessment between the actual image and the faint duplicate.

Operators typically rely on their own experience and visual estimation to decide on corrective actions at the press unit. However, this lack of standardization and objective measurement introduces challenges in maintaining consistent quality across production runs. Moreover, doubling defects often include overlapping regions where the actual and duplicate images

blend, making it particularly challenging to distinguish the defect with accuracy by image processing techniques.

To address these challenges, the Structural Similarity Index Measure (SSIM) algorithm has been employed as a tool for detecting and identifying doubling defects. The SSIM algorithm provides a mathematical framework for comparing the structural similarity between two images, enabling a more objective and quantitative assessment of the defect. This method enhances precision by analyzing luminance, contrast, and structural information in the image. However, the application of SSIM in real-world offset printing scenarios introduces additional complexities. Test samples from offset presses often contain other print defects, which can interfere with the accurate detection and quantification of doubling defects. These extraneous issues create noise in the data, which complicates the identification process.

Despite these complexities, the integration of the SSIM algorithm into quality control processes represents a significant advancement in the printing industry. The use of computational algorithms like SSIM also provides the opportunity to standardize the identification and quantification of doubling defects, paving the way for more reliable and reproducible quality assurance practices. While manual identification of doubling defects has been the conventional approach in the printing industry, but the automated approach with SSIM algorithm offers a more accurate identification and quantification despite of presence of other printing defects.

Here in this study along with SSIM algorithm another two print quality metrics: Mean Square Error (MSE) and Feature Similarity Indexing Method (FSIM) algorithm are also applied for comparative study and to prove the strength of print quality metrics for detecting the doubling print defect.

5.4 Proposed Method

Figure 5.3 illustrates the flowchart [114] of the presented method for identifying and quantifying doubling print defects. The process begins with the collection of samples from a diverse range of offset commercial presses, ensuring the inclusion of various printing conditions and substrates. Both text-based and image-based prints are considered for the analysis. A comparative study is conducted using print quality assessment metrics, including Mean Square Error (MSE), Feature Similarity Indexing Method (FSIM), and the Structural Similarity Index Measure (SSIM) algorithm.

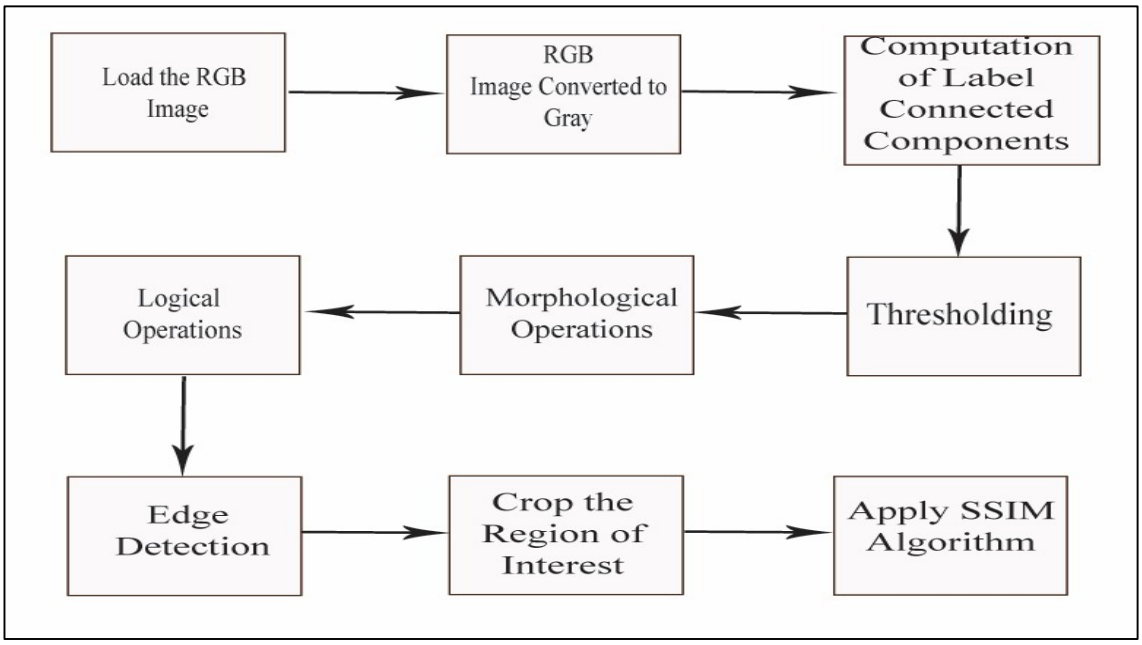

Figure 5.3 Flow chart of Proposed Method of Doubling Printing Defect Detection

5.4.1 Image Conversion

The defective printed samples are first digitized using a mobile camera (48MP) in a handmade setup (Figure 3.3) environment. Subsequent steps for identifying doubling printing defects are performed using MATLAB programming software. After capturing the image of defective samples, the image is converted from the RGB color space to the grayscale color space.

The choice between using grayscale or RGB color space for image processing depends on the specific requirements of the task. In this case, grayscale is preferred because it has only one channel representing intensity or brightness, simplifying image processing operations and reducing computational load. The intensity values in a grayscale image range from 0 (black) to 1 (white), [103] with fractional values in between. Since tasks like identifying printing defects often rely on intensity rather than color information, grayscale images are better suited for this purpose. Moreover, this simplifies image processing operations, reduces computational complexity, and enhances efficiency. Unlike RGB, which is device-dependent [110], grayscale images do not vary across devices. RGB values can differ based on the manufacturer or device specifications, potentially introducing inconsistencies in processing. RGB images consist of three channels—Red, Green, and Blue—used for representing color information.

Transforming an image from RGB to grayscale can be done using a weighted sum of the color channels. The weights reflect the human visual system's sensitivity to different colors :

- Red contributes 29.9% to luminance,
- Green contributes 58.7% to luminance,
- Blue contributes 11.4% to luminance.

The commonly used formula for conversion is:

$$Gray = 0.299*Red + 0.587*Green + 0.114*Blue \qquad \text{eq. 5.1}$$

The weights (0.299, 0.587, 0.114) are derived from luminance values as perceived by the human eye, with the green channel contributing the most due to higher sensitivity in this spectrum. This ensures the resulting grayscale image accurately represents the intensity levels while discarding unnecessary color information. Therefore, using gray color space simplifies the computations, minimizes resource usage while ensuring the consistency. It is quite easy to perform image processing operations like edge detection, segmentation of image, smoothening, sharpening [103] etc. in gray color space rather than RGB color space.

5.4.2 Computation of Label Connected Components

The next step involves computing the pixel connectivity to address the issue of doubled print defects, where overlapping impressions of the print occur. This defect results in regions where one impression overlaps another, creating challenges in distinguishing between distinct and overlapped areas. To tackle this, 8-connectivity [103] of pixels is used for edge detection and labelling.

Pixel connectivity refers to the relationship between a given pixel and its neighbouring pixels, which can be evaluated based on certain connectivity rules. It is a fundamental concept in image processing, particularly in tasks such as edge detection, segmentation, region labeling, and object identification. Pixel connectivity defines how pixels are considered connected based on their spatial relationship and intensity values. Generally, there are three types of connectivity rules: 4-connectivity, 6-connectivity and 8-connectivity. In this study 8-connectivity [103] is chosen to detect edges and overlaps in all direction- horizontal, vertical and diagonal.

In 8- connectivity [103] a pixel is connected to its eight neighbours which provides more comprehensive connectivity than others. For an example here pixel (X) is connected to all adjacent pixels (1's) in every direction.

8-Connectivity 

1	1	1
1	X	1
1	1	1

The use of 8-connectivity for pixel labeling is essential to ensure that no overlapped regions are missed as because diagonal pixels are considered. By analysing the labeled connectivity of pixels, it becomes possible to identify areas where the defect occurs. The method effectively distinguishes between well-printed regions and those affected by doubling defects. As pixels are grouped into distinct regions based on their connectivity, tasks like object segmentation to identify the doubling printing defect simplifies.

5.4.3 Background Extraction

The next step involves with background and foreground segmentation with adaptive thresholding algorithm. This step is needed to separate the background from the foreground for further analysis or processing. It is useful in tasks like masking, segmentation, or measuring the area covered by the background. To apply the print quality metrics and to check the similarity between the double printed regions it is necessary to segment the foreground and background images.

5.4.4 Morphological Operation

After segmentation morphological operations [103] [111] has been performed where small gaps, holes or breaks are filled in edges using morphological closing with a linear structuring element. It combines dilation followed by erosion, effectively smoothing the edges and connecting nearby components. When using a linear structuring element, the closing operation emphasizes filling gaps along a specific direction. Dilation expands the edges outward, bridging gaps and filling small holes whereas erosion shrinks the expanded edges back to their

original size, removing any extension added during dilation. Therefore, when these two operations combined together as a result small gaps or discontinuities in the edges are filled while maintaining overall shape and size of the image.

A linear structuring element specifies the size and orientation of the gap-filling operation. It allows filling gaps along a specific direction, useful for images with oriented patterns. Moreover, length and angle can be adjusted to suit specific edge characteristics. Here for this study a morphological closing operation on the detected edges using a linear structuring element with a length of 10 and an angle of 5 degrees is applied.

5.4.5 Logical Operation

After performing morphological operation bitwise XOR [103] [111] between the binary image and the closed edges is performed to identify regions by removing small and irrelevant features. In a bitwise XOR operation, for each pixel:

- The result is 1 if the two corresponding pixel values are different (0 and 1 or 1 and 0).
- The result is 0 if the two corresponding pixel values are the same (0 and 0 or 1 and 1).

After logical operations the result of XOR is then processed with a function, 'bwareaopen', commonly found in Matlab software libraries, which removes connected components smaller than 1000 pixels. This step eliminates noise or small, insignificant regions.

5.4.6 Edge Detection

The next step follows the edge detection operation of the grayscale image using Prewitt [103] [111] operator. The Prewitt operator is a gradient-based edge detection method used in image processing which identifies edges by calculating the gradient of image intensity, highlighting areas where there are sharp changes in brightness. Application of Prewitt operator is easy as well as fast and effective for detecting edges in images with well-defined intensity transitions. The Prewitt operator generally uses 3x3 convolution kernels to approximate the gradient in the horizontal (G_x) and vertical (G_y) directions:

$$\text{Horizontal kernel } (G_x): \begin{matrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{matrix}$$

Vertical kernel (G_y) : $\begin{matrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{matrix}$

These masks are convolved with images to calculate horizontal and vertical gradients at each pixel.

The magnitude of the gradient at each pixel is computed by combining the horizontal and vertical gradients to find the edge strength.

Magnitude= $\sqrt{G_x^2 + G_y^2}$ eq. 5.2

Edge Direction(θ)= arctan $(\frac{G_y}{G_x})$ eq. 6.3

After that a thresholding is applied to the gradient magnitude to retain only significant edges. After the edge detection, the region of interest from double printed edges are cropped. Here the two edges of double printed area are taken to check the structural similarity. Print quality metrics like Structural Similarity Measurement (SSIM), Mean Square Error (MSE), Feature Similarity Indexing Method (FSIM) is applied to measure the similarity between the double printed edge.

5.4.7 Print Quality Assessment Metrics:

This study focuses on evaluating print quality using three key assessment metrics. After segmenting the background and foreground in the image, the region of interest (ROI) from the double-printed edges is carefully cropped. The two edges within the double-printed area are analysed to assess their structural similarity.

To achieve this, the following quality metrics are applied, ensuring a comprehensive evaluation of print consistency:

1. **Structural Similarity Index Measurement (SSIM)**: Measures perceptual similarity between the edges, capturing subtle differences in luminance, contrast, and structure [69] [71] [73].
2. **Mean Square Error (MSE)**: Provides a quantitative measure of the pixel-wise error between the two edges, reflecting overall similarity [71].
3. **Feature Similarity Indexing Method (FSIM)**: Focuses on feature-level analysis, leveraging gradient magnitude and phase congruency to assess structural alignment [14] [71].

The use of these metrics ensures a robust and detailed evaluation of print quality. Below, the detailed explanation of each metrices are discussed.

5.4.7.1 Structural Similarity Index Measurement

The Structural Similarity Index Measure (SSIM) [69] [71] [73] is a popular algorithm used to evaluate the perceived quality of images. It is widely used in image processing to compare the similarity between two images, typically for assessing the quality of a compressed or processed image relative to the original. Unlike traditional metrics, SSIM measures image quality based on how the human visual system perceives structure, rather than just pixel differences.

SSIM carries the structural information of pixels which are closed spatially. SSIM is calculated between two common size (N X N) images x and y.

$$\mathrm{SSIM}(x,y)=\frac{2(\mu_x\mu_y+c_1)\,(2\sigma_{xy}+c_2)}{(\mu_x^2+\mu_y^2+c_1)(\sigma_x^2+\sigma_y^2+c_2)} \qquad \text{eq. 5.4}$$

Where, μ_x = average of x;

μ_y = average of y

σ_x^2 = variance of x

σ_y^2 = variance of y

σ_{xy} = covariance of x and y

$c_1 = (K_1\,L)^2\,,\ c_2 = (K_2\,L)^2$

L= dynamic range of the pixel values.

k1 = 0.01 and k2= 0.03 by default.

Image distortion between the sample images can be modeled by SSIM by comparing three factors: contrast distortion(c) , luminance distortion (l), and loss of structural correlation(s).

$$l(x,y)=\frac{2(\mu_x\mu_y+c_1)}{(\mu_x^2+\mu_y^2+c_1)} \qquad \text{eq. 5.5}$$

$$c(x,y)=\frac{2(\sigma_x\sigma_y+c_2)}{\sigma_x^2+\sigma_y^2+c_2} \qquad \text{eq.5.6}$$

$$s(x,y)=\frac{\sigma_{xy}+c_3}{\sigma_x\sigma_y+c_3} \qquad \text{eq.5.7}$$

Luminance (l(x,y)): Compares the brightness of the images.

Contrast (c(x,y)): Measures the contrast differences of the images.

Structure (s(x,y)): Evaluates the structural correlation of the image pixels.

In this presented study, the percentage of structural similarity is measured between double-printed edges to analyze and detect doubling defects. Doubling defects occur when printed edges are overlapped due to double impressions. This issue necessitates comparing the structural similarity between the overlapped region of the image. By quantifying this similarity, the defect can be detected and assessed based on the overlap and distortion caused by the double impression. SSIM is sensitive to perceptual changes, such as blurring or contrast changes. Therefore, it is applicable for image compression quality assessment. Moreover, for image denoising and restoration evaluation SSIM is applied. The main limitation of SSIM algorithm is its sensitivity to small shifts or rotation in images. SSIM is not effective when comparing images of different resolutions.

5.4.7.2 Mean Square Error

The Mean Square Error (MSE) is a widely used statistical measure in print quality assessment. It quantifies the average squared difference between reference (ideal) values and observed (actual) values. Generally, MSE measures the error between the images where the average of squared errors is represented by MSE [71].

$$\text{MSE}= \frac{1}{\text{MN}}\sum_{n=0}^{m}\sum_{n=1}^{m}[\hat{g}(n,m)-g(n,m)]^2 \qquad \text{eq. 5.8}$$

Where:

- $g(n,m)$: Reference value
- $\hat{g}(n,m)$: Observed or printed value.
- MN: Total number of data points.

MSE is used to evaluate deviations in color reproduction by comparing the printed colors to reference values [115]. Here in this study MSE is calculated between the cropped region of detected double printed image to check the similarity.

5.4.7.3 Feature Similarity Indexing Method

The Feature Similarity Indexing Method (FSIM) [14] [71] is a perceptual image quality assessment metric designed to evaluate the similarity between two images by focusing on low-level visual features. Phase congruency and Gradient magnitude are the two most important key components to be followed to measure the characteristics of images.

5.4.7.3.1 Phase Congruency

Phase Congruency (PC) is a powerful feature descriptor used in image processing and computer vision to identify important image features such as edges, lines, and corners. It provides a way to analyse the structure of an image based on the alignment of phase information across multiple frequency components. One of its characteristics is that it is invariant to lighting variation in an image [14]. A signal can be represented using its frequency components, each characterized by an amplitude (strength) and phase (alignment). The phase of a frequency component specifies the position of the wave relative to a reference point, while amplitude measures its contribution to the signal. Phase Congruency (PC) captures edge and corner information in the image. It measures the alignment of phase across different frequency components, which correlates strongly with human visual perception. It is computed as [72]:

$$PC(x) = \frac{\sum_n W_n . A_n(x) . \cos\left(\emptyset_n(x) - \emptyset_{mean}(x)\right)}{\sum_n A_n(x) + \epsilon} \qquad \text{eq. 5.9}$$

Where:

- n: Index of the frequency component.
- $A_n(x)$: Amplitude of the n-th frequency component at pixel x.
- $\emptyset_n(x)$: Phase of the n-th frequency component at x.
- $\emptyset_{mean}(x)$: Weighted mean phase at pixel x.
- W_n: Weighting function to emphasize meaningful components.
- ϵ: A small constant to avoid division by zero.

Since PC depends on phase alignment rather than amplitude, it works well across images with different contrast or illumination makes invariant to contrast [71]. PC effectively detects edges, corners, and textures, focusing on regions with significant structural information. Moreover, for image quality assessment metrics FSIM Phase Congruency is used to evaluate perceptual quality.

5.4.7.3.2 Gradient Magnitude

Gradient magnitude measures the horizontal and vertical gradient of images unlike phase congruency. Convolution masks are used to compute the gradient components by filtering the image. Common operators include: Sobel operator, Prewitt operator, Scharr operator Visual FSIM [14] [71]. Gradient Magnitude of image F(x) can be represented as: [14]

$$GM= \sqrt{G_x^2 + G_y^2} \qquad \text{eq. 5.10}$$

Where, Gx ,Gy is horizontal and vertical gradient of image F(x).

GM emphasizes regions where human vision is naturally more sensitive, such as areas with sharp transitions. While PC is invariant to contrast and emphasizes feature localization, GM reflects contrast and texture changes, offering a balanced assessment.

Here in this study FSIM is calculated depending on phase congruency and gradient magnitude between two cropped regions of interest where double printing is occurred. If the phase congruency derived from two images are PC_1 and PC_2 and gradient magnitude are G_1 and G_2, then the similarity based on phase congruency calculated as [71]:

$$S_{pc} = \frac{2\, PC_1\, PC_2 + T_1}{PC_1^2 + PC_2^2 + T_1} \qquad \text{eq. 5.11}$$

Where, T_1 is a positive constant to increase the stability of S_{pc}

Likely gradient magnitude is calculated as:

$$S_G = \frac{2\, G_1\, G_2 + T_2}{G_1^2 + G_2^2 + T_2} \qquad \text{eq. 5.12}$$

Where, T_2 is a positive constant depending on the gradient magnitude values dynamic ranges.

Next S_{pc} and S_G are combined to calculate the similarity between the images:

$$S_L(x) = \left[S_{pc}(x)\right]^{\alpha} . [S_G(x)]^{\beta} \qquad \text{eq. 5.13}$$

Here, α and β are parameters used to control the relative importance of phase congruency and gradient magnitude features.

5.5 Result and Discussion

The proposed method [114] is extensively tested on a diverse set of samples collected from commercial offset presses, with a primary focus on newsprint. To ensure a cost-effective and automated approach, a 48-megapixel mobile camera is utilized within a custom-built experimental setup (Figure 3.3) illuminated under standardized D65 lighting conditions. The use of a mobile camera highlights the practicality and affordability of this detection method, making it accessible for widespread implementation.

Here, the results and analysis of the tested images is presented, showcasing the effectiveness and reliability of this approach. Figure 5.4, 5.5, 5.6, 5.7, 5.8 and 5.9 illustrate the detailed results for four representative samples, encompassing both text-based and image-based prints.

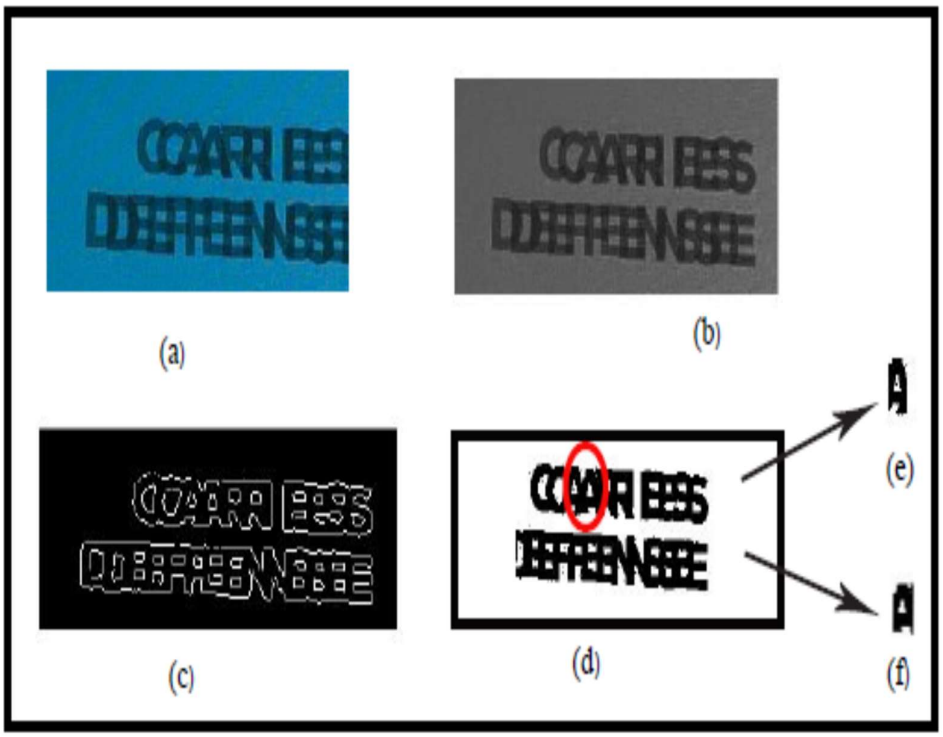

Figure 5.4 Resulted Images of Doubling Defect Detection (Sample 1)
(a) Original Print Image (b) Gray Scale Image (c) Foreground Image
(d) Detected Edge of Double Print Text (e) ROI-1 (f) ROI-2.

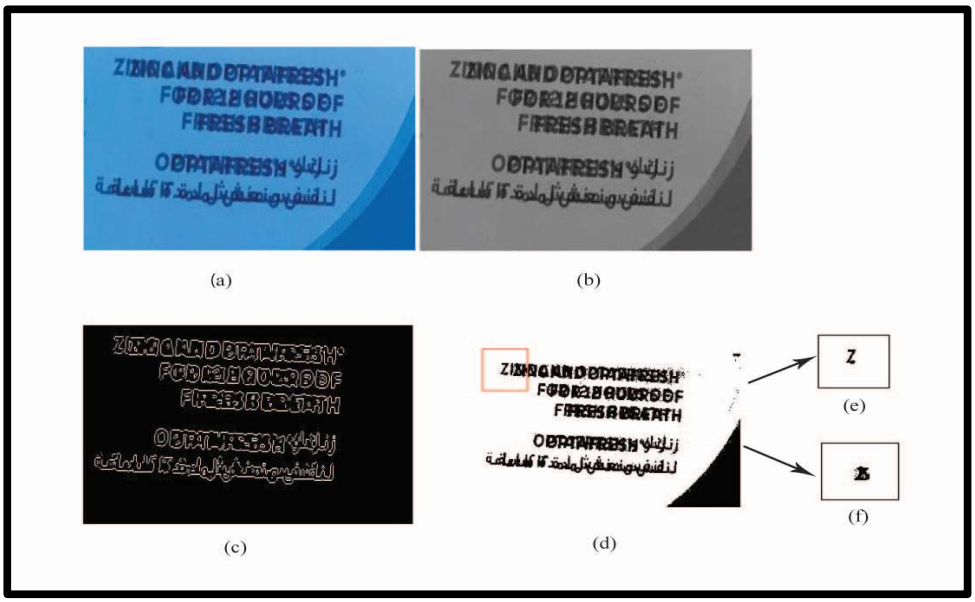

Figure5.5 Resulted Images of Doubling Defect Detection (Sample 2) (a) Original Print Image (b) Gray Scale Image (c) Foreground Image (d) Detected Edge of Double Print Text (e) ROI-1 (f) ROI-2.

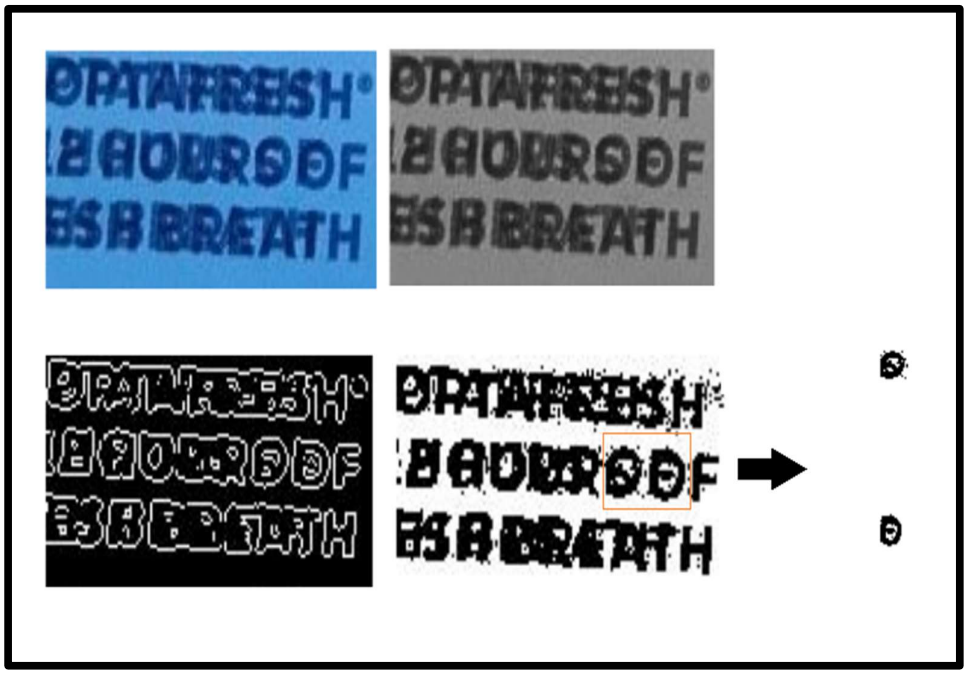

Figure 5.6 Resulted Images of Doubling Defect Detection (Sample 3) (a) Original Print Image (b) Gray Scale Image (c) Foreground Image (d) Detected Edge of Double Print Text (e) ROI-1 (f) ROI-2.

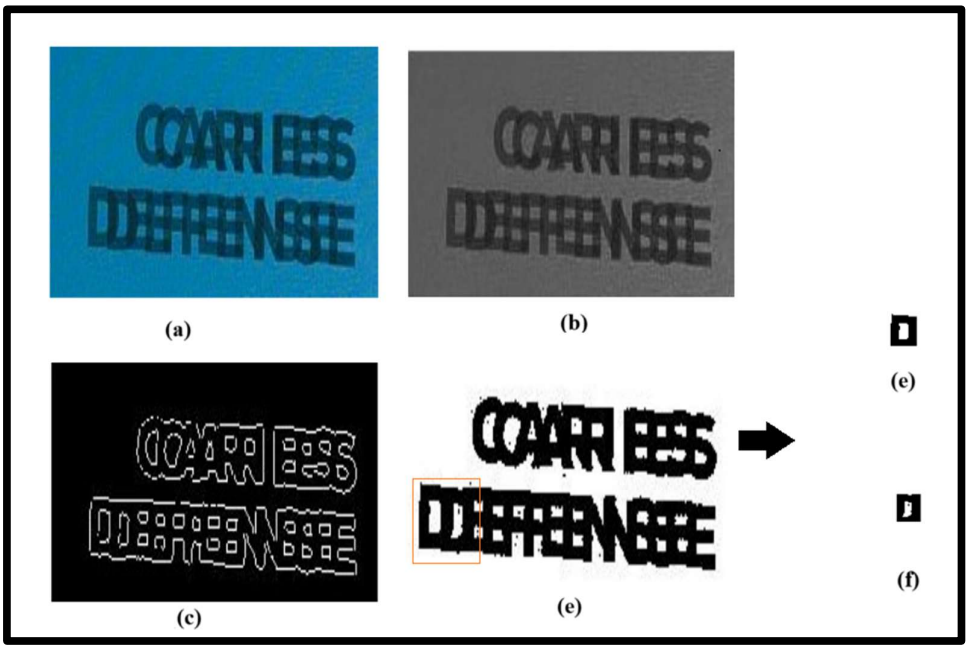

Figure5.7 Resulted Images of Doubling Defect Detection (Sample 4) (a) Original Print Image (b) Gray Scale Image (c) Foreground Image (d) Detected Edge of Double Print Text (e) ROI-1 (f) ROI-2.

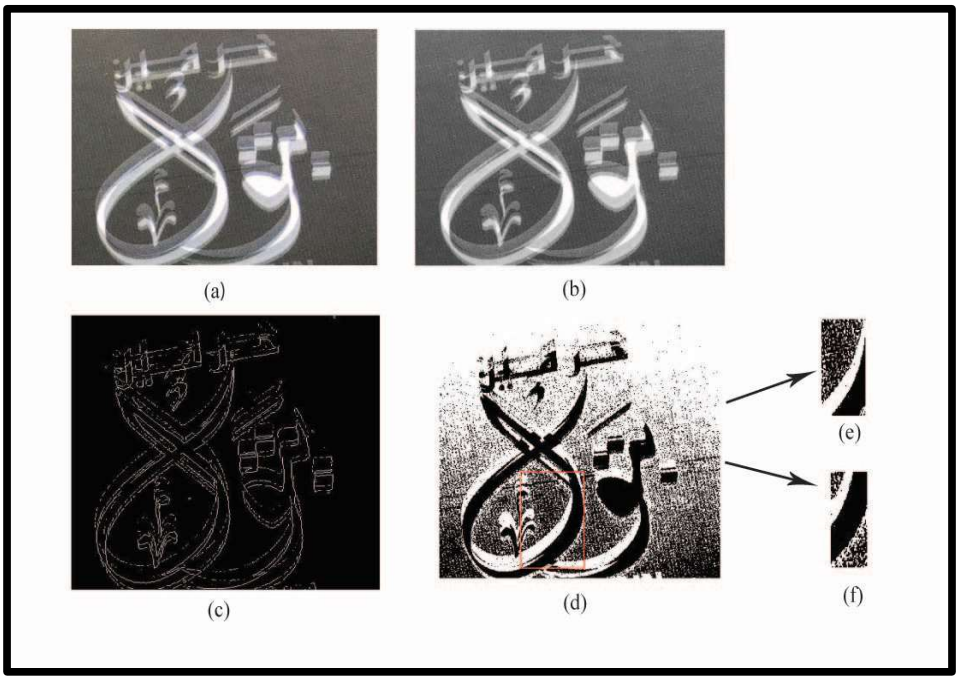

Figure 5.8 Resulted Images of Doubling Defect Detection (Sample 5) (a) Original Print Image (b) Gray Scale Image (c) Foreground Image (d) Detected Edge of Double Print Text (e) ROI-1 (f) ROI-2.

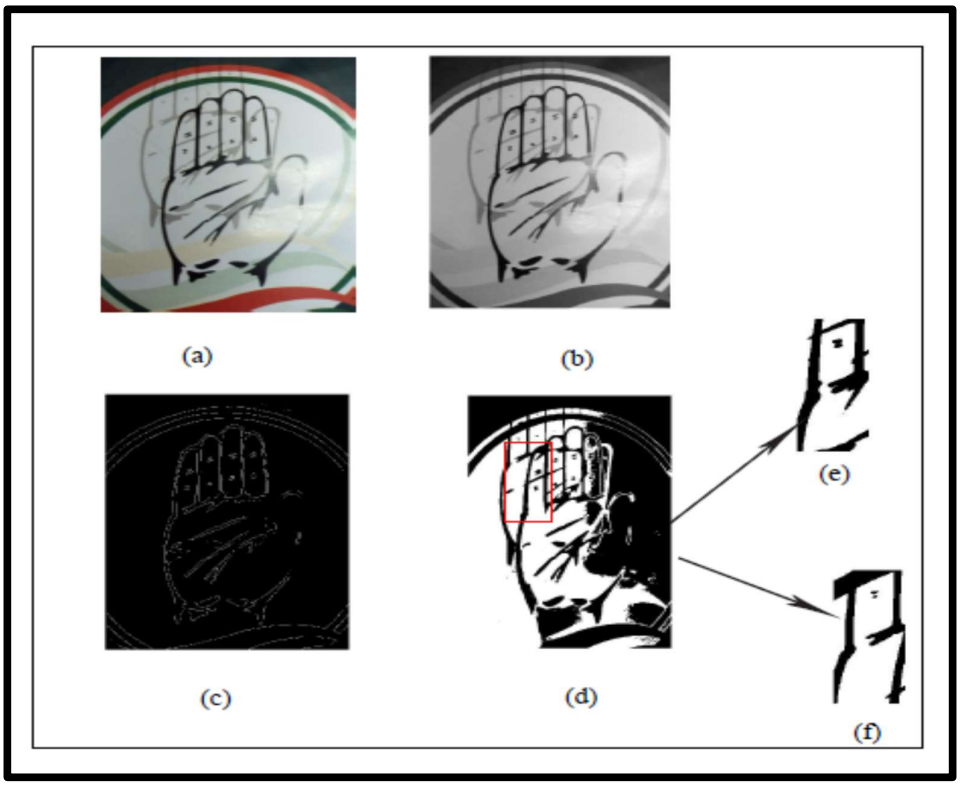

Figure 5.9 Resulted Images of Doubling Defect Detection (Sample 6) (a) Original Print Image (b) Gray Scale Image (c) Foreground Image (d) Detected Edge of Double Print Text (e) ROI-1 (f) ROI-2.

Figure 5.4, 5.5, 5.6 and 5.7 samples are text-based image. In Figure 5.4(a), the captured RGB image of a double-printed text sample is presented. To detect the edges of the double-printed area, the image is converted to grayscale, as shown in Figure 5.4(b). Adaptive thresholding is applied to segment the foreground, producing the image in Figure 5.4(c). Following morphological and logical operations, the detected edge of the doubling defect is displayed in Figure 5.4(d). Figure 5.4(e) and 5.4(f) show the cropped Regions of Interest (ROIs) extracted from the detected edges. These ROIs are compared using the Structural Similarity Index (SSIM) algorithm, Mean Square Error (MSE) and Feature Similarity Indexing Method (FSIM). In Figure 5.8, 5.9 other sample of image-based images are shown. The same procedure is followed for these samples also. Here also print quality metrics are applied to assess the similarity percentage into the cropped region of interests. Similarity percentage results are given in Table 5.1.

The similarity percentage calculated using the SSIM algorithm is proposed as a reliable metric:

- If the SSIM percentage is ≥90%, the tested ROIs are considered structurally identical; otherwise, they are identified as different. For the text-based sample, the structural similarity exceeded 95%, demonstrating high accuracy in detecting doubling defects. This highlights the potential of this method to outperform subjective measurement techniques. According to the results shown in Table5.1 Figure 5.4(e) and (f) the structural similarity percentage based on SSIM algorithm is 98%. For Figure 5.5(e) and (f) SSIM percentage is 97%. Another two samples (Figure 5.6 and 5.7), the tested ROI has similarity of 98% and 97% respectively. Whereas, the two cropped region in Figure 5.9 has less than 90 percent structural similarity based on contrast, luminance and structure

Mean Square Error (MSE) is one of the most commonly used tools for assessing image quality, as it represents the absolute pixel-wise error between two images. In this study, MSE is applied to the cropped Regions of Interest (ROIs) extracted from double-printed edges. The resulting value indicates the degree of error, with a value closer to zero signifying better similarity between the compared regions. However, MSE has limitations—it provides only a measure of absolute error and is not normalized, making it less informative for detailed assessments of image quality. In contrast, Structural Similarity Index (SSIM) and Feature Similarity Indexing Method (FSIM) offer more sophisticated evaluations, as they are normalized and focus on the perceptual features and structural integrity of the images. These metrics are particularly suited for tasks where repeatability of patterns, such as the edges of printed pixels, is crucial for detecting defects. The result depicted the same. According to the findings in table 5.1 in Figure 5.4. 5.5,5.6,5.7 and 5.8, FSIM supports the results got from SSIM algorithm whereas MSE gives only the absolute error between the cropped regions.

Figure 5.8 and 5.9 samples are image-based sample. For an image-based doubling defect, Figure 5.8(a) shows the original RGB image of the defected sample, while Figure 5.8(b) illustrates the corresponding grayscale image. After applying thresholding, the segmented foreground is presented in Figure 5.8(c), and the detected edge of the doubling defect is shown in Figure 5. 8(d). Figure 5.8(e) and 5.8(f) depict the cropped ROIs, which are analysed for structural similarity using SSIM, resulting in a similarity percentage of 90% and for Figure 5.9 sample structural similarity based on SSIM algorithm is 87 %. FSIM also supports this resulting fact. Due to the complex nature of overlapping edges in image-based doubling defects, the

accuracy of edge detection and similarity measurement is somewhat reduced compared to text-based samples. Despite this, the proposed method remains effective, particularly for quantitative assessments.

Table 5.1 *Result of MSE, FSIM, SSIM for Doubling Print Image Samples*

Sample	MSE	FSIM (in percentage)	SSIM (in percentage)
Sample 1 (Figure 5.4)	22.20	96	98
Sample 2 (Figure 5.5)	22.75	97	97
Sample 3 (Figure 5.6)	14.09	95	98
Sample 4 (Figure 5.7)	14.63	95	97
Sample 5 (Figure 5.8)	12.34	86	90
Sample 6 (Figure 5.9)	13.15	72	87

The results presented in Table 5.1 underscore these findings:

- MSE provides only a numerical measure of absolute error, which, while useful for basic error quantification, does not reflect the perceptual quality or structural alignment of the images. This limits its applicability for detecting doubling defects.
- FSIM outputs align well with the similarity percentages obtained from SSIM, confirming that FSIM supports feature-based similarity in the tested samples. However, its lack of sensitivity to contrast and luminance variations makes it less ideal for this specific application.
- SSIM demonstrates superior performance for doubling defect detection, as it accounts for all three factors—contrast, luminance, and structural correlation. This holistic approach ensures a more accurate representation of the similarity between double-printed edges, particularly in scenarios where precise detection is critical.

5.6 Conclusion

This paper presents a novel approach for identifying doubling print defects using the Structural Similarity Index (SSIM) algorithm. The proposed technique is rigorously tested on a variety of samples collected from offset presses, demonstrating its robustness and applicability. By implementing the detection process entirely within the MATLAB environment, the method significantly reduces manual intervention, making it a more efficient alternative to traditional subjective measurement techniques.

The key advantages of the proposed technique are, the automation and efficiency ensuring minimal manual effort and enabling faster detection of print defects. The approach lays the foundation for a mobile application, which could further streamline the defect detection, reducing the time required for problem identification and ensuring consistent production of high-quality prints. The incorporation of artificial intelligence offers exciting possibilities for improving defect detection and enabling advanced functionalities, such as automated correction of doubling defects.

The results validate the potential of the proposed approach, particularly for text-based images, where the SSIM algorithm excels in identifying structural similarities between double-printed edges. However, for image-based doubling defects, certain limitations are observed. Structural similarity measurement for overlapping edges in complex images remains a challenge. Moreover, the inability to achieve 100% similarity between edge patterns highlights the need for further refinement, potentially through post-processing operations. Despite these limitations, the effectiveness of SSIM is clearly established when compared to other image quality assessment metrics, such as Mean Square Error (MSE) and Feature Similarity Indexing Method (FSIM). Unlike these alternatives, SSIM provides a more perceptually aligned evaluation by factoring in contrast, luminance, and structural variations of double-printed edges. In conclusion, the SSIM-based approach offers a more reliable method for detecting doubling print defects, outperforming traditional techniques. The work opens places for future research, including the method of removing double printed text or images by reconstructing the original image. Also, the usage of artificial intelligence can make the doubling defect detection procedure more accurate with automated defect correction optimizing the overall cost of printing process and enhancing the quality of printed product.

CHAPTER - 6

Identification and Evaluation of Misregistration Printing Defect

CHAPTER-6

Identification and Evaluation of Misregistration Printing Defect

6.1 Introduction

Misregistration can occur during the print production of various applications such as newspapers, magazines, book printing, and package printing. Misalignment or improper superimposition of colors during multicolour printing [107] can result in print misregistration. While it can happen in any printing process, it is more likely in lithography due to the presence of the water fountain solution [107]. This can result in distorted or blurred prints that affect the visual presentation as well as quality of the final product [2]. Mostly in printing presses registration marks is verified by using linen glass to check the misalignment of colors. Figure 6.1(a) and Figure 6.1(b) depicts the perfectly registered and misregistered image of registration marks.

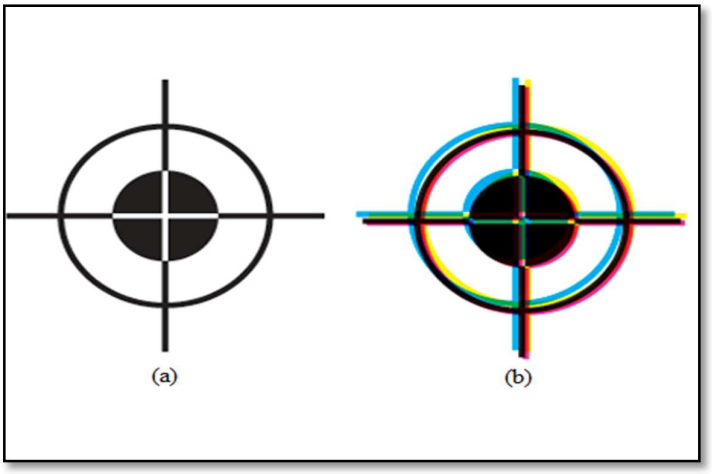

Figure 6.1 Registration Marks (a) Perfect Registration; (b) Misregistration

Paper exhibits a characteristic known as hysteresis [4], meaning its dimensions is related to its history or previous conditions like storage environment or conditions. The hysteresis nature of paper refers to the phenomenon where the material's properties, particularly its dimensions (such as thickness, length, and width), depend not only on the current environmental conditions (like humidity or temperature) but also on its past conditions. When paper absorbs or loses moisture, its dimension changes, but this process is not completely reversible. The response of the paper to changes in humidity or moisture content, is different when these atmospheric conditions are increasing or decreasing, creating a lag or hysteresis effect.

For example, if the humidity increases, the paper may expand, but when the humidity decreases, the paper does not immediately return to its original dimensions. The difference in behaviour during these cycles is the hysteresis effect. This characteristic is important in the paper industry because it can influence the dimensional stability and accuracy of printing, storage, and handling, especially in environments with fluctuating humidity levels [4].

There are several reasons for the changes in the dimension of the paper. One of them is moisture imbalance between pressroom and paper [106]. According to the relative humidity in air, paper may shrink or expanded. Therefore, relative humidity of the pressroom is always monitored by the press operator. Paper may pick up or lose moisture from air and according to that it may enlarge or shrink [4] [106]. In winter the relative humidity is too low which will cause problems with static electricity and tight edged sheets. As a result, misregister or wrinkles occurs. Therefore, moisture should be added to the environment to avoid this kind of problem during

production [4]. Moreover, there is a chance to receive cartons of paper which may not be flat because of packing conditions. So before packing, make sure that the paper must be in a moisture-vapor barrier to avoid any gain or loss of moisture [4]. High relative humidity can cause wavy edged sheets and also slow down the drying process of ink [4]. Misregister can occur if printers do not have temperature and humidity- control systems fitted, particularly in their paper store room [107].

Cellulose fibres are cylindrical [4]. Paper is formed by crisscrossing and interlocking between these fibres. But the fibres tendency of alignment is always in the direction of the paper web travels through the paper machine [4]. Grain of paper is created from this fibre alignment. The edge of the paper web is parallel to the grain direction. When paper absorbs moisture, the fibres swell more across the width than in length. It expands in the grain direction two to eight times than original [4]. Therefore, in sheetfed presses the paper is cut in a way that the grain direction should be parallel to the long side of sheets [4]. In offset printing paper used is also long grain, where the grain is in long dimension of the sheets and also parallel to the face of the press cylinders. Dimension changes like expansion of paper due to water absorption in long grain sheet can be compensated by shifting the cylinder packing under the plate and blanket [4] [106]. However, in short grain paper dimensional changes occur from side to side which cannot be compensated in any way [4]. Due to moisture absorption long grain paper will have waviness around the cylinder direction. Therefore, there is a good chance of paper being ironed out from the gripper to the trailing edge avoiding wrinkling or misregister [106].

Therefore, to avoid such kind of print defect, the pressrooms should be air-conditioned where the relative humidity and temperature of air can be controlled. Paper should have slightly higher (about 5%) [106] relative humidity than the pressroom relative humidity. Usage of water in fountain solution should be minimized by alcohol or an alcohol substitute to keep the plate clean and thus reducing the possibilities of such kind of print defect. [106].

Mechanical stretching [116] of press sheet is another reason behind the misregister in printing [106]. In offset printing, while separating paper from blanket, it tends to stretch in the direction of its travel. For long grain paper the sheet ironed out and regains its original dimension as long as the stretching of paper is not beyond its elastic limit. If the stretch is beyond the elastic limit, it can cause permanent dimensional change. Mostly in a multicolour press stretching occurs in first printing unit due to mechanical stress as well as absorption of fountain solution. If this happens, the succeeding colors are printed wider while the first one print short. And it becomes

difficult to register all the colors properly that leads to misregistration. This situation can be handled by the operator by packing the plate and blanket [106].

In web printing to maintain register the paper must run without weaving through the press [106]. Inadequate web tensions, wound rolls, malfunctioned devices can cause weaving. Side-to- side misregister can occur due to uneven moisture content across the web, tight or baggy edges at roll ends, unwrapped rolls leaving in a high humidity area causing distortions. when web tension is lower in printing units than at the infeed, the width of the web increased after first color is printed. And as a result, the succeeding colors are printed short which leads to misregistration. To avoid this situation, bustle wheels [106] are used in the centre of the web ahead of later printing units to narrow the web, increasing the printing width.

Sheet to sheet misregister occurs generally in lightweight papers. Primarily it happens in the around-the-cylinder direction but it becomes worst in the back edges of the sheet. While ironed out the sheet, by differences in sheets because of multiple-roll sheeting or lack of flatness in sheet this kind of misregister occurs. To minimize this kind of problem printers can run with a minimum of back-cylinder pressure, using lowest tack ink, using lower –tack blanket and run the paper through the press without printing prior to actual pressrun. It is recommended that paper should run through the press prior to actual press-run can minimize the chances of sheet-to-sheet misregister [106].

6.2 Problem Statement

Generally, in printing presses misregistration print defect identification is done manually that depends on the ability of a person's identification capacity and can be varied man to man. This conventional identification procedure is very much time consuming and relies on individuals' ability. Moreover, there is no particular way to detect that how much color or which color is shifted except eye assumption. And according to the operator's ability of prediction the necessary step in particular press unit is taken to rectify this kind of print defect. There is no other standard procedure where the shifting can be measured or quantified without human intervention. Therefore, in this presented experimental work identification and quantification of colors has been done. But the challenge comes while processing the image digitally as the whole process has to deal with four colors: Cyan, Magenta, Yellow and Black. For quantification purposes, the printed samples' images are digitally acquired in the RGB color space. But the actual color shifting happens in CMYK color space in printed sample as printing

is done in CMYK color space. Therefore, RGB image has to convert again into CMYK color space in digital platform where there is a chance of degradation in color values during conversion process. Moreover, real life offset test samples contain other print problems besides misregistration which can hinder the process of identification and quantification of shifting of colors.

6.3 Sample Preparation

For this experimental work, the printed samples are collected from different offset printing presses. All sample used for this study are real-life samples. Image acquisition of these samples is done by using mobile camera with 48-megapixel resolution and in D65 illumination. The whole camera setup is made with a simple handmade approach (Figure 3.3), where camera is placed in the top and the sample is placed right under the camera. The distance between camera and samples are same for each tested sample.

6.4 Proposed Method

The steps of presented method [117] for identification and quantification of colors of printed sample image is shown in Figure 6.2. All the real-life samples used in this experimental work are collected from offset commercial presses. Digitization or acquisition of printed sample images is done by using mobile camera with 48 megapixels with D65 illumination. The whole experiment is done in MATLAB programming software. Here each step of proposed method is described briefly. Regarding this context, here in this experiment two distance metrics are considered to prove the strength of proposed method. One is Euclidean distance metrics and another one is Manhattan distance metrics. Prior to distance measurement the same steps are followed for both these two distance metrics.

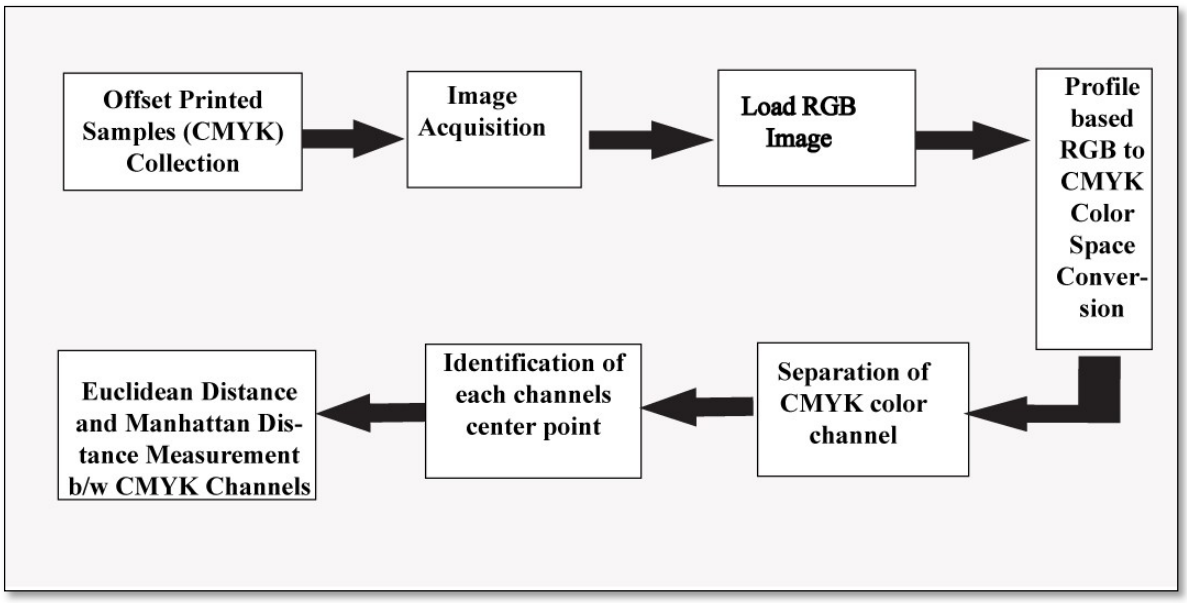

Figure 6.2 Flow Chart of Proposed Method of Misregistration Printing Defect Detection

6.4.1 Image Acquisition and Conversion

After collecting printed image samples, it is needed to be converted into digital image for performing any image processing operations. For the digitization of printed samples, a mobile camera with 46-megapixel resolution in D65 illumination is used. After image acquisition the digital images are used for further operations. Now as in misregistration the shifting of colors happens in four color that is cyan, magenta, yellow and black, acquired RGB digital image has to be converted into CMYK color space to detect the misalignment. Therefore, in next step RGB image is converted into CMYK color space. But the problem arises during the conversion process because there are chances of degradation of the color values. Therefore, a profile-based conversion has been done to convert the RGB image into CMYK color space.

RGB (Red, Green, Blue) is an additive color [110] model commonly used in devices such as screens, monitors, and digital cameras. In this model, colors are created by combining light in varying intensities of red, green, and blue. The absence of all colors, represented as (0, 0, 0), produces black, while the presence of full intensity, represented as (255, 255, 255), creates white. Whereas, CMYK (Cyan, Magenta, Yellow, Black) is a subtractive color model [110] primarily used in color printing. In this model, colors are created by subtracting light from white through the use of cyan, magenta, yellow, and black inks. The absence of all inks results

in white, (assuming white paper is used), while full ink coverage produces a rich black. Now the challenge arises because direct translation between the two models is not feasible due to differences in their color gamut's, or the range of colors they can represent. RGB can produce more vivid and diverse colors than CMYK, making conversion essential for adapting digital designs for printing while maintaining as much color accuracy as possible. To address these challenges, tools like International Color Consortium (ICC) profiles [118] are employed to standardize color reproduction across different devices and ensure consistency.

International Color Consortium (ICC) [118] has developed such standards and architectures which supports color management transforms and workflows. The architecture incorporates two basic components: one is called profile and the second component is Color Management Module (CMM) [118]. Profile contains information concerning the transform which is input or source profiles and output or destination profiles. According to the International Color Consortium (ICC), the color management system uses device profiles in the form of look-up tables, metrics etc [118]. ICC profile performs two operations during the color transform. First its co-ordinates transformations where device color code values relate to colorimetric code values in the Profile Connection Space (PCS) [118]. PCS serves as an intermediate color space during color transformations between different color profiles. When converting colors from one color space to another using ICC profile, the colors are first converted to the PCS which is a standardized color space designed to be device independent. And second operation is color rendering or re-rendering. In color rendering the colorimetry of an original color values is changed to better suited reproduction medium [118].

Color Management Module; the second component of the ICC architecture performs the actual processing of image data through profiles [118] [119]. Basically, CMM is a software component which is responsible for handling color transformations and representations across different devices. The CMM translates colors between various color spaces by applying rendering intents which defines how out of gamut colors are handled during color conversion. It performs gamut mapping to adjust the out of gamut colors that fall outside the gamut of the destination device or color space [120]. Therefore, it can be said that CMM are crucial for maintaining color consistency in workflows of digital imaging, printing and other applications related to color. Overall color management environment is based on profiles and CMM where color-image data can be interchanged among imaging systems.

Here in this presented work the color conversion from RGB to CMYK color space has been done using ICC profiles. All the tested samples used in this work are offset sheetfed, therefore, the conversion has been done according to that printer profile. Otherwise, the converted CMYK color spaced image data may lose some information related to color profile.

6.4.2 Color Channel Separation and Centre Point Identification

After image conversion, color channels are separated. Separation of CMYK color channels involves isolating the cyan, magenta, yellow and black components of an image. In image processing, this process is done to work with individual channels and their information. Generally, in misregistration misalignment of colors occurred. Whereas, registered images have superimposed colors. Therefore, it is necessary to segregate each channel to determine the shifting of colors. Mostly in four color printing, which is also known as CMYK printing, Cyan represents the blue green spectrum, Magenta represents the purplish-red spectrum, yellow represents the yellow spectrum and black represents the black [110] [120]. Each color is printed as a separate layer in a specific order. And during printing the colors are superimposed together to produce final full-color image. Therefore, shifting of any colors can only be detected if they are separated. Overall, color channel separation is a crucial step in the preprocessing workflow of detection of misregistration. Now after the color channel separation it is needed to find the centre point of each channel. In multicolour printing, as colors are superimposed, each color unit is positioned on top of its succeeding color unit. Therefore, it can be said that in registered image, the centre point of each color channel passes through all the four-color channel at same Cartesian point. If misregister in any color unit happens, then that particular color unit will have slight displacement from the other color channels centre point. Therefore, it is necessary to find the centre point of each color unit.

Now to find the centre point of each individual channel of CMYK,

I. First represent the image as a matrix where each element corresponds to a pixel intensity value. Here dimensions of each channel image are same. Therefore, size of the matrix will be same for each channel.
II. Next the 2D image matrix is converted or flattens into 1D array. Then sort each flattened array to arrange the pixel intensity values in ascending or descending order. If the number of pixel values (suppose N) in a color channel is odd, then the median index will be (N/2). If N is even the average of the two middle values is chosen.

III. Now this pixel values must be converted into co-ordinates to locate the point. The median index corresponds to a pixel at (i, j), where i is (median index |M|) and j is (median index |M|). Here M represents the number of rows in the 1D array.

IV. Repeat the steps for each color channel of the image. As a result, median point co-ordinates of each color channel are found which is also said to be the center point of individual color channel.

This process is useful for various applications in image processing to analysis the image for further operation. Here in this experimental work, Euclidean and Manhattan distance between each color channel's centre point is measured to identify the misalignment of colors. As there are four color channels, so the selected combination of color channels for the distance measurement are like CM, MY, YK, CK, CY and MK. If the tested image is properly registered, then the distance (centre to centre) between these channels will be zero or else the resulted distance shows how much color is shifted from each other. In Figure 6.3, the separated CMYK channels are shown. The centre point P is considered for C- channel and Q is considered as centre point of M-Channel.

6.4.3 Euclidean Distance Measurement

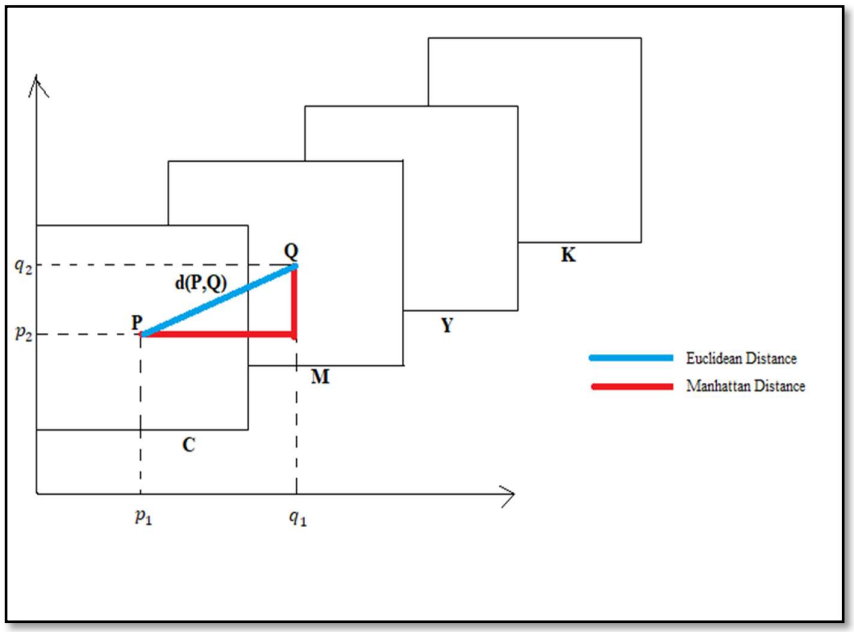

Figure 6.3 Euclidean and Manhattan Distance Between CMYK Color Channels

Now to identify and quantify the misregistration according to the proposed method, distance between centre points of each channel is measured by using Euclidean distance metrics. Euclidean distance [75] is a fundamental concept in geometry and has various applications in image processing and machine learning. It is the most common method of calculating distance between two points in a Cartesian co-ordinate system. Moreover, in the field of computer vision, operations like image analysis, measuring differences between images, object recognition, there is an extensive usage of Euclidean distance metrics.

Let's assume in Figure 6.3 C-channel has the centre point P (p1, p2) and similarly M-channel has the centre point Q (q1, q2). Then the Euclidean distance [75] between P and Q can be measured as:

$$ED(P,Q) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2} \qquad \text{eq. 6.1}$$

This formula (eq. 6.1) generalizes to higher dimensional spaces as well. It calculates the straight –line distance between two points in Euclidean space. The square root of the distance ensures that the result is a positive value providing a measure of the magnitude of the vector between the two points.

Despite all the application in various fields there are some limitations that can affect certain performance. Euclidean distance assumes that there is linear relationship between features. In case of non-linear relationships, it may not give accurate results.

After separation of each color channel Euclidean distance is measured with every possible combination of colors like CM, MY, YK, CK, CY and MK. The results are shown in Table 6.1 and 6.2 for misregistered and registered sample respectively. if the outcome of the Euclidean distance between two channel is zero then it can be said that those color channels are overlapped with each other. Otherwise, there is misregister in that particular color channel.

6.4.4 Manhattan Distance Measurement

Manhattan distance [75] is a metric which is used to measure the distance between two points in a grid-like space. The grid-based movement can only occur along orthogonal pathway like up, down, right and left. It is also called 'city block distance'. Here in this experiment despite Euclidean distance metric, Manhattan distance metric is also used to measure the color shifting in a printed sample image. In Figure 6.3, the distance between P and Q can be measured as:

$$MD(P,Q) = |q_1 - p_1| + |q_2 - p_2| \qquad \text{eq. 6.2}$$

Mathematically, the Manhattan distance (eq. 6.2) between two points in a 2D Cartesian co-ordinate system is calculated as the sum of the absolute differences of their respective co-ordinates [75]. Visually it represents the shortest path between two points travelling through horizontally and vertically. Manhattan distance between any two points is always non-negative and symmetric. This property is true regardless of the orientation of the grid or co-ordinate system and also consistent. But this distance metrics have orthogonal dependency. That means it is sensitive to changes in the orthogonal direction. This property makes Manhattan distance suitable for grid-based environments where movements are constrained to these directions. There are various applications of Manhattan distance in the field of image processing and computer vision tasks like edge detection, segmentation and object recognition. It helps to measure the similarity between pixel values or features in images. Moreover, it is applied in urban planning and geographical analysis, clustering algorithm, grid-based game development etc. The simplicity, computational efficiency and suitability in grid-like structures make the application of Manhattan distance metrics versatile with widespread utility in diverse fields. In Table 6.1 and 6.2 the distance between four color channels for misregistered and registered samples are shown respectively. Depending on the outcome the misregister in any color channel can be detected.

6.5 Result and Discussion

The presented method is tested with numerous samples collected from commercial offset presses. A mobile camera with 48 megapixel and D65 illumination in a handmade experimental set up is used to capture the digital form of these samples. The main purpose behind using mobile camera is to make the procedure of detection cost effective with an automated approach. Here some of the tested image's result and discussion are given below.

6.5.1 Result and Discussion for Misregistered Sample

Here some of the tested Misregistered sample images with their respective images in CMYK color space are shown in Figure 6.4, 6.5 ,6.6 and 6.7.

Figure 6.4 Misregistered Sample 1: (a) Original Image; (b) Converted CMYK Image; (c) Image in C-channel; (d) Image in M- channel; (e) Image in Y-channel; (f) Image in K-channel.

Figure 6.5 Misregistered Sample 2: (a) Original Image; (b) Converted CMYK Image; (c) Image in C-channel; (d) Image in M- channel; (e) Image in Y-channel; (f) Image in K-channel.

Figure 6.6 Misregistered Sample 3: (a) Original Image; (b) Converted CMYK Image; (c) Image in C-channel; (d) Image in M- channel; (e) Image in Y-channel; (f) Image in K-channel.

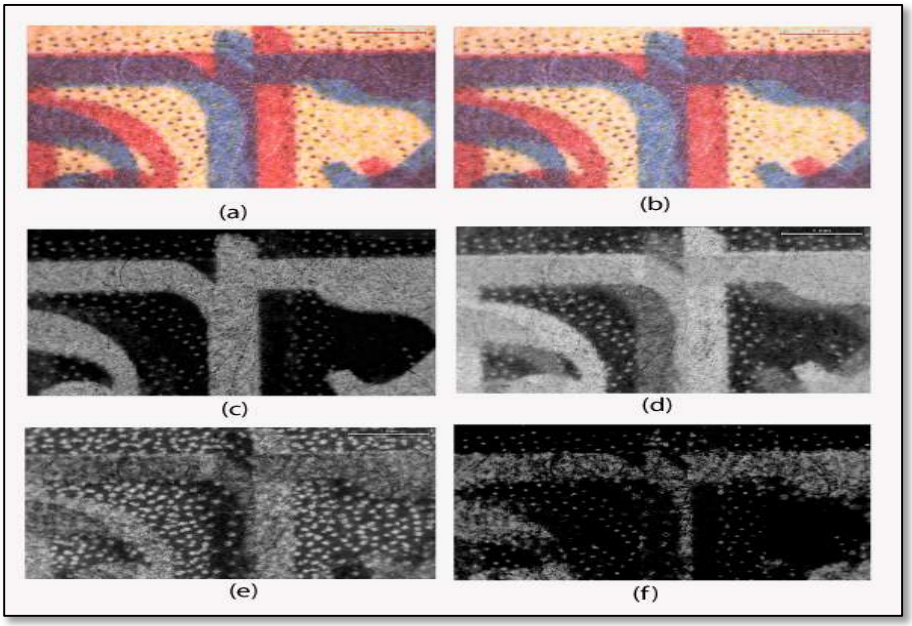

Figure 6.7 Misregistered Sample 4: (a) Original Image; (b) Converted CMYK Image; (c) Image in C-channel; (d) Image in M- channel; (e) Image in Y-channel; (f) Image in K-channel.

In Figure 6.4 (a) sample of Misregistered printed image in RGB color space after image acquisition is shown. Next the RGB image is converted into CMYK color space which is shown in Figure 6.4 (b). Now as per the proposed method the converted CMYK image is separated and sliced. Figure 6.4 (c), Figure 6.4 (d), Figure 6.4 (e) and Figure 6.4 (f) depicts the image in C, M, Y and K channel respectively. Now considering each color channel separately (the images are of same size) the centre point is identified. After that, Euclidean and Manhattan distance between the channels from one channel's centre point to another channel's centre point is calculated. Though visually this printed image sample does not depict any color sifting but if it is tested with conventional way through linen glass then it shows the misalignment of colors. This fact is also supported by the results calculated by Euclidean and Manhattan distance metrics shown in Table 6.1. The results of Euclidean distance show the presence of misregistration as well as quantifying the amount of misregistration. The same fact is also proven by Manhattan distance metrics and it supports the result of Euclidean distance metrics measurement. For this sample, it is shown that cyan channel is superimposed with magenta channel. Whereas other color channel has distance and can be said that they are misregistered. Therefore, the potential of using these two-distance metrics proves its strength in quantification of color that is how much and which color channel is shifted during the process. Whereas in conventional way operator can only identified the color shifting which takes time to identify and depends on their ability of detection. Euclidean and Manhattan distance is measured in pixels (Table 6.1). However, the distances measured using Euclidean and Manhattan metrics can be converted into inches by dividing them by the resolution of the sample images (measured in pixels per inch, PPI or DPI).

In Figure 6.5, another misregistered printed image sample is shown with their individual image in CMYK color channels. Misregister in this sample is also not possible to identify with naked eye and can be considered as a good copy mistakenly. But if it is tested through linen glass misregistration of colors will be shown. The computed result in Table 6.1 depicts the quantification of color shifting. The result shows that none of the color is aligned with each other. Both the distance metrics measure distance between color channels. This much misregister is unacceptable in any way. Therefore, the necessary action should be taken according to the result of color shifting.

In Figure 6.6, misregistration of sample image affects the visual appearance and readability of the printed matter. Moreover, presence of noise is also responsible for affecting the print quality of the image. The result of Euclidean and Manhattan distance metrics (Table 6.1) shows that

all channels in this sample image are misregistered with each other. As the amount of misregistration is more in this printed image, the misalignment of colors can be identified visually to some extent. But how much or particularly in which color channel misregistration happens can only be identified through this proposed method.

In Figure 6.7 another sample image of misregistration is shown. The Euclidean distance metrics (Table 6.1) shows that only magenta and yellow channel are superimposed. The other distance metrics that is Manhattan distance also depicts the same fact. The amount of color shifting in this image affects the visual appearance and readability with naked eye.

Table 6.1: *Euclidean and Manhattan distance between color channels in CMYK color space for misregistered sample*

Sample	**CM**	**CY**	**CK**	**MY**	**MK**	**YK**	**CM**	**CY**	**CK**	**MY**	**MK**	**YK**
	Euclidean Distance						**Manhattan Distance**					
Sample 1 (Figure 6.4)	0	12.16	73.76	12.16	73.76	61.68	0	14	91	14	91	77
Sample 2 (Figure 6.5)	10.198	3.60	146.932	7	155.438	148.929	12	5	193	7	201	194
Sample 3 (Figure 6.6)	135.203	121.276	107.365	38	35.117	17.923	184	146	136	38	47	23
Sample 4 (Figure 6.7)	22.825	22.825	94.366	0	116.481	116.481	31	31	113	0	144	144

6.5.2 Result and Discussion for Registered Sample

Here some of the tested registered sample images with their respective images in CMYK color space are shown in Figure 6.8, 6.9 and 6.10.

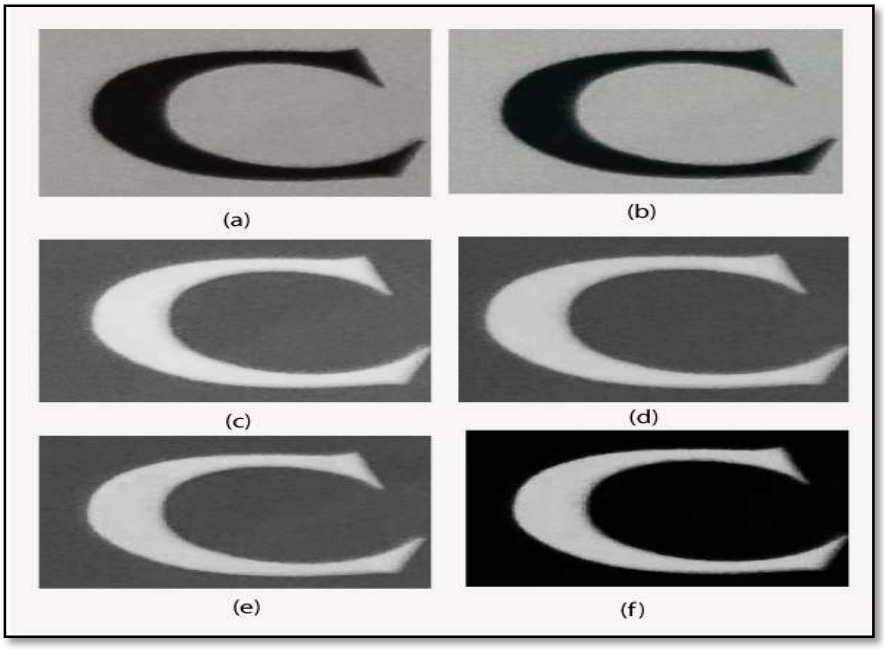

Figure 6.8 Registered Sample 5: (a) Original Image; (b) Converted CMYK Image; (c) Image in C-channel; (d) Image in M- channel; (e) Image in Y-channel; (f) Image in K-channel.

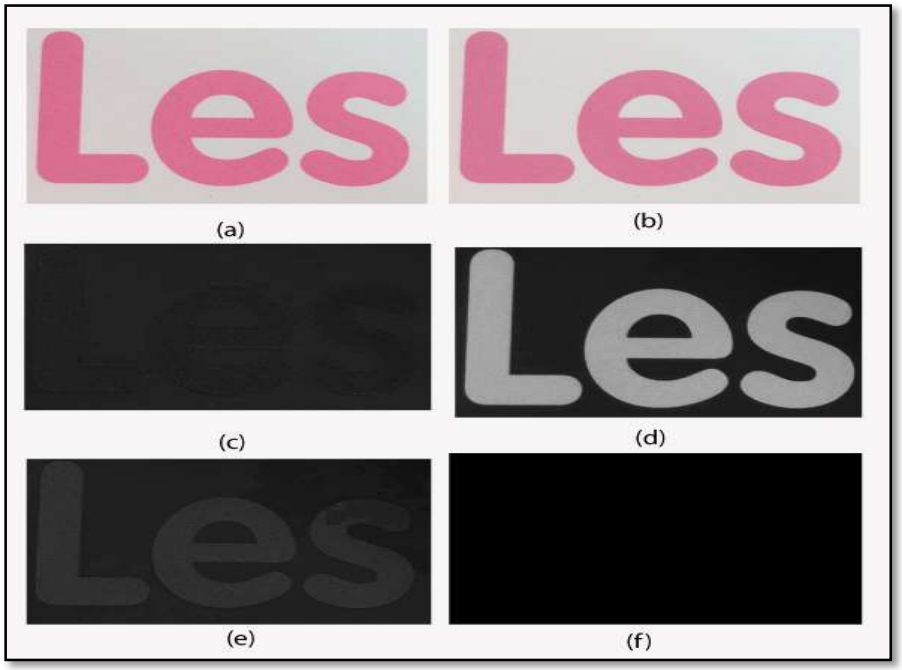

Figure 6.9 Registered Sample 6: (a) Original Image; (b) Converted CMYK Image; (c) Image in C-channel; (d) Image in M- channel; (e) Image in Y-channel; (f) Image in K-channel.

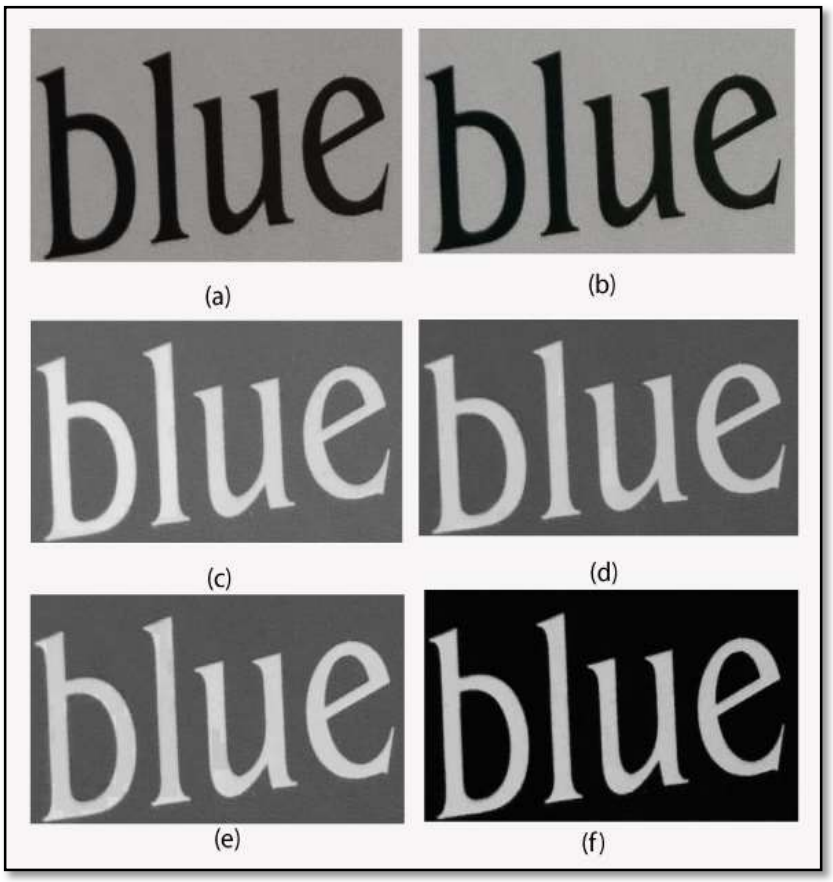

Figure 6.10 Registered Sample 7: (a) Original Image; (b) Converted CMYK Image; (c) Image in C-channel; (d) Image in M- channel; (e) Image in Y-channel; (f) Image in K-channel.

The printed sample images shown in Figure 6.8, 6.9 and 6.10 are registered printed products and considered as good copies. Table 6.2 shows the result of Euclidean distance and Manhattan distance measurement of these samples. Visual appearance of these samples is better however the results show some misregistration is present in black channel only. This much misregistration is negligible and does not affect the quality of the product output. The distance between other channels is zero which depicts other color channels are superimposed and registered properly. Therefore, it can be said that though there is some misregistration in black channel but as the other channels are properly aligned with each other, visually the image appears as good and considered for quality output.

Table 6.2: *Euclidean and Manhattan distance between color channels in CMYK color space for registered sample*

Sample	CM	CY	CK	MY	MK	YK	CM	CY	CK	MY	MK	YK
	Euclidean Distance						Manhattan Distance					
Sample 5 (Figure 6.8)	0	0	0.5	0	0.5	0.5	0	0	0.5	0	0.5	0.5
Sample 6 (Figure 6.9)	0	0	133.66	0	133.66	133.66	0	0	154	0	154	154
Sample 7 (Figure 6.10)	0	0	0.7	0	0.7	0.7	0	0	1	0	1	1

6.6 Conclusion

Quality of image in terms of visual appearance and readability is affected for many reasons. Misregistration printing defect is one of the print defects which causes blurred image or fuzzy image and as a result it affects the visual quality of the printed material. Generally, in offset print production, when colors are not aligned properly this printing defect occurs. Generally linen glass is used in industry to check the alignment of colors. That means identification of misregistration depends on manual efficiency or the ability of identification of the press operator. There is no such way to quantify the shifting of colors or measure how much color is shifted or which color is shifted. Here in this presented work Euclidean and Manhattan distance metrics is used to quantify the amount of misregistration in CMYK color space. This proposed method can be a potential alternative for less manual approach with a proper quantification of this print problem. Conventional way of detection of misregistration is time consuming whereas this presented approach based on image processing and computer vision method becomes popular research direction.

CHAPTER - 7

Prediction of Print Contrast and Density using Machine Learning

CHAPTER-7

Prediction of Print Contrast and Density using Machine Learning

7.1 Introduction

Image quality depends on the ability to distinguish difference between density values within the image [106]. When there are greater differences in density between midtones, highlights, and shadow tones, it becomes easier to see the details. The difference in tonal density is referred to as print contrast [106]. Print contrast is determined by the whiteness and brightness of the paper, as well as the density of the ink [106]. Insufficient paper opacity and the resulting excessive show-through can diminish print contrast that can affect overall print quality. In printing, print contrast is a measure used to evaluate the quality of printed output by comparing the density of the shadow areas to the solid ink density [121]. Solid ink density represents the optical density of a fully saturated (100%) ink area. Higher solid ink density generally contributes to richer, more vivid colors. Print contrast is typically used to assess the printer's ability to reproduce details in dark regions of an image. It refers to the ability of a printing system to maintain detail in the shadow areas of an image, while also achieving high solid ink density. It is an essential parameter for evaluating print quality, especially in processes like offset, flexographic, and digital printing. Print contrast compares the density of a printed solid color with the density of a 75% tint of the same color [113]. Print contrast values are specifically useful for determining the optimal ink density under current printing conditions

and assessing the quality of a printed reproduction, as high print contrast typically indicates superior print quality [121]. This print attribute is very much important to measure because it indicates the image's tone reproduction at a critical point on the tone curve [113]. Moreover, this attribute helps the buyers to check the quality of print. Although print contrast itself carries a huge information about ink laydown density, dot gain in three -quarter tone area [116] that refers to the increase in size or darkness of printed halftone dots in the 75% tonal region compared to their intended size. It affects the accuracy of tonal reproduction in darker areas of an image. It can also be predicted from contrast that if the value of contrast is high it can be said that solid ink density is increased without increase the density of three-quartertone patch. As a result, visual contrast is increased between solid and three-quartertone areas [113].

Achieving very high print contrast requires optimal printing conditions, including well-maintained equipment that transfers images with faintest impression squeeze, highly stiff or tacky inks, and premium-quality coated paper [121]. Some commercial processes operate at higher contrasts than those recommended for accurate color reproduction. These higher contrasts enhance the brightness differences between light and dark areas of the image and also boost colourfulness. Here are some possible reasons for the preferred practice of increasing contrast:

When the original scenario is lit by natural daylight, the reproduction often has much lower luminance. This reduced luminance diminishes perceived contrast, brightness, and colorfulness. Raising the contrast helps create images that more closely resemble the original scenario. Increasing contrast in the imaging system can mimic the partial discounting of haze effects on distant objects [122].

Optimal print contrast is achieved by using bright blue-white paper with sufficient opacity, combined with the highest possible ink film density. White paper provides better contrast with both black and colored inks compared to colored paper [106]. Print contrast is critical in quality control for offset, digital, and flexographic printing processes to ensure sharp, high-quality reproductions of images and text. It is measured using a densitometer or spectrophotometer during print runs.

The relative printing contrast can be calculated using the measured values of solid ink density and the ink density in the halftone screen . For greater accuracy, it is recommended to measure the value of halftone ink density in the three-quarter tone region [113] Therefore, print contrast can be expressed [121] as:

$$\text{Print contrast} = 100 \times \left(\frac{Solid\ ink\ density - Densit\ \ of\ 75\%\ tint}{Solid\ ink\ density}\right) \qquad \text{eq. 7.1}$$

The reproduction of printed images, particularly during the printing phase, is highly sensitive to tonal and color variations. These variations are primarily influenced by factors such as dot size, ink trapping, and ink film thickness. Ink film thickness is typically measured using a densitometer and is expressed in units of density, abbreviated as optical density [121].

The theory of light in relation to optical density is grounded in how light interacts with a material, specifically the absorption, reflection, and transmission of light. Optical density (OD) quantifies the extent to which a material, such as an ink film in printing, reduces the amount of light that passes through or reflects off it. When light strikes a material (such as ink on paper), part of it is absorbed by the material, and the rest is either transmitted through the material or reflected back. Optical density primarily relates to the absorption of light by the ink or coating. In printing, ink layers on paper primarily absorb light, but they can also reflect some. The total optical density can be a result of both absorption and reflection, depending on the specific printing process and ink used. For example: Absorption reduces the light passing through the ink, contributing to a higher optical density. Reflection occurs when light bounces off the surface, affecting the perceived brightness and color.

The density equation (eq.7.2) represents the logarithm of the ratio of the light absorbed of an absolute white reference to the light absorbed by a measured ink [121]. Densities are expressed as logarithmic values as because it is related to the human eye's visual response to radiant energy [121]. The human eye perceives increasing amounts of light in an approximately logarithmic manner—each time the radiant energy is doubled, the observer's perceptual system experiences only a tenth of that increase. The density equation can be expressed as:

$$\text{Density (D)} = \log_{10}\frac{1}{R} \qquad \text{eq. 7.2}$$

Where,

D is the **Print Density**.

R is the **Reflectance** (the ratio of light reflected from the surface).

7.2 State of Art

In (2017) flower classification has been done by Xia et al. [78] with inception v3 model with accuracy 94-95%. In (2018) Ghosh et al. [41] classified printed circuit boards defects are using inception v3 model with accuracy 91.125%. Mittal et al [79] (2018) presents the Inception-V3 model to break image-based CAPTCHAs, achieving over 91% accuracy on a real Facebook CAPTCHA dataset. Li et al (2019) [81] presented a transfer learning approach using the Inception-V3 model for classifying fresh tea leaves. Experiments on a self-created dataset achieved over 95% accuracy, improving to 98% with data augmentation. The method enhances efficiency and reliability for industrial tea production. Bae et al. (2020) [83] proposed a heart sound classification model using mel-spectrogram images and a fine-tuned Inception-V3 network. Feature extraction and selection techniques enhanced classification accuracy, achieving 97.5% with top features. In (2021), inception v3 based face shape classification had been done by Nabil et al. [86] with accuracy 94.3%. Mamatha Balipa and Ashton Castalino [89] in (2022) detected Alstonia Tree using CNN and Inception V3 Algorithms with accuracy more than 80%. In (2023) covid-19 classification using Inception v3 by Iqbal et al. [92] shows the accuracy above 90%. In (2023) Kalpana et al. [94] did a study based on classification and prediction of disease on cotton leaves using inception v3 with accuracy 96%. Susanti *et al.* [96] in (2023) did a classification of beef images using the Inception V3, Transfer Learning model with accuracy 87.14%. Joshi et al (2023) [125] did a study which explores tea leaf disease identification using YOLOv5 and Inception-V3 models on a dataset of 6,556 healthy and diseased leaf images. The Inception-V3 model achieved a 94.3% F1-score for Red Rust classification, while YOLOv5 reached a mAP of 0.78. Verma et al [97] (2023) compares Inception-V3 and VGG-19 models for traffic sign detection using a dataset of 44,028 images. The Inception-V3 model achieved a higher accuracy of 94.38% compared to 90.12% for VGG-19. Kankariya *et al.* (2024) [98] did a study to explore advancements in Sign Language Recognition (SLR) using deep learning, specifically CNN and Inception-V3 models. The proposed method achieved 98.94% accuracy. Kaur *et al.* (2024) [99] presented a fine-tuned Inception-V3 model for the multiclass classification of groundnut diseases, including Early Leaf Spot, Late Leaf Spot, Rust, and Groundnut Rosette Disease. The model, trained on a comprehensive dataset of groundnut leaf images, achieves high accuracy of 96%. Rani *et al.* (2024) [100] proposed an Inception-V3-based CNN model for malware detection, achieving 90.6% accuracy and 92.6% specificity. Comparing the state of art of previous work with

Inception v3 model, for color classification from unknown patch, inception v3 gives the 98% accuracy in this presented method.

7.3 Proposed Method

This study aims to predict printed color contrast and density using a deep learning approach. For this purpose, 700 color patches graduated 0 to 100 dot percentage of Cyan (C), Magenta (M), Yellow (Y), and Black (K) are printed in six types of different paper. Initially, the physical density and contrast of the color patches are measured using a SpectroEye Spectrophotometer (make: X-rite) [126] showing in Figure 7.1. Subsequently, the color patches image acquisition has been done using a mobile camera with 48 megapixels. The captured images are then augmented using rescale, shear, zoom and horizontal flip and divided into training and testing datasets. A conventional classifier i.e. Inception v3 is applied to the training dataset to train the model, which is subsequently tested on the testing dataset to classify the color into four groups i.e. C, M, Y and K. After that, features are extracted from these digitized sample using Matlab environment for regression analysis. Before regression analysis, again data augmentation has been done using time wrap, quantize, drift and reverse method. After that Exploratory Data Analysis (EDA) [123] [124] has been done to visualize and detect the features suitable for further operations and identify anomalies or outliers. Here for the contrast and density prediction three types regression model is used i.e Logistic Regression (LR), Support Vector Regression (SVR) and Random Forest (RF). Also, a comparative study has been done to find out the suitable regression model for the prediction of density and contrast. The basic workflow of the proposed model is illustrated in Figure 7.2.

Figure 7.1 X-rite Spectrophotometer

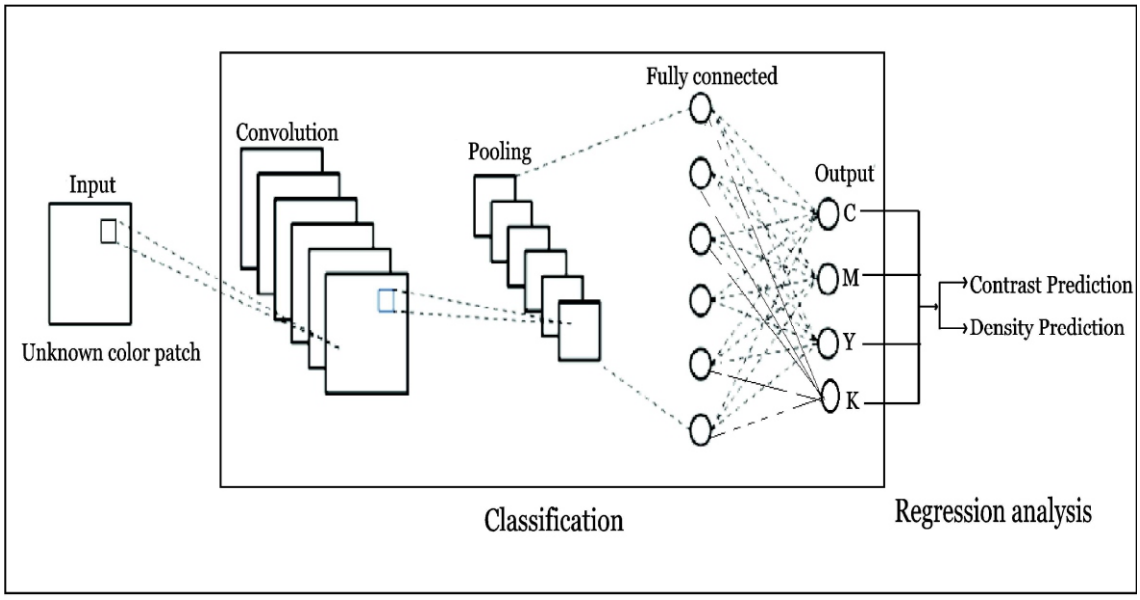

Figure 7.2 Proposed Model for Contrast and Density Prediction

7.3.1 Sample Preparation

Color patches are printed on six different types of paper using the offset printing process. For this experiment, cyan, magenta, yellow, and black color patches are selected, with a total of 700 target data points considered. The physical contrast and density of these patches are measured using a spectrophotometer (Figure 7.1). Subsequently, the color patches are digitized using a 48-megapixel mobile camera under D65 illumination, employing a specific camera setup (Figure 3.3). Next this digitized samples are divided into training and testing dataset for classification.

7.3.2 Augmentation

Augmentation [127] [128] refers to the process of artificially increasing the size and diversity of a dataset by applying various transformations or modifications to the original data. This technique is commonly used in machine learning, particularly in training deep learning models, to improve the model's generalization capabilities and make it more robust to variations in the data.

The purpose of data augmentation is to enhance machine learning models by addressing various challenges in training. It increases the dataset size by generating new, synthetic data from the existing data, which is particularly useful when the original dataset lacks diversity and reduces overfitting [130]. Augmentation also improves model generalization by exposing the model to varied versions of the data, helping it generalizes better to unseen examples and reducing overfitting. It simulates real-world variations, such as changes in lighting, perspective, noise, or timing, that the model might encounter during deployment. Models trained on augmented data become more robust to imperfections like sensor noise, distortions, or environmental changes. Additionally, augmentation increases diversity by creating a broader range of training examples, which improves the model's ability to handle a wide variety of inputs.

7.3.2.1 Augmentation for Classification

Before classification captured images are augmented using Rescale, Shear, zoom and flipping techniques. These techniques are called geometrical transformations [127].

Rescaling adjusts the range of pixel values in an image, typically normalizing the pixel intensity values to a range of 0 to 1 or -1 to 1. This process is essential for improving the stability and

convergence of deep learning models, as many neural networks are sensitive to the range of input values. Rescaling prepares the image data for better compatibility with the model's architecture. For instance, if an image has pixel values in the range [0, 255], dividing these values by 255 effectively rescales them to the range [0, 1].

Shearing applies a geometric transformation that shifts the image's pixels in one direction, effectively creating a "slanted" or skewed appearance. This technique is used to simulate changes in perspective or camera angles, enhancing the diversity of the dataset and making models more robust to such variations. The degree of shearing is controlled by the shear intensity, which can be specified in degrees or as a fraction.

Zooming involves cropping the image inward to simulate a zoom-in effect or adding a border to create a zoom-out effect. This technique is used to mimic varying camera distances or resolutions, allowing models to learn from different scales of the image. Additionally, zooming can help the model focus on specific regions within the image, enhancing its ability to recognize details at different levels of magnification.

A horizontal flip reverses the image along its vertical axis, effectively mirroring it horizontally. This technique is used to simulate real-world variations, such as different viewpoints, and is particularly useful for images where the horizontal orientation does not impact the class.

7.3.2.2 Augmentation for Regression

Before regression analysis data from different feature of images has been augmented by time wrap, quantize, drift and reverse method [129] [130]

Time warping for data involves altering the timing of data points while keeping their order intact. This can be done by compressing or stretching parts of the sequence to simulate timing variations, introducing variability that helps models generalize and handle irregularities in data collection intervals. Techniques include interpolation to stretch or compress sequences by adding or removing points, random sampling while maintaining sequence order, and elastic distortion.

Quantization in data augmentation involves reducing the precision of data by rounding values or simulating discrete levels, introducing small perturbations to mimic real-world imprecision. This technique is particularly useful for making models robust to noise, such as those caused by low-resolution sensors or limited precision in data collection devices. By emulating these

imperfections, quantization helps models to generalize better to noisy or degraded inputs, enhancing their performance in practical scenarios.

Drift in data augmentation involves intentionally introducing small, gradual changes or offsets to data values over time or across observations to simulate real-world scenarios where data collection processes or environmental factors may cause slight deviations. The purpose of this technique is to improve model robustness against naturally occurring drifts in datasets and help models generalize better by learning to adapt to small variations caused by sensor noise, environmental changes, or system inconsistencies.

Drift can be implemented by adding random noise, such as Gaussian or uniform noise, to introduce small random offsets to data points. Gradual shifts, either linear or non-linear, can also be applied over time to mimic progressive drifts.

Reverse in data augmentation involves flipping the order of data points in a sequence, effectively reversing their temporal or spatial arrangement. This technique is often applied to sequential data, such as time-series or text, and is particularly beneficial in tasks where the sequence's overall content or features remain meaningful even when its order is inverted. The purpose of reversing is to increase dataset diversity and introduce symmetry-based variations, enabling models to generalize better by learning sequence-invariant features when the task allows. It is also useful for testing a model's ability to adapt to reversed patterns or sequences.

7.3.3 Classification with Inception V3 Classifier

A conventional classifier refers to a traditional machine learning model or algorithm that is used to categorize data into predefined classes or categories. These classifiers are typically trained using labeled data, where the model learns patterns or features in the data that correspond to the target labels. Depending on the specific problem, characteristics of data, interpretability requirements, classifier is chosen. For this present study, the Inception v3 model is utilized as a classifier to categorize unknown color patches into four classes: Cyan (C), Magenta (M), Yellow (Y), and Black (K).

7.3.3.1 Inception v3 model

Inception v3 [131] [132] is a widely used convolutional neural network (CNN) model in machine learning, primarily designed for image classification and feature extraction tasks. It is a part of the Inception family of models developed by Google, building on the success of earlier versions. Inception v3 is particularly known for its efficiency and accuracy, achieved through its sophisticated architecture.

7.3.3.1.1 Key Features of Inception v3 in Machine Learning:

Inception v3 is a deep neural network with multiple convolutional, pooling, and fully connected layers for complex feature learning. It employs factorized convolutions to reduce computational cost, auxiliary classifiers to aid training, and efficient grid size reduction for better down-sampling. Its scalable architecture makes it suitable for diverse image recognition tasks.

7.3.3.1.2 Objectives of Inception v3:

I. **Improve Classification Accuracy**: Enhance the ability to classify images into predefined categories with minimal error rates, as demonstrated in benchmarks like the ImageNet Large Scale Visual Recognition Challenge (ILSVRC).

II. **Reduce Computational Complexity**: Optimize the architecture to reduce the computational load through techniques like factorized convolutions and efficient grid size reduction, making the model suitable for various hardware platforms.

III. **Enable Deeper Network Training**: Address challenges such as the vanishing gradient problem in deep networks by incorporating auxiliary classifiers and advanced design principles, allowing the model to learn from complex datasets effectively.

IV. **Scalability and Versatility**: Ensure the model is adaptable to different image-related tasks, such as transfer learning, object detection, and domain-specific applications like medical imaging and printed color analysis.

V. **Balance Performance and Resource Usage**: Provide state-of-the-art performance while maintaining computational feasibility, enabling deployment in both research and practical applications.

7.3.3.1.3 Architecture of Inception V3 model

Inception v3 is a deep convolutional neural network (CNN) designed by Google for image classification tasks. It builds on earlier versions (Inception v1 and v2), incorporating several key improvements to enhance performance while optimizing computational efficiency. Below is an overview of its architecture (Figure 7.3) [83]:

I. Initial Convolutional Layers:

The network begins with a series of convolutional layers followed by max-pooling, enabling it to capture fundamental features like edges and textures in the image.

II. Inception Modules:

The core of the Inception architecture is the Inception module, which allows the model to capture various types of features at the same depth. Each module consists of parallel convolutional layers with different kernel sizes (such as 1x1, 3x3, and 5x5), as well as a pooling layers [81]. The outputs of these operations are concatenated along the depth axis, enabling the model to learn diverse features at multiple scales simultaneously.

III. Factorized Convolutions:

Inception v3 improves on earlier versions by factorizing larger convolutions into smaller ones. For instance, instead of using a 5x5 convolution, it employs two consecutive 3x3 convolutions [81], thus reducing the number of parameters and the overall computational load.

IV. Auxiliary Classifiers:

To facilitate more efficient training, Inception v3 introduces auxiliary classifiers at intermediate layers (towards the end of each Inception module). These additional classifiers help by providing supplementary gradients during backpropagation, which accelerates convergence, particularly during the early stages of training.

V. Global Average Pooling:

Instead of using traditional fully connected layers, which can introduce a large number of parameters and overfitting, Inception v3 opts for global average pooling. This method averages the outputs of the final convolutional layer, producing a single feature map per class and significantly reducing the number of parameters.

VI. Architectural Optimizations:

- Grid Size Reduction: The architecture incorporates grid size reduction to down-sample feature maps at specific layers, which helps minimize computation.
- Factorized 7x7 Convolutions: Inception v3 replaces large 7x7 convolutions with two smaller 3x3 convolutions, further improving computational efficiency.
- Rational Activation Functions: Inception v3 uses Rational ReLU (RReLU) [81], a variant of the ReLU activation function that introduces slight nonlinearity, leading to enhanced model performance.

VII. Final Fully Connected Layer:

The model ends with a softmax layer [83], which generates class probabilities for the given image based on the extracted features.

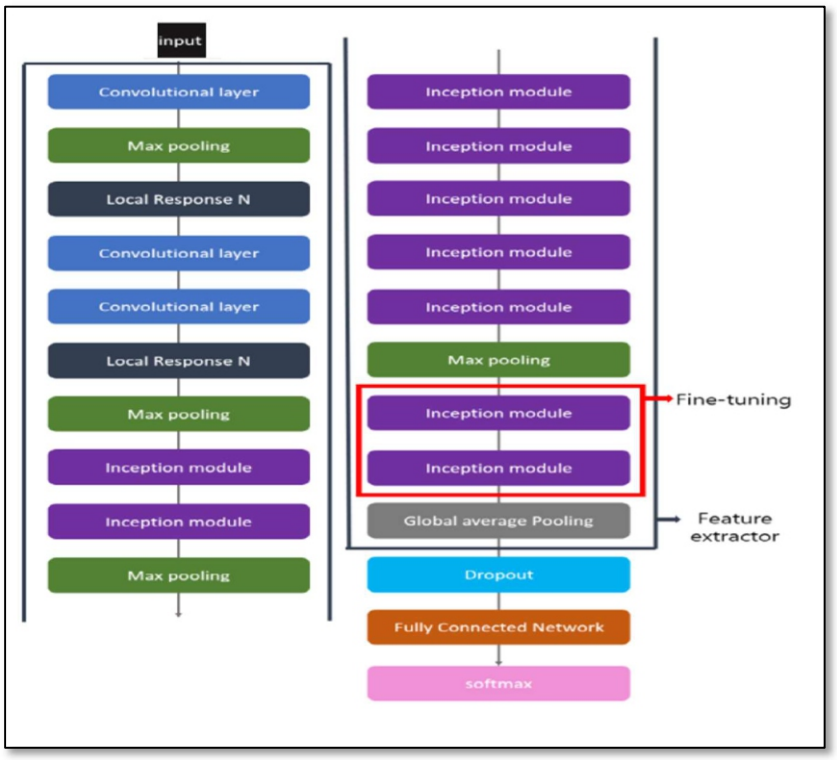

Figure 7.3 Architecture of Inception V3

7.3.4 Feature Extraction

After classification classified color patches of cyan, magenta, yellow and black are converted into HSV color space. Subsequently, features of each sample in the HSV color space are extracted using MATLAB programming. The extracted features include mean, median, minimum, maximum, entropy, variance, correlation, homogeneity, and Local Binary Pattern (LBP) for each channel of the image which is subsequently used for EDA analysis and regression analysis.

7.3.5 Feature Selection

7.3.5.1 Feature Selection by EDA Analysis

The purpose of Exploratory Data Analysis (EDA) [123] [124] is to understand and summarize the key characteristics of a dataset before applying more complex statistical or machine learning methods. It serves as a foundational step in any data analysis project and ensures that the data is suitable for the intended analysis. For this study EDA analysis has been done to

understand the correlation of features to the target contrast and density values. This helps to identify and detect the underlying patterns, trends, or relationships between variables. Moreover, to check the dataset quality, missing values are identified. Also, outliers or unusual data points which can affect the analysis is identified in this step through EDA analysis. This analysis of data helps to detect the suitability of data i.e. features of colors in HSV color space for statistical modelling or machine learning to detect the color contrast and density. It is very much important before applying any regression analysis, to determine which features are relevant for further analysis. After the observation, an understanding of the data is built up which helps in hypothesis generation and decision making to choose the suitable feature for the contrast and density prediction purpose. Here, to visualize and convey the data characteristics scatterplots between features and target contrast or density (Figure 7.10) are presented which shows the distributions or correlations of data or trends.

7.3.5.1.1 Correlation Analysis

Correlation analysis in EDA [133] (Figure 7.10) is a statistical method used to measure the strength and direction of a linear relationship between two variables. It helps to understand how one variable change concerning other. Correlation can be positive or negative. Positive correlation indicates if one variable increases other also increased, whereas negative correlation indicates if one variable increases other decreases. If there is no linear relationship between the variables it indicates zero correlation [123] [124].

Pearson Correlation Coefficient (r):

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})}\sqrt{\sum(y_i - \bar{y})}} \qquad \text{eq. 7.3}$$

Where, $x_i \; and \; y_i$ are individual data points, $\bar{x}$ and $\bar{y}$ are the means of $x \; and \; y$ respectively.

If r = 1 then it can be said it is perfect positive correlation, if r= -1 then it is perfect negative correlation, r= 0 depicts no correlation, 0 < r < 1 interprets positive correlation, -1 < r < 0 interprets negative correlation.

7.3.5.1.2 Data Visualization Technique

Exploratory Data Analysis also involves graphical method to interpret data patterns visually. In this presented method Scatter plot and heat map are used to see the correlation between the variables.

A scatter plot (Figure 7.10) is a graphical representation used to visualize the relationship between two numerical variables. It helps identify trends, patterns, and possible correlations. Each point on the scatter plot represents an observation with two values: one on the x-axis and the other on the y-axis. The pattern of points indicates the type and strength of correlation.

A heatmap (Figure 7.11) is a graphical representation of data using color gradients to show relationships between multiple numerical variables. It is commonly used to visualize a correlation matrix. Dark colors interpret strong correlation whereas light color shows weak or no correlation [134].

7.3.6 Prediction model for Regression Analysis

7.3.6.1 Logistic Regression

Logistic Regression [76] [77] [135] is a statistical method used for binary or multi-class classification tasks. Despite its name, it is a classification algorithm and also used as regression technique. It models the probability that a given input point belongs to a certain class. The purpose of logistic regression is to model and predict the probability of a binary outcome or categorical outcome, based on one or more predictor variables (features). It is used primarily for classification tasks, where the goal is to categorize data into distinct classes or categories. Logistic regression model can work in two forms:

- **Binary Logistic Regression**: The simplest form, used for binary classification tasks where the output is either 0 or 1 (for example, "spam" vs. "not spam").
- **Multinomial Logistic Regression**: Used for multi-class classification problems where the output can be one of more than two classes.

Here are the key concepts related to logistic regression:

i. Logistic Function (Sigmoid Function)

The core of logistic regression is the logistic function (also called the sigmoid function), which maps any real-valued number into a probability between 0 and 1. It is also referred as sigmoid function which employs to model the relationship between the probability of event occurring and the independent variables. The sigmoid function [135] is:

$$\sigma(z) = \frac{1}{1+e^{-z}} \qquad \text{eq. 7.4}$$

$$z = w_1x_1 + w_2x_2 + \cdots + w_nx_n + b \qquad \text{eq.7.5}$$

Where, $x_i = input\ features$, $w_i = corresponding\ weights$, z = linear combination of the input features.

ii. Binary Classification

Logistic regression is commonly used for binary classification problems, where the output is either 0 or 1 (e.g., "yes" or "no," "spam" or "not spam"). The model predicts the probability of the target variable belonging to class 1. A threshold (usually 0.5) is applied to convert the probability into a discrete class label, Thus, the final logistic regression model [136] [137] is:

$$P(y = 1 \mid x) = \frac{1}{1+e^{-(w_1x_1+w_2x_2+\cdots+w_nx_n+b)}} \qquad \text{eq. 7.6}$$

If $P(y = 1 \mid x) \geq 0.5$, classify as 1.

If $P(y = 1 \mid x)) < 0.5$, classify as 0.

iii. Odds Ratio

Odds are the ratio of the probability of an event occurring to the probability of it not occurring. In logistic regression, the output is modelled as the log-odds (logarithm of the odds). If the odd ratio is greater than 1 it means an increase in the odds of the event, whereas odd ratio less than 1 indicates the decrease in the odds of the event.

iv. Maximum Likelihood Estimation (MLE)

Logistic regression estimates the model parameters by maximizing the likelihood of the observed data. MLE finds the parameters that make the observed data most probable given the

model. The goal of logistic regression is to determine the best-fitting model that maximizes the likelihood of predicting the observed outcomes.

v. Gradient Descent

To optimize the model and minimize the cost function, gradient descent is commonly used. The algorithm iteratively updates the model parameters in the direction that reduces the cost function. It uses the gradient (partial derivatives) of the cost function with respect to each parameter.

vi. Feature Scaling

Logistic regression can be sensitive to the scale of the features. Features with different ranges can cause issues during optimization. Feature scaling (standardization or normalization) can help improve the model's performance by ensuring that each feature contributes equally to the model.

vii. Interpretability

One of the key strengths of logistic regression is its interpretability. The coefficients can be directly interpreted to understand the influence of individual features on the prediction.

7.3.6.1.1 Workflow of Logistic Regression

Logistic Regression is typically used for classification, but when adapted for regression analysis the basic workflow of logistic regression (Figure 7.4) [138] model is as follows:

- Data Preparation: Collect and preprocess continuous independent variables and dependent variables representing probabilities or log-odds.
- Model Specification: Define the logistic regression model using a sigmoid (logistic) function to model the relationship between independent variables and the dependent variable.
- Parameter Estimation: Use Maximum Likelihood Estimation (MLE) to determine the best-fitting coefficients.
- Model Fitting: Train the model using optimization techniques like Gradient Descent or Newton-Raphson.

- Prediction: Apply the logistic function to new input data to estimate probabilities.
- Evaluation: Assess model performance using metrics like log-likelihood, AIC, or R^2, RMSE, MAE if transformed back to the original scale.
- Interpretation & Deployment: Analyse coefficients for insights, validate assumptions, and apply the model for predictive analysis.

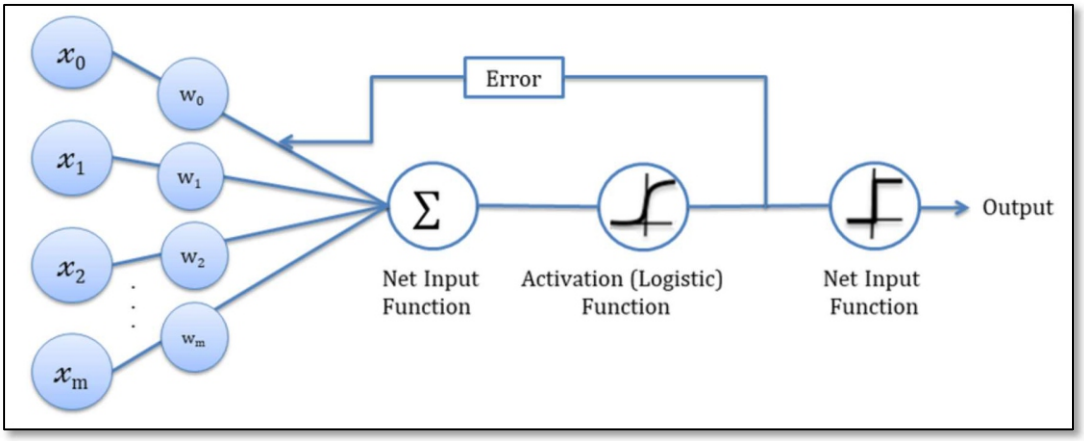

Figure 7.4 Workflow of Logistic Regression model

7.3.6.2 Random Forest

A Random Forest [139] [140] [141] is a popular machine learning algorithm used for both classification and regression tasks. It is an ensemble method that builds multiple decision trees and combines their outputs to make predictions, improving accuracy and reducing the risk of overfitting compared to a single decision tree. Key features of Random Forest are as follows:

I. **Decision Trees:** Random Forest is based on decision trees, which are tree-like structures that split data recursively based on feature attributes to perform classification or regression tasks.

II. **Ensemble Learning:** Combines multiple decision trees (weak learners) to produce a robust and generalized model. Uses bagging (Bootstrap Aggregation) to train each tree on a random subset of the data.

III. **Random Subset Selection:** At each split in a decision tree, a random subset of features is considered. It introduces diversity among trees, reducing correlation between them and improving overall performance.

IV. **Voting/Averaging:** In classification tasks, each decision tree in the forest produces a class prediction which is called voting. The final output of the Random Forest is determined by a majority vote among all the trees. Each tree "votes" for a class label. The class with the most votes across all trees is the final prediction. For an example:

If a Random Forest has 5 trees, and they predict:

Tree 1: Class A

Tree 2: Class A

Tree 3: Class B

Tree 4: Class A

Tree 5: Class B

The final output will be **Class A**, as it has the majority vote.

Averaging is for regression. In regression tasks, each decision tree predicts a numeric value. The final output of the Random Forest is the average of the predictions from all trees.

If a Random Forest has 5 trees, and they predict:

- Tree 1: 10.5
- Tree 2: 12.0
- Tree 3: 9.8
- Tree 4: 11.7
- Tree 5: 10.2

The final prediction will be the average:

$$Y = \frac{10.5+12.0+9.8+1\ .7+10.2}{5}$$

=10.84

V. **Out-of-Bag (OOB) Error:** Since each tree is trained on a bootstrap sample, some data points (about 37% of the dataset) are left out. These out-of-bag samples can be used for validation to estimate model accuracy without requiring a separate validation set.

VI. **Feature Importance:** Random Forest provides a measure of feature importance by evaluating the contribution of each feature to the model's predictive power. This measurement basically based on average decrease in node impurity across the decision trees.

VII. **Robustness and Randomness:** Random Forest is inherently robust to noise and outliers due to its ensemble approach. Its randomness (in bootstrap samples and feature selection) reduces the risk of overfitting and increases generalization to unseen data.

These advantages make Random Forest a popular choice for machine learning practitioners and researchers in a variety of applications. However, there are some limitations in Random Forest application. Those are:

- **Overfitting in Single Decision Trees:** Individual decision trees are prone to overfitting, but Random Forest mitigates this by combining multiple trees.
- **Sensitivity to Feature Scaling:** Random Forest does not require normalization or scaling, unlike many other algorithms.
- **Bias-Variance Trade-off:** Random Forest strikes a balance between bias and variance by averaging multiple diverse trees.

7.3.6.2.1 Workflow of Random Forest algorithm

Here is a step-by-step breakdown of how the Random Forest algorithm works. The workflow in Figure 7.5 [142] simplifies the Random Forest algorithm process for regression analysis.

- Data Preparation: Collect and clean data, handle missing values, and split into training/testing sets.
- Model Initialization: Set hyperparameters like number of trees, max depth, and feature selection.
- Training: Create multiple decision trees using bootstrap sampling. Each tree selects a random subset of features at each split.
- Prediction: Each tree predicts a value, and the final output is the average of all tree predictions.
- Model Evaluation: Use metrics like MSE, RMSE, MAE, and R^2 to assess performance. Compute Out-of-Bag (OOB) error for internal validation.
- Hyperparameter Tuning: Optimize parameters using grid search or cross-validation.
- Model Interpretation & Deployment: Analyse feature importance, deploy the model, and monitor performance.

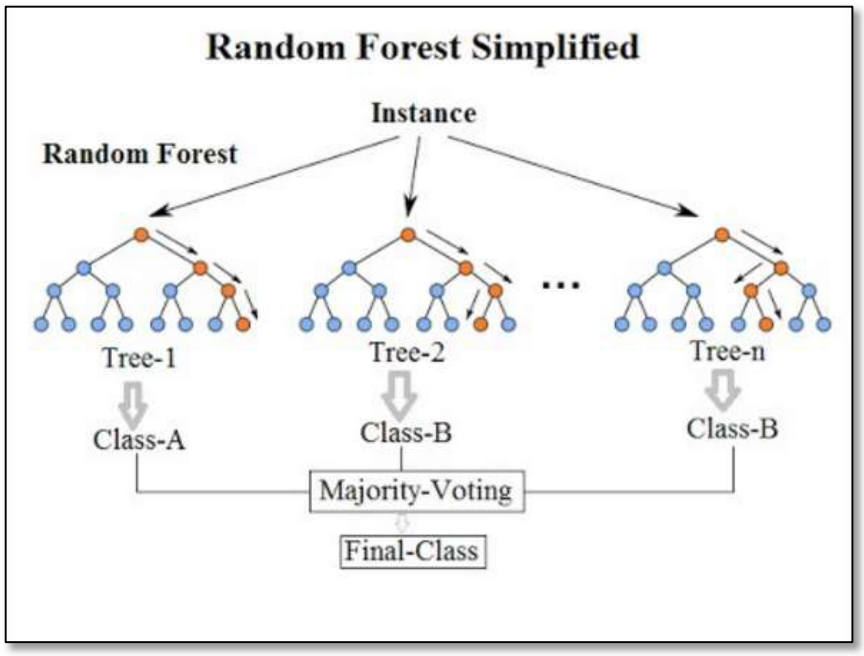

Figure 7.5 Workflow of Random Forest

7.3.6.3 Support Vector Regression (SVR)

Support Vector Regression (SVR) [143] [144]is a supervised machine learning algorithm that extends the principles of Support Vector Machines (SVM) for regression tasks. SVR works by finding the best-fit line (or hyperplane in higher dimensions) within a margin of tolerance, known as the epsilon (ε) tube. This makes SVR particularly effective in handling both nonlinear and linear regression problems while maintaining robustness against outliers.

Support Vector Regression (SVR) incorporates several key concepts that make it effective for regression tasks. The epsilon (ε) margin is a margin of tolerance around the true target values within which predictions are considered acceptable, meaning deviations within this margin are not penalized. To handle complex, nonlinear relationships, SVR uses kernel functions such as linear, polynomial, or radial basis function (RBF) kernels, which allow the data to be mapped into higher-dimensional spaces where linear separation is possible. Support vectors play a critical role in SVR, as only the data points lying outside or on the margin contribute to defining the model, thereby improving efficiency and reducing overfitting. Additionally, the regularization parameter (C) balances the trade-off between model complexity and tolerance for errors. A smaller C allows for a wider margin with more tolerance for errors, promoting

generalization, while a larger C focuses on minimizing errors but can increase the risk of overfitting. These concepts collectively enable SVR to perform robust and flexible regression analysis.

SVR models the relationship between the input features x and the output y using a linear equation (eq. 7.7):

$$f(x) = w^T x + b \qquad \text{eq. 7.7}$$

Where, $f(x)$ is the predicted value; x is the feature vector; w is the weight vector.

SVR uses an epsilon-insensitive loss function (eq.7.8), where no penalty is applied for errors within a margin ϵ, but penalties are applied for deviations larger than ϵ [136] [145]

$$L(y, f(x)) = \begin{cases} 0, & if \mid y - f(x) \mid \leq \epsilon \\ \mid y - f(x) \mid - \epsilon & otherwise \end{cases} \qquad \text{eq.7.8}$$

7.3.6.3.1 Workflow of SVR

The workflow of Support Vector Regression (SVR) involves a series of steps to effectively model and predict continuous target variables. Below is a structured outline of the SVR workflow:

- Data Preparation
    - Collect, clean, and preprocess data (handle missing values, normalize features).
    - Split data into training and testing sets.
- Model Initialization
    - Choose the kernel function (linear, polynomial, RBF).
    - Set hyperparameters like C (regularization), ϵ (margin of tolerance), and γ (kernel coefficient).
- Training the Model
    - Fit the SVR model to the training data using the selected kernel.

- Optimize the function by minimizing errors within the ϵ\epsilon-insensitive tube.

- Prediction
  - Use the trained SVR model to predict continuous values for new data.
- Model Evaluation
  - Assess performance using MSE, RMSE, MAE, and R^2 score.
- Hyperparameter Tuning
  - Use grid search or cross-validation to optimize kernel type, C, ϵ, and γ.
- Model Interpretation & Deployment
  - Analyse support vectors' influence, deploy the model, and monitor predictions.

7.3.7 Model Performance Evaluation

The confusion matrix [146] is a widely used tool for evaluating classification performance. It provides a comprehensive summary of key classification metrics in a single representation, including accuracy, sensitivity, precision, and specificity. The matrix derives four essential measures: true positive (TP), true negative (TN), false positive (FP), and false negative (FN). These measures are further utilized to compute precision, recall, overall accuracy, F-measure, macro and weighted averages of the model shown in eq. (7.9–7.18).

$$Recall = \frac{TP}{TP+\mathrm{FN}} \quad \text{eq. 7.9}$$

$$Precision = \frac{TP}{TP+FP} \quad \text{eq. 7.10}$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+F} \quad \text{eq.7.11}$$

$$F1-score = \frac{2X\ PrecisionXRecall}{Precision+Recal} \quad \text{eq.7.12}$$

$$Macro\_Avg(Recall) = \frac{1}{N}\sum_{i=1}^{N} Recall_i \quad \text{eq. 7.13}$$

$$Macro\_Avg(Precision) = \frac{1}{N}\sum_{i=1}^{N} Precision_i \quad \text{eq.7.14}$$

$$Macro\_Avg(F1-score) = \frac{1}{N}\sum_{i=1}^{N} F1-score_i \quad \text{eq.7.15}$$

$$weighted\_Avg(Recall) = \frac{1}{N}\sum_{i=1}^{N} w_i \times Recall_i \quad \text{eq. 7.16}$$

$$weighted\_Avg(Precision) = \frac{1}{N}\sum_{i=1}^{N} w_i \times Precision_i \quad \text{eq.7.17}$$

$$weighted\_Avg(F1-score) = \frac{1}{N}\sum_{i=1}^{N} w_i \times F1-score_i \quad \text{eq.7.18}$$

Where, w_i is the weight of class, N is number of classes.

A comparative study is conducted with three performance evaluation metrics: coefficient of determination (R^2) [147]., Mean absolute error (MAE) [148] and Root Mean Square Error (RMSE) [148] for regression model to determine the most suitable regression model for density and contrast prediction. R^2 measures the proportion of variance in the target variable explained by the model where higher values of R^2 indicates best fit of the model. MAE represents the average of absolute differences between predicted and actual values, providing a straightforward measure of prediction accuracy. Whereas, RMSE calculates the square root of the average squared differences between predicted and actual values. Lower values for MAE and RMSE indicate best fit of the model as they represent smaller errors between the predicted and actual values.

I. Coefficient of Determination (R^2):

- **Definition:** Measures the proportion of variance in the dependent variable that is predictable from the independent variables.
- **Range:** $0 \leq R^2 \leq 1$, though it can be negative for models that fit worse than a horizontal line (mean of the data).
- **Interpretation:**
  - $R^2 = 1$: Perfect fit (model explains 100% of the variance).
  - $R^2 = 0$: Model explains no variance beyond the mean of the data.
  - $R^2 < 0$: Model performs worse than the mean prediction.

II. Root Mean Squared Error (RMSE):

- **Definition:** Measures the square root of the average squared differences between predicted and observed values.
- **Range:** RMSE ≥ 0where lower values indicate better model performance.
- **Interpretation:**
    - Indicates the standard deviation of the residuals.
    - Penalizes large errors more than small errors due to squaring.

III. Mean Absolute Error (MAE):

- **Definition:** Measures the average of the absolute differences between predicted and observed values.
- **Range:** MAE ≥0 where smaller values indicate better performance.
- **Interpretation:**
    - Provides an easily interpretable measure of average error magnitude.
    - Less sensitive to outliers compared to RMSE.

7.4 Result and Discussion

In this study, offset-printed color patches of CMYK are used, with the physical density and contrast of each color measured individually. A mobile camera is used to digitize the color patch samples. Image augmentation has been done before classification of colors. A conventional classifier Inception v3 is used to classify the colors. After that feature of these digital color patch samples are extracted individually in HSV color Channel. These features are needed for EDA analysis and regression analysis to predict the color contrast and density. Data augmentation has also been done before regression analysis. After these, samples are subsequently divided into training and testing datasets for the prediction of color contrast and density. Here classification and prediction results are discussed as follows.

7.4.1 Result and Discussion for Classification

The Inception V3 classifier is utilized to classify the four colors (C, M, Y, K). Before classification image augmentation has been done using rescale, shear, zoom and horizontal filp method. After classification, the accuracy and loss graphs are analyzed to evaluate the model's performance. Accuracy is a key metric, represents the proportion of correctly classified examples out of the total number of examples. It is used to assess the model's effectiveness on both training and validation datasets during and after training. Loss, on the other hand, measures how far the predicted values deviate from the true labels. For classification tasks, it quantifies the difference between the predicted probability distribution and the true distribution. Figure 7.6 & 7.7 shows the accuracy and loss graph for the training and testing dataset for color classification. The blue line defines the training dataset whereas the red line depicts testing dataset. The graphs for the training and testing datasets closely align, with minor deviations in a few peaks from the training dataset. Moreover, the training and validation curves are showing small gap at the end the model which can be considered as generalized to avoid over- and under-fitting problems while classifying the unknown test data. The examples of classified color using Inception V3 form unknown color patch for C, M, Y and K is shown in Figure 7.8.

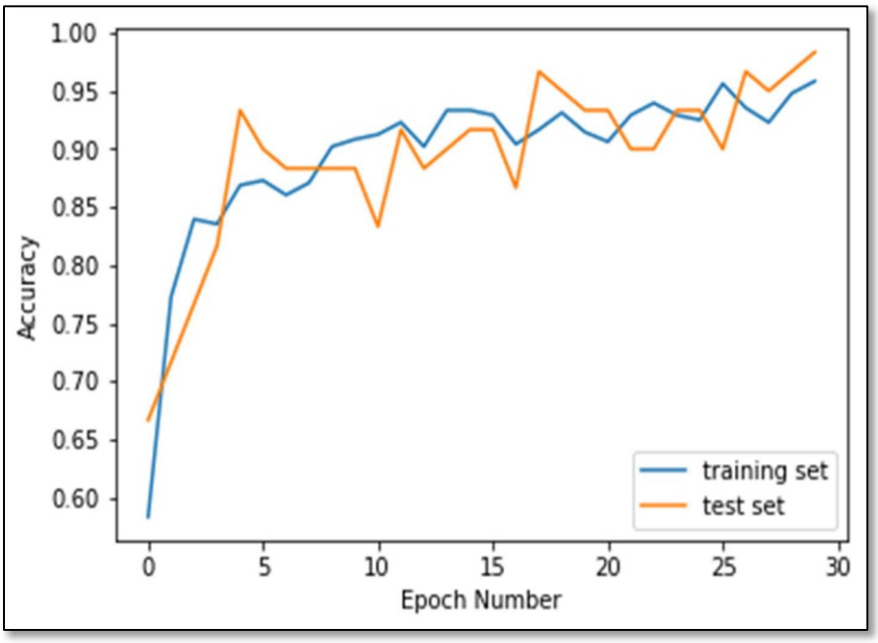

Figure 7.6 Color Classification Accuracy graph for Training and Testing Dataset

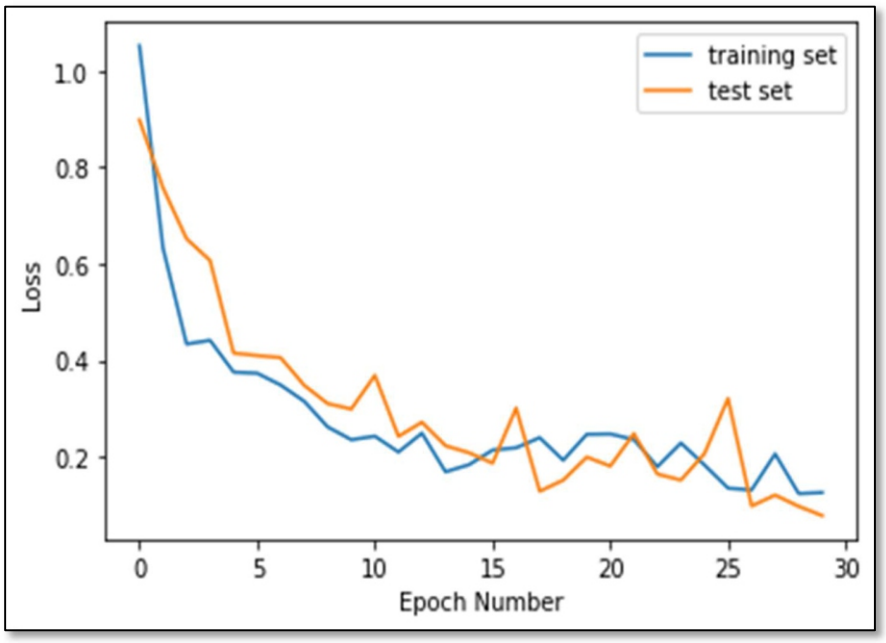

Figure 7.7 Color Classification Loss graph for Training and Testing Dataset

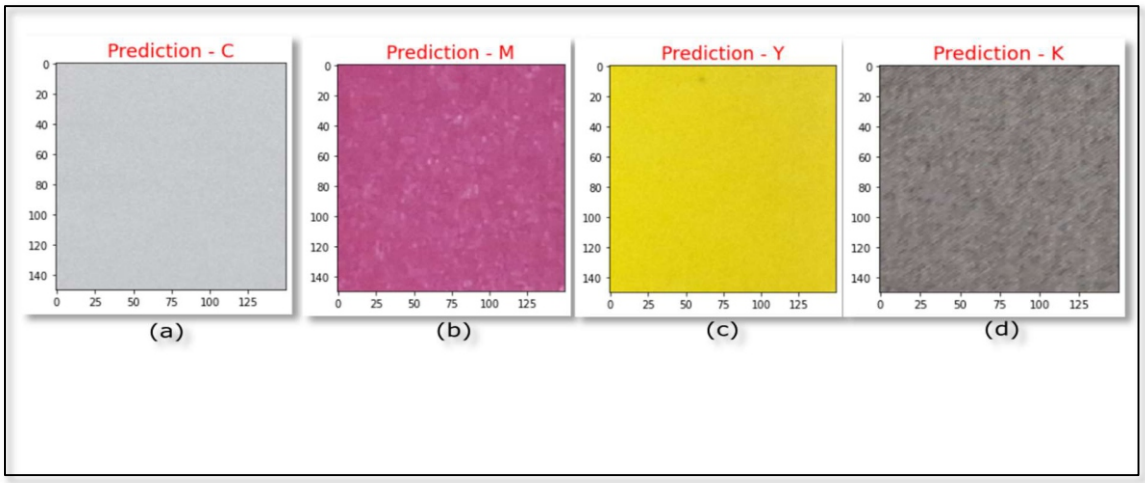

Figure 7.8 Examples of Inception V3 output for Color Classification

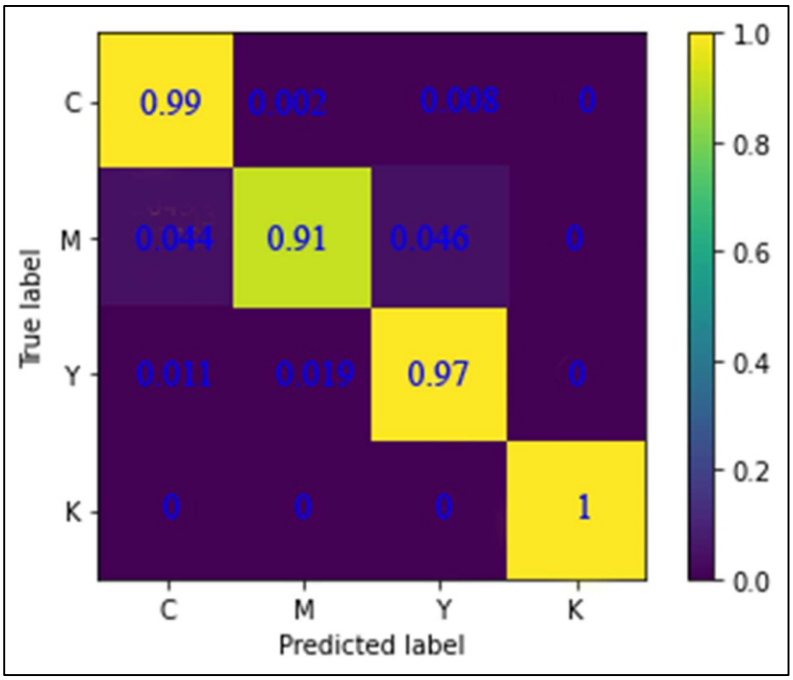

Figure 7.9 Confusion Matrix of Color Classification

In Figure 7.9 the normalized confusion matrix of color classification visualizes the performance of a classification model, where each cell represents the proportion of instances classified into specific categories, normalized so that the values sum to 1 across each row. The rows correspond to the true class labels, while the columns represent the predicted labels. Diagonal elements, shown yellow to green, indicate correct classifications, with values closer to 1 reflecting better model performance. For instance, the model demonstrates excellent performance across all four-color classifications—Cyan, Magenta, Yellow, and Black. For Cyan, the model predicts it correctly 99% of the time, with only minor misclassifications to Magenta and Yellow. Similarly, Magenta is correctly predicted 91% of the time, with slight misclassifications to Cyan and Yellow. Yellow is predicted with high accuracy of 97%, with only small errors occurring as misclassifications to Cyan and Magenta. Black is perfectly predicted, with all true Black labels correctly classified as Black. The model shows strong accuracy in distinguishing the different colors, with minimal confusion between Cyan, Magenta, and Yellow, and flawless performance in predicting Black. Overall, the model exhibits robust classification abilities with only negligible misclassifications.

***Table 7.1**: Classification Performance of Inception V3*

	Precision	Recall	f1-score
Cyan	0.96	1	0.98
Magenta	1	0.91	0.95
Yellow	0.96	1	0.98
Black	1	1	1
accuracy			0.98
macro avg	0.98	0.98	0.98
weighted avg	0.98	0.98	0.98

Performance of Inception V3 is depicted in Table 7.1 which summarizes the model's performance across four classes—Cyan, Magenta, Yellow, and Black—using three key evaluation metrics: Precision, Recall, and F1-score. The model predicted Cyan instances very accurately, with no false negatives. The model perfectly predicted the Magenta instances (Precision = 1), but 9% of actual Magenta instances are missed (lower Recall). Similar to Cyan, the model has no false negatives for Yellow. The model perfectly classified all Black instances without errors. The overall accuracy of the model is 0.98, meaning 98% of the predictions are correct. The weighted average of Precision, Recall, and F1-score across all classes is 0.98. This treats each class equally. The average Precision, Recall, and F1-score, weighted by the number of instances in each class, is also 0.98. The model performs exceptionally well, achieving high Precision, Recall, and F1-scores across all classes. The Magenta class has slightly lower Recall (0.91), indicating some instances of Magenta are misclassified, but overall, the model achieves excellent results with a balanced performance across all metrics.

7.4.2 Result and Discussion for Prediction

After classification, EDA analysis has been done to identify the features suitable for regression analysis for contrast and density prediction. In Figure 7.10,7.12,7.14 and 7.16 shows the scatterplot between features of cyan, magenta, yellow, black and target contrast respectively. And in Figure 7.22,7.24,7.26 and 7.28 shows the scatterplot between features of four colors and target density. Based on this analysis features are selected for regression analysis**.** Moreover, thresholding operations are also done to select the features from the heatmap (Figure 7.11,7.13,7.15,7.17 for contrast prediction and Figure 7.23,7.25,7.27,7.29 for density prediction) of correlation matrix between features and target contrast or density evaluated by

EDA. EDA ensures the data is well-understood, clean, and ready for subsequent analysis, helping to reduce errors and improve the reliability of results.

After EDA analysis, chosen features are employed to the prediction model for estimation of the color contrast and density of the color patches. For this study, three prediction models—Logistic Regression (LR), Support Vector Regression (SVR), and Random Forest (RF)—are applied for a comparative analysis for contrast and density prediction. The performances of these models are evaluated using three metrics: R^2 (Coefficient of Determination), Root Mean Square Error (RMSE), and Mean Absolute Error (MAE).

7.4.2.1 Contrast Prediction

7.4.2.1.1 EDA Analysis for Feature Selection (for Contrast)

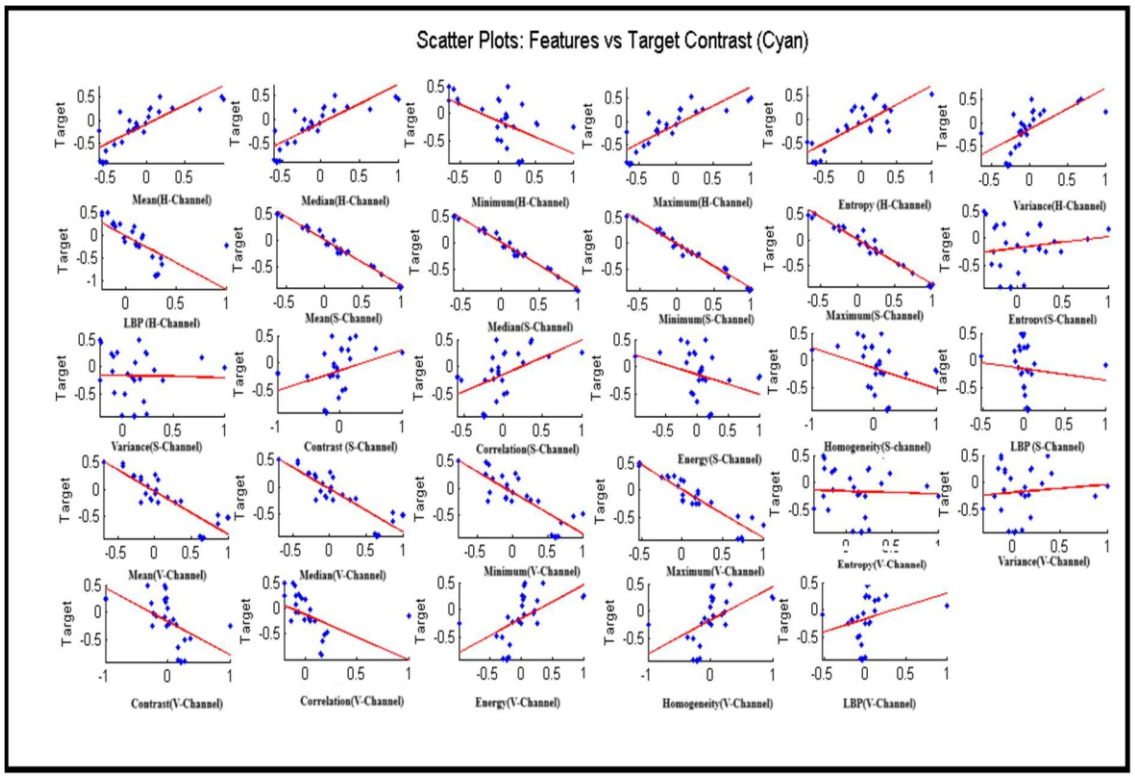

Figure 7.10 Correlation Scatterplot Between Cyan Color Target Contrast and Features

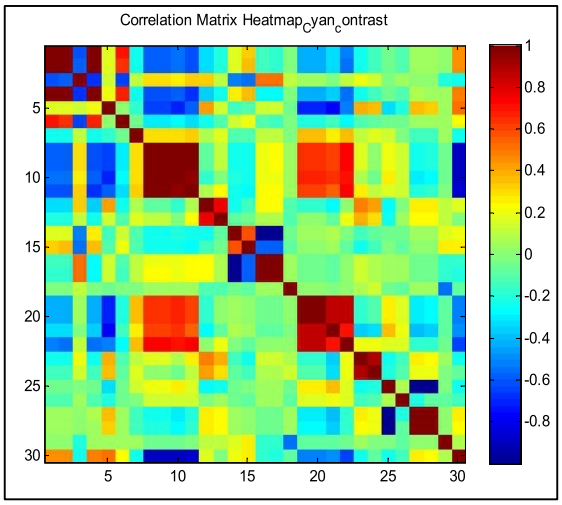

Figure 7.11 Heatmap for Cyan correlation matrix (contrast Prediction)

In Figure 7.10, correlation between target contrast and features for cyan color is depicted. This grid of scatter plots visualizes the relationship between various features (Feature 1 to Feature 29) and a target variable i.e. Contrast. Each subplot of scatter plot represents one feature on the x-axis plotted against the target contrast on the y-axis. The blue dots indicate individual data points, showcasing the variability of the features and target through their horizontal and vertical spreads. Overlaid on each scatter plot is a red line, representing a linear regression line fitted to the data points, which illustrates the trend or correlation between the feature and the target. These plots are used to visually assess the strength and direction of the relationships, identifying positive correlations with an upward-sloping red line (e.g. Mean in hue channel), negative correlations with a downward-sloping red line (e.g., Minimum in hue channel), and no correlations where the red line is approximately flat (e.g., variance in saturation channel). Scatter plot where datapoints are near the regression line strongly correlate with the target contrast. This analysis provides insight into how each feature contributes to or contrasts prediction with the target variable. Features such as Mean in hue channel, Median in hue channel, Maximum in hue channel, Entropy in hue channel, Variance in hue channel, and Contrast in saturation channel, Correlation in saturation channel, Energy in value channel, Homogeneity in value channel, LBP in value channel exhibit positive correlations, as indicated by the upward slope of the red regression line, suggesting that the target contrast tends to increase as these features increase. Conversely, features like Minimum in hue channel, LBP in hue channel, Mean in saturation Channel, Median in saturation channel, Minimum in saturation

channel, Maximum in saturation channel, Energy in saturation channel, Homogeneity in saturation Channel, Mean in value channel, Median in value channel, Minimum in value channel, Maximum in value channel, Contrast in value channel, Correlation in value channel, and Energy in value channel show negative correlations, with a downward slope of the red line, indicating that the target variable decreases as these features increase. Features such as Entropy in saturation channel, variance in saturation channel, LBP in saturation channel, Entropy in value channel and Variance in value channel show weak or no correlation, as the scatter points appear randomly distributed and the red line is nearly flat, suggesting little to no linear relationship with the target contrast. Now thresholding value 0.2 is applied on the heatmap of correlation matrix (Figure 7.11) by trial and error to select the suitable features which is strongly correlated with the target contrast. Finally mean in hue channel, median in hue channel, Maximum in hue channel, Entropy in hue channel and variance in hue channel, Correlation in saturation channel, energy in value channel and homogeneity in value channel is selected for regression analysis.

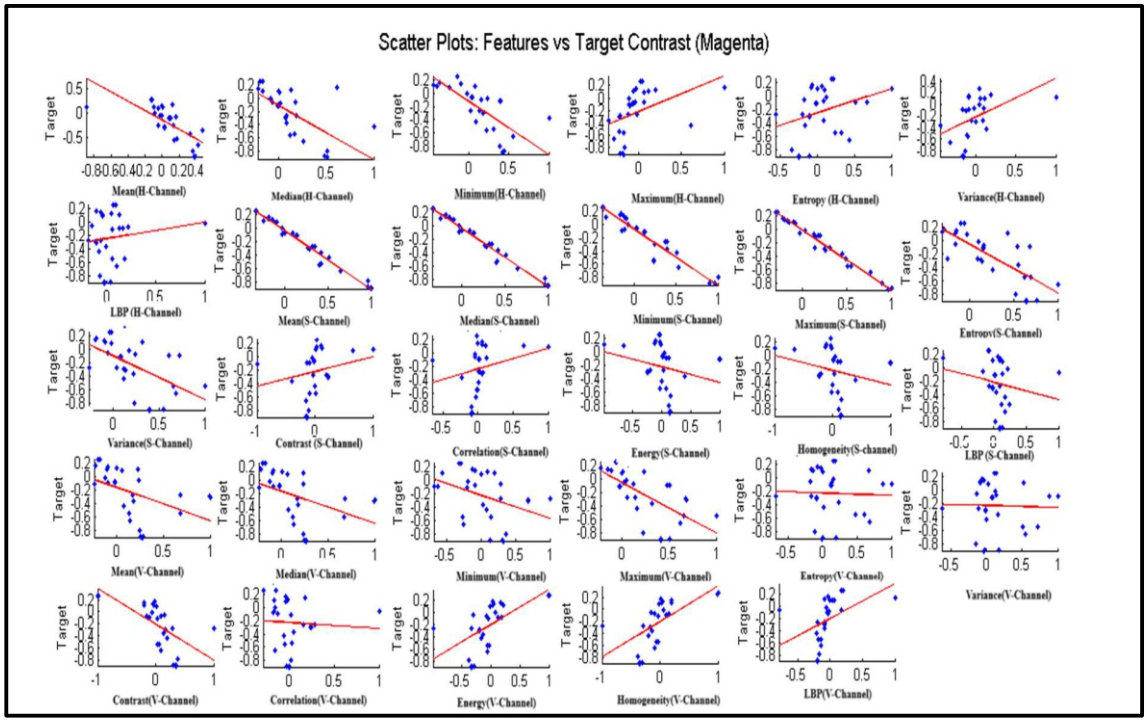

Figure 7.12 Correlation Scatterplot Between Magenta Color Target Contrast and Features

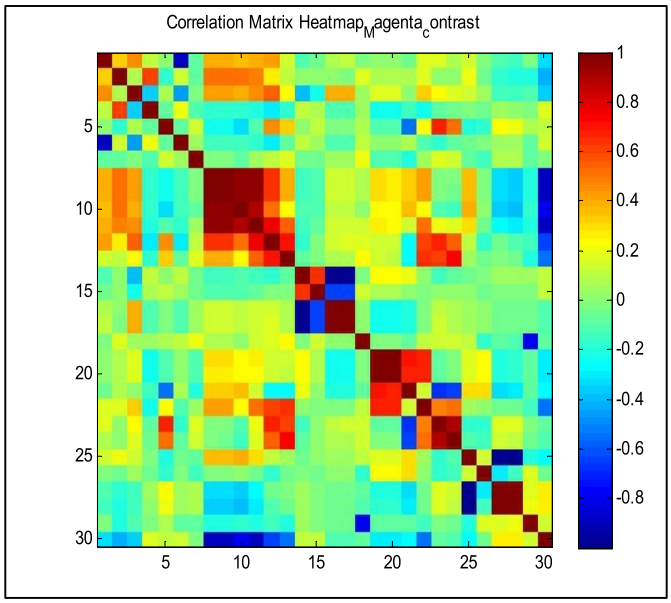

Figure 7.13 Heatmap for Magenta correlation matrix (contrast Prediction)

In Figure 7.12 the correlation between magenta color's target contrast and features is depicted. Now from the plots it is clearly visible that features like Maximum in hue channel, Entropy in hue channel, Variance in hue channel, LBP in hue channel, Contrast in saturation channel, Correlation in saturation channel, Energy in value channel, Homogeneity in value channel and LBP in value channel is positively correlated. On the other hand features like Mean in hue channel, Median in hue channel, Minimum in hue channel, Mean in saturation channel, Median in saturation channel, Minimum in saturation channel, Maximum in saturation channel, Entropy in saturation channel, Variance in saturation channel, Energy in saturation channel, Homogeneity in saturation channel, LBP in saturation channel, Mean in value channel, Median in value channel, Minimum in value channel, Maximum in value channel and Correlation in value channel are negatively correlated. Other features have little to no correlation. Now thresholding value 0.11 is applied on the heatmap of correlation matrix (Figure 7.13) by trial and error to select the suitable features which are strongly correlated with the target contrast. The selected features for this color are: Max in hue channel, Entropy in hue channel and Variance in hue channel, Contrast in saturation channel and Energy in saturation channel and Homogeneity in value channel and LBP in value channel. Features are selected by applying a channel-dependent threshold to retain only the strongly associated ones with target contrast, while further eliminating redundant features to ensure that the final set contributes distinct and complementary information.

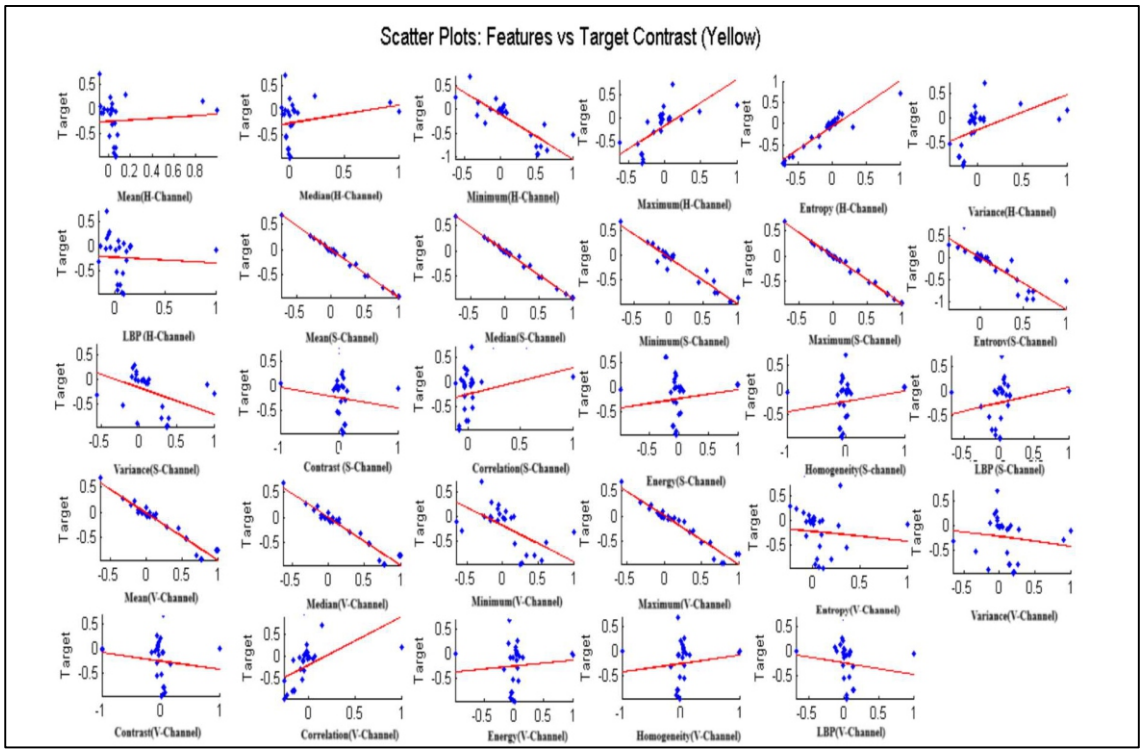

Figure 7.14 Correlation Scatterplot Between Yellow Color Target Contrast and Features

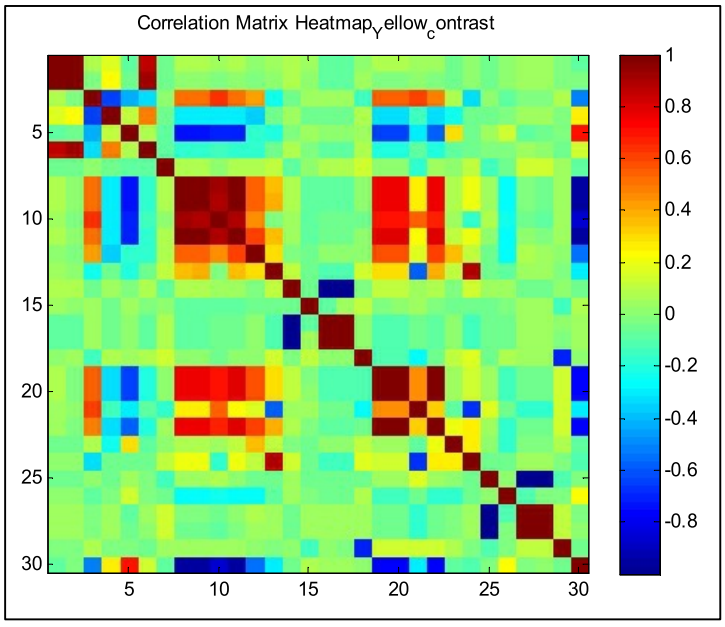

Figure 7.15 Heatmap for Yellow correlation matrix (contrast Prediction)

In Figure 7.14 correlation between target contrast and features for yellow color is shown. Where features like Maximum in hue channel, Entropy in hue channel, Variance in hue channel and Correlation in value channel is identified as positively correlated with target contrast whereas features like Minimum in hue channel, Mean in saturation channel, Median in saturation channel, Maximum in saturation channel, Entropy in saturation channel, Mean in value channel, Median in value channel, Minimum in value channel and Maximum in value channel exhibits negative correlation. For yellow color contrast prediction, features are selected by thresholding (0.2) by trial and error from the heatmap of correlation matrix (Figure 7.15) are: Maximum in hue channel and Entropy in hue channel and Correlation in value channel.

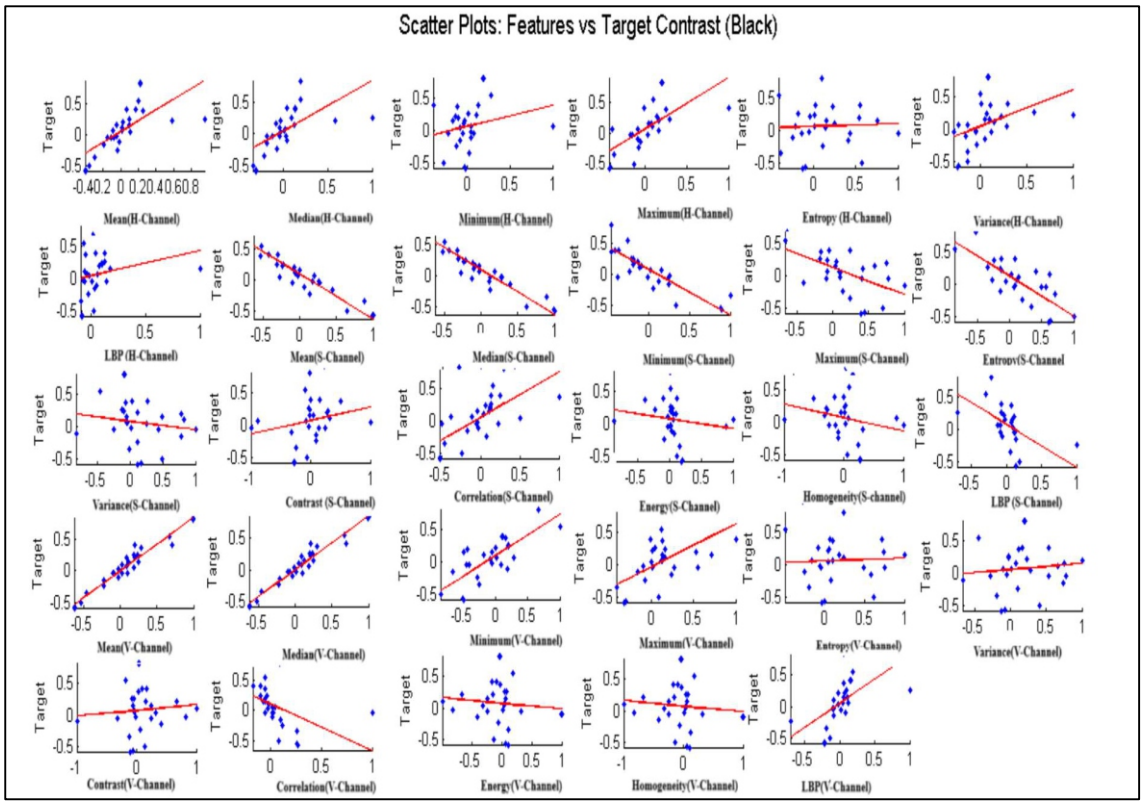

Figure 7.16 Correlation Scatterplot Between Black Color Target Contrast and Features

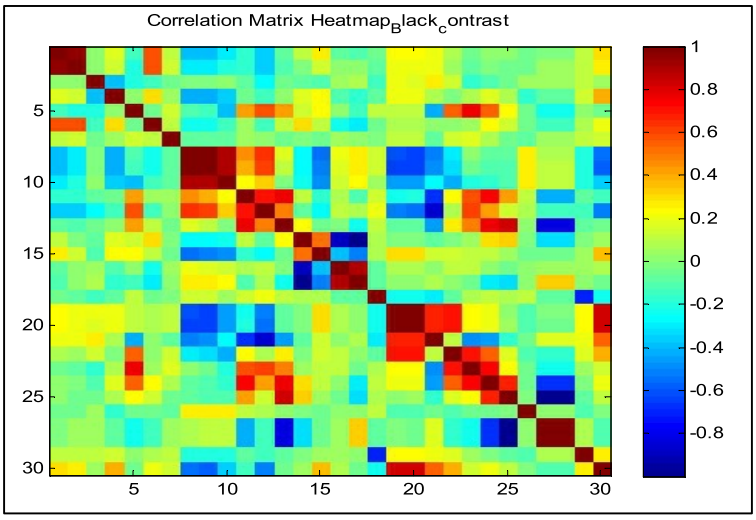

Figure 7.17 Heatmap for Black correlation matrix (contrast Prediction)

In Figure 7.16, target contrast and features correlation plot is depicted for black color. Features like Mean in hue channel, Median in hue channel, Minimum in hue channel, Maximum in hue channel, Variance in hue channel, LBP in hue channel, Correlation in saturation channel, Mean in value channel, Median in value channel, Minimum in value channel, Maximum in value channel and LBP in value channel is shown as a positive correlator. Whereas features like Mean in saturation channel, Median in saturation channel, Minimum in saturation channel, Maximum in value channel, Entropy in value channel, LBP in saturation channel and Correlation in value channel depicts as negative correlator. From the heatmap (Figure 7.17), by applying threshold value 0.3 by trial and error, features selected for regression analysis are: maximum in hue channel, correlation in saturation channel, Mean in value channel, Median in value channel, Minimum in value channel and Maximum in value channel.

7.4.2.1.2 Result of Contrast Prediction

From Figure 7.18, 7.19, 7.20, and 7.21, the results for cyan, magenta, yellow and black contrast prediction results for three prediction model are shown respectively, contrast prediction indicate that the RF prediction model performs comparatively better than the other two models in predicting contrast.

In case of cyan, the area plot for the prediction of contrast (Figure 7.18) is evaluated using three performance metrics which is R^2, RMSE and MAE. R^2 indicates how well the model explains the variance in data. Higher values are better for this metric. Whereas, RMSE represents average magnitude of errors and MAE represents the average absolute error. Both these metrics values are better if low. From the prediction graph (Figure 7.18) it is shown that area under R^2 in Random Forest (RF) model is high whereas, in other two model i.e. Linear Regression (LR) and Support Vector Regression (SVR), area of R^2 is low for both training and testing datasets. In case of RMSE and MAE metric, RF shows the best performance with lower value whereas, LR and SVR performance is not well as it shows highest errors in terms of surface area.

In magenta contrast prediction (Figure 7.19), the performance metrics, area under error metrics should be lower and area under R^2 should be higher. The figure shows that the Random Forest (RF) model achieves the highest area under R^2 value compared to Support Vector Regression (SVR) and Linear Regression (LR), indicating that RF provides the best fit to the data. Additionally, RF exhibits the lowest area under RMSE and MAE values for both training and testing datasets, whereas LR and SVR display the highest area for these error metrics. This suggests that the predictions made by LR and SVR deviate more significantly from the desired values compared to RF. Therefore, RF can be considered the most suitable model for predicting magenta contrast.

In yellow contrast prediction (Figure 7.20) RF consistently has the highest R^2, whereas other two model shows the lowest values for this metric. In case of RMSE and MAE, RF shows lower value compared to LR and SVR proving its strength as the prediction model for contrast.

In case of black color contrast prediction (Figure 7.21) among these three-prediction model SVR shows the moderate performance where area of R^2 is high compare to LR and area under error metrics is low compare to LR, RF achieves the highest R^2 values in terms of area and minimum area in error metrics, indicating the best fit to the dataset during both training and testing phases.

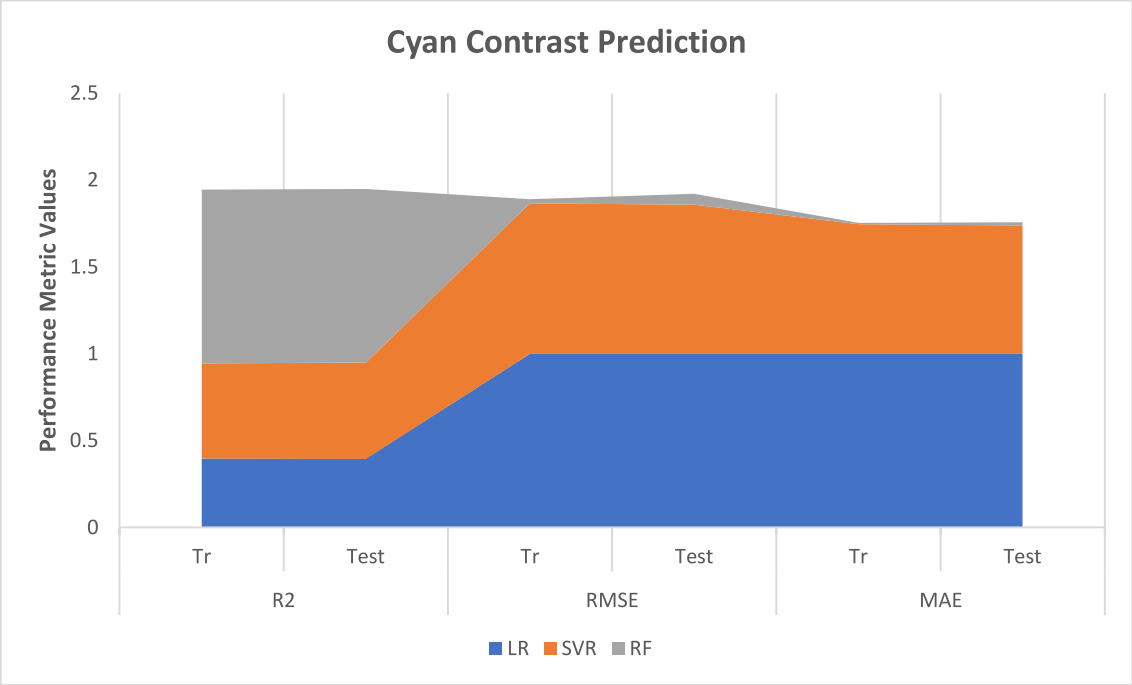

Figure 7.18 Area Plot of Performance Metrics for Contrast Prediction (Cyan)

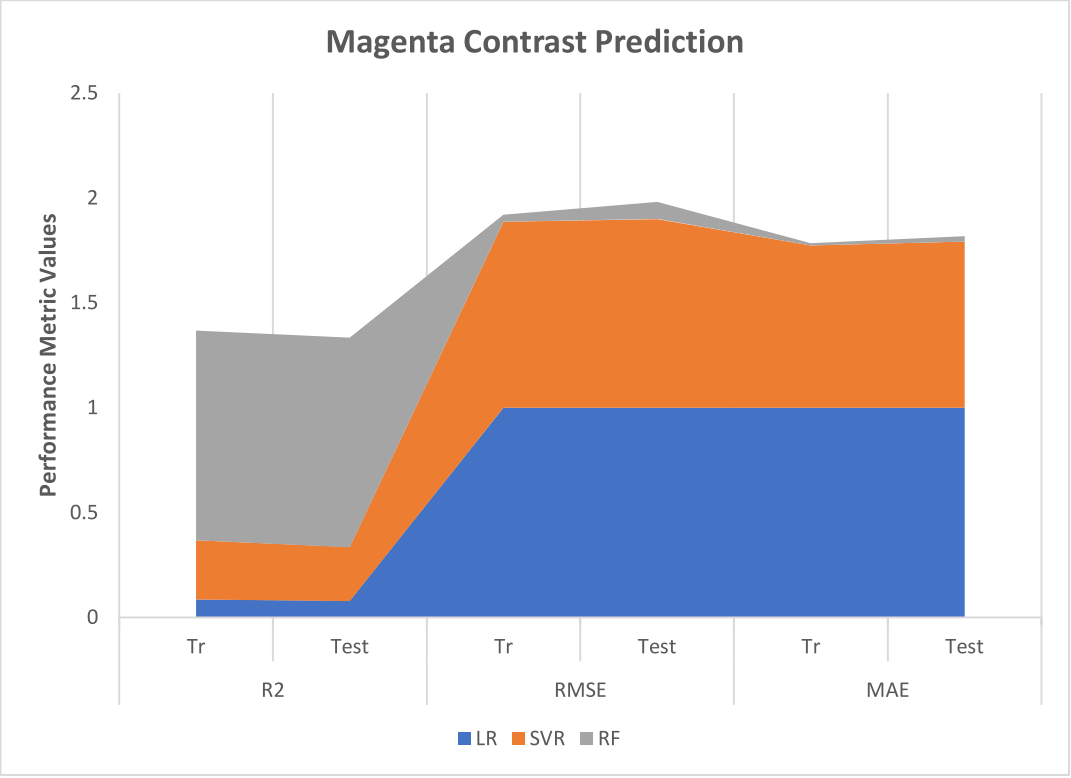

Figure 7.19 Area Plot of Performance Metrics for Contrast Prediction (Magenta)

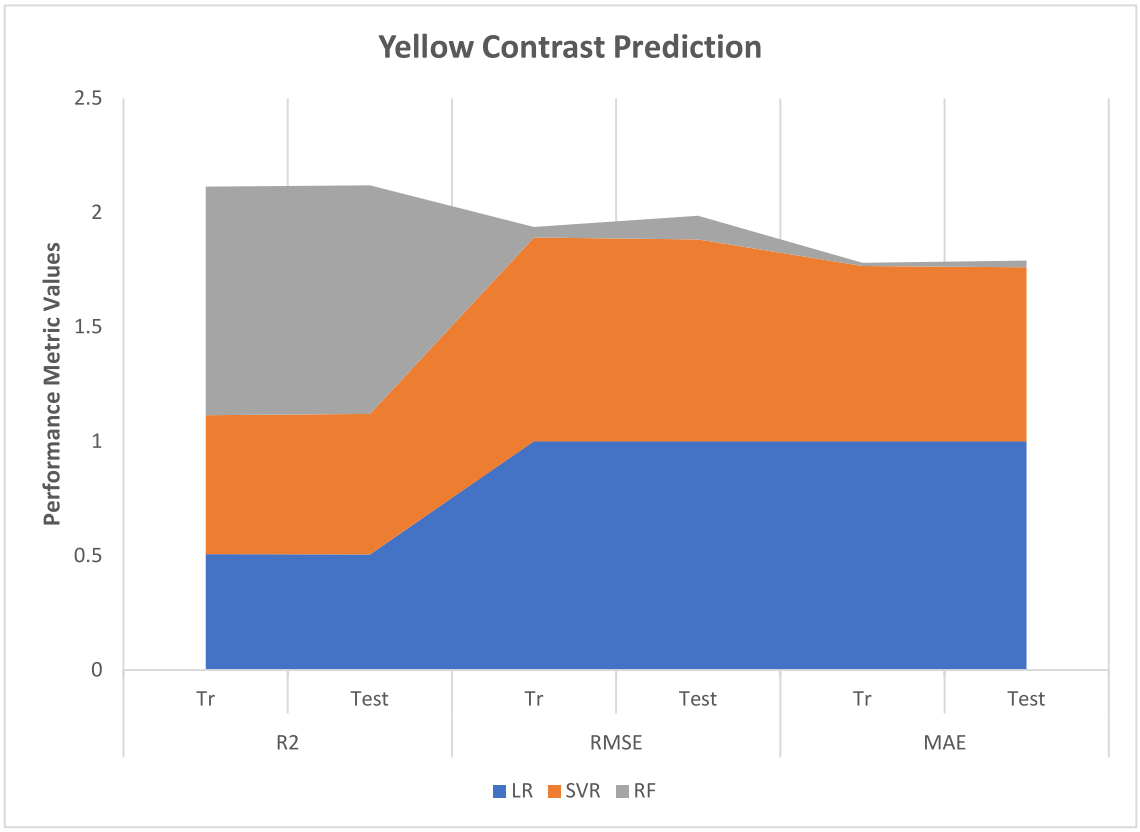

Figure 7.20 Area Plot of Performance Metrics for Contrast Prediction (Yellow)

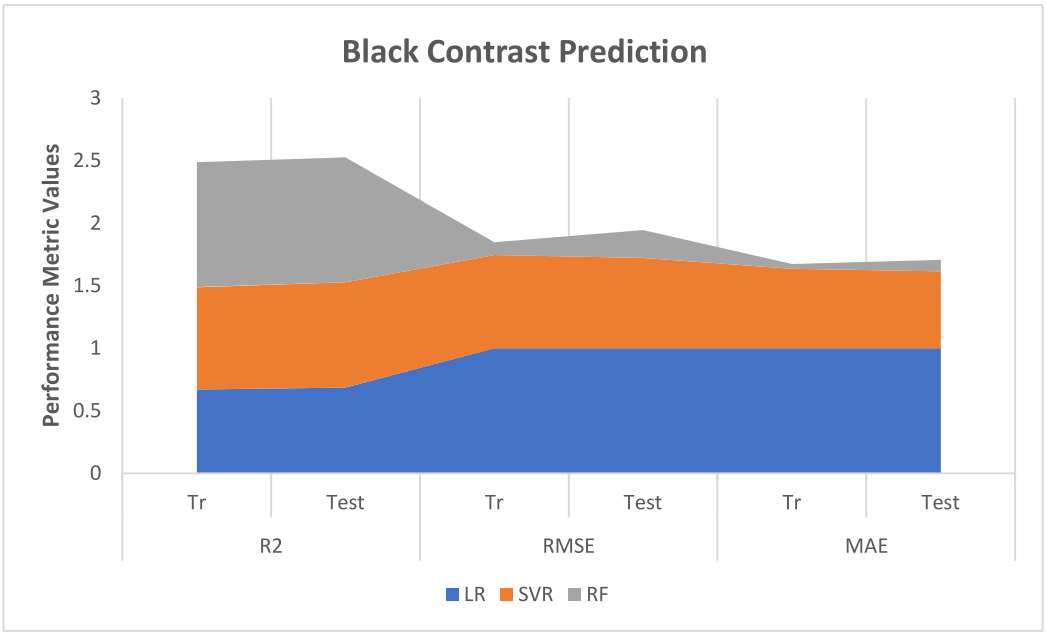

Figure 7.21 Area Plot of Performance Metrics for Contrast Prediction (Black)

7.4.2.2 Density Prediction

7.4.2.2.1 EDA Analysis for Feature Selection (for Density)

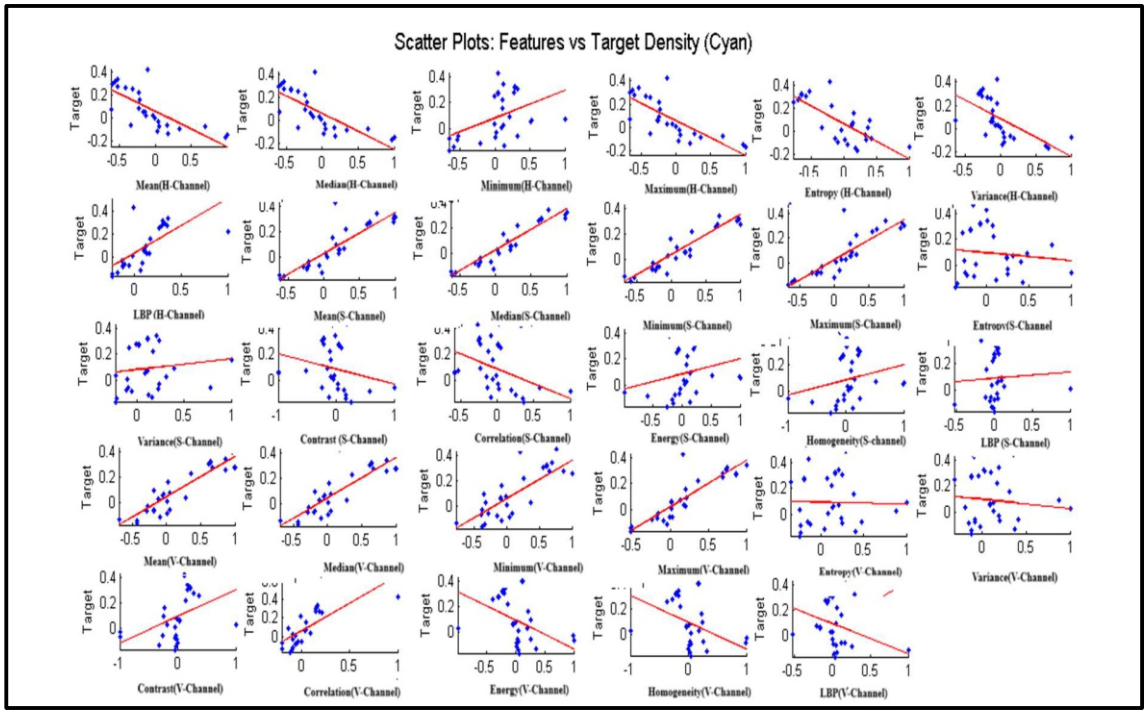

Figure 7.22 Correlation Scatterplot Between Cyan Color Target Density and Features

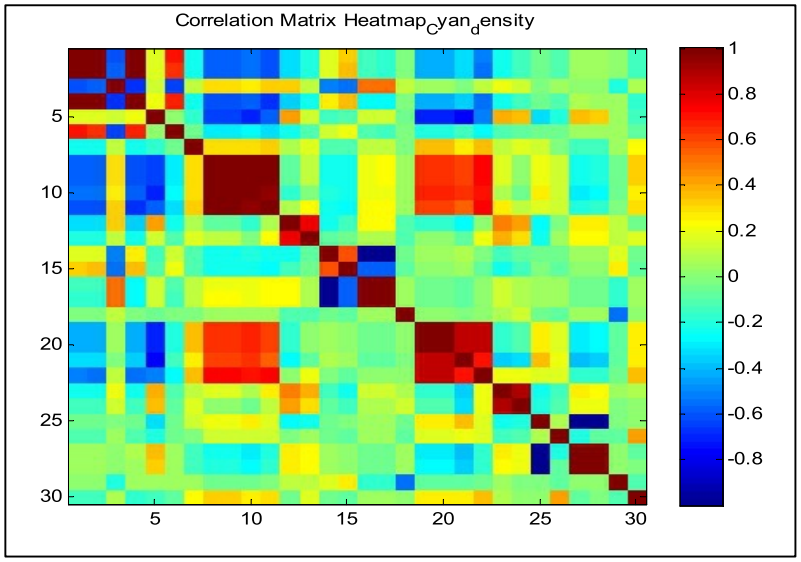

Figure 7.23 Heatmap for Cyan correlation matrix (Density Prediction)

In Figure 7.22 correlation plots are depicted for cyan density where correlation between target density and features are plotted. Here from the depicted plot, positive correlation is found in features like Minimum in hue channel, LBP in hue channel, Mean in saturation channel, Median in saturation channel, Minimum in saturation channel, Maximum in saturation channel, Energy in in saturation channel, Homogeneity in saturation channel, Mean in value channel, Median in value channel, Minimum in value channel, Maximum in value channel, Contrast in value channel, Correlation in value channel. Whereas features like Mean in hue channel, Median in hue channel, Maximum in hue channel, Entropy in hue channel, Variance in hue channel, Correlation in saturation channel, Energy in value channel and LBP in value channel has negative correlation. From the correlation heatmap (Figure 7.23) for cyan in density prediction, features selected for further operation by applying thresholding value 0.2 by trial and error are: LBP in hue channel, Mean in saturation channel, Median in saturation channel, Minimum in saturation channel, and Maximum in saturation channel, Mean in value channel, Median in value channel, Minimum in value channel, and Maximum in value channel and Correlation in value channel in value channel.

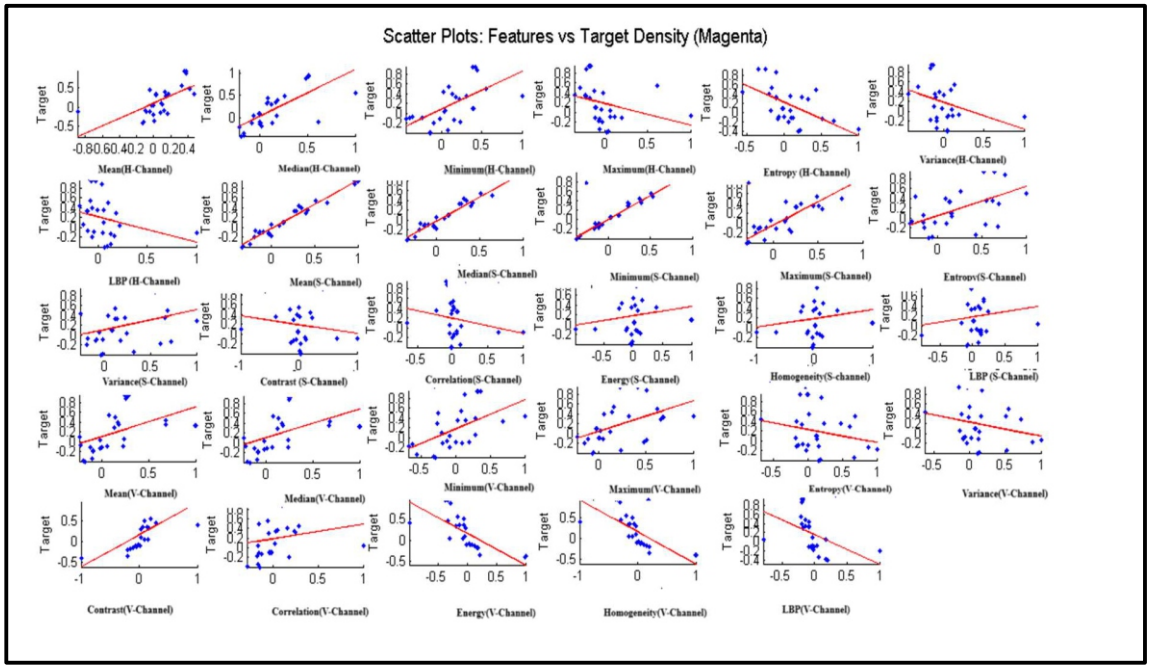

Figure 7.24 Correlation Scatterplot Between Magenta Color Target Density and Features

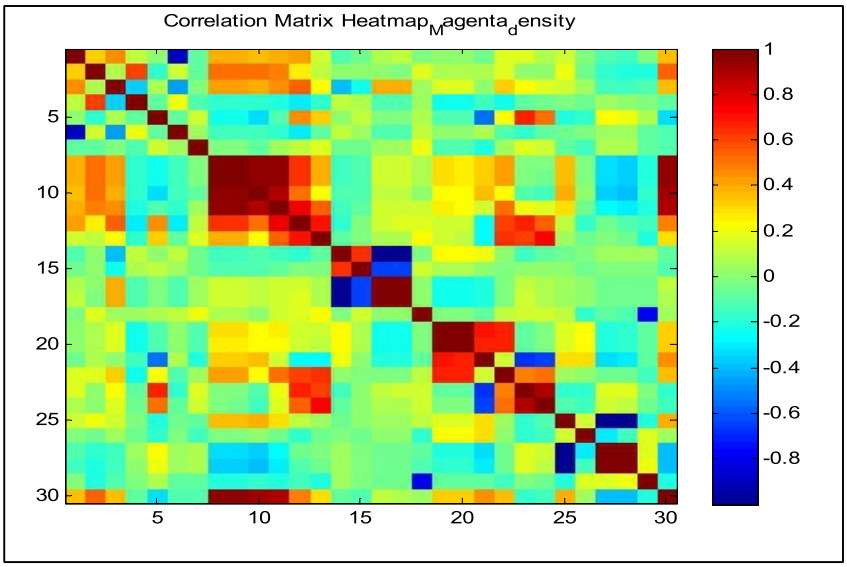

Figure 7.25 Heatmap for Magenta correlation matrix (Density Prediction)

Figure 7.24 depicts correlation scatter plot between target density and features of color magenta. From the plot positive correlation of features are identified. They are: Minimum in hue channel, LBP in hue channel, Mean in saturation channel, Median in saturation channel, Minimum in saturation channel, Maximum in saturation channel, Energy in in saturation channel, Homogeneity in saturation channel, Mean in value channel, Median in value channel, Minimum in value channel, Maximum in value channel, Contrast in value channel, Correlation in value channel. And the features which have negative correlation are: Mean in hue channel, Median in hue channel, Maximum in hue channel, Entropy in hue channel, Variance in hue channel, Correlation in saturation channel, Energy in value channel, Homogeneity in value channel, LBP in value channel. In heatmap of correlation matrix (Figure 7.25), threshold value 0.1 is applied after trial and error, to select features for regression analysis. Selected features are: Mean in hue channel, Median in hue channel, Minimum in hue channel, Mean in saturation channel, Median in saturation channel, Minimum in saturation channel, Maximum in saturation channel, Entropy in saturation channel, Variance in saturation channel and Contrast in value channel.

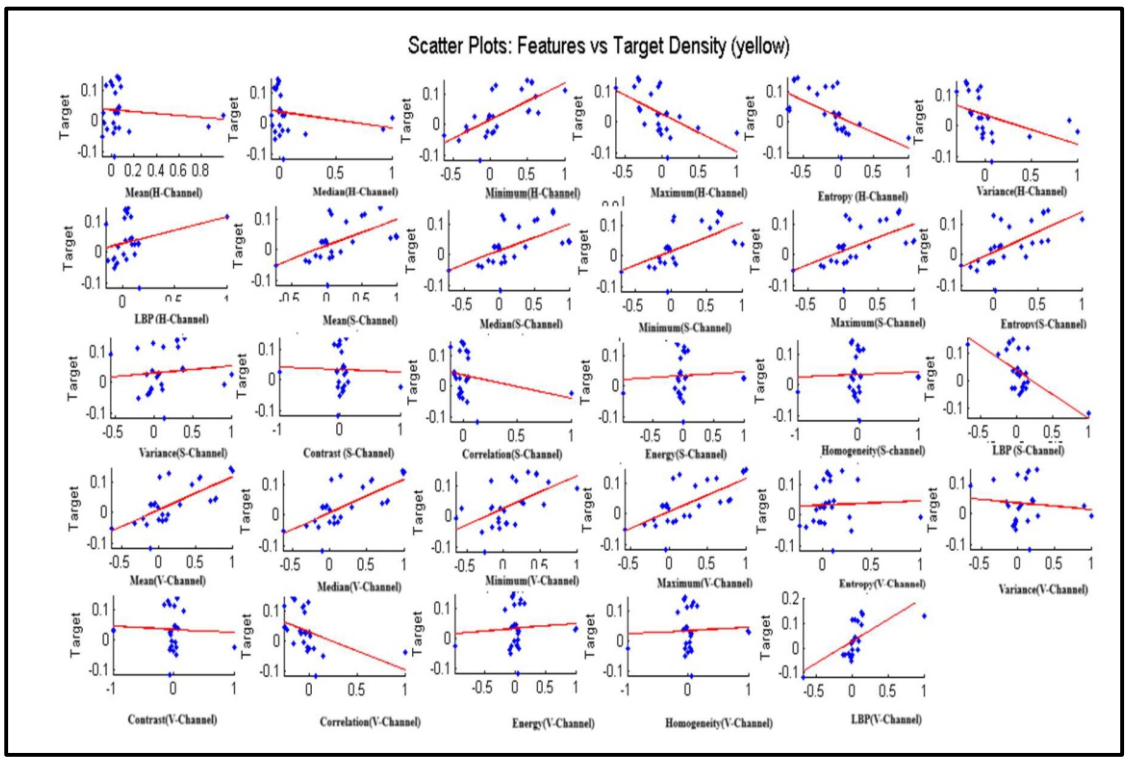

Figure 7.26 Correlation Scatterplot Between Yellow Color Target Density and Features

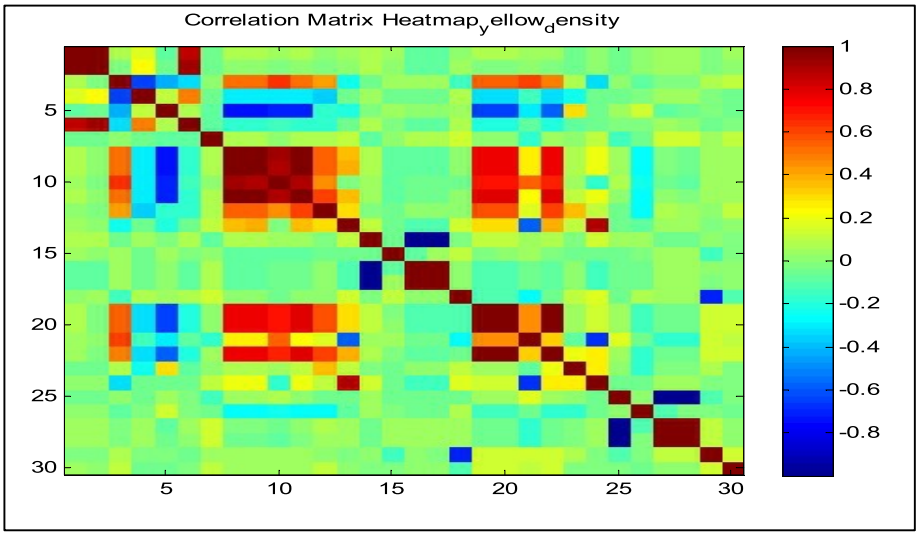

Figure 7.27 Heatmap for Yellow correlation matrix
(Density Prediction)

Figure 7.26 shows the correlation between target density and yellow contrast where it is found that features like Minimum in hue channel, LBP in hue channel, Mean in saturation channel), Median in saturation channel, Minimum in saturation channel, Maximum in saturation channel, Entropy in saturation channel, Mean in value channel, Median in value channel, Minimum in value channel, Maximum in value channel and LBP in value channel are having positive correlation with contrast. Whereas features like Maximum in hue channel, Entropy in hue channel, Variance in hue channel, Mean in saturation channel, Correlation in value channel have negative correlation. After trial-and-error method of selection the feature from correlation heatmap (Figure 7.27), threshold value is kept 0.1. Final selected features are: Minimum in hue channel and LBP in hue channel, Entropy in saturation channel, Mean in value channel, Median in value channel, Maximum in value channel and LBP in value channel in value channel. Other features have not any correlation and not selected for further operations.

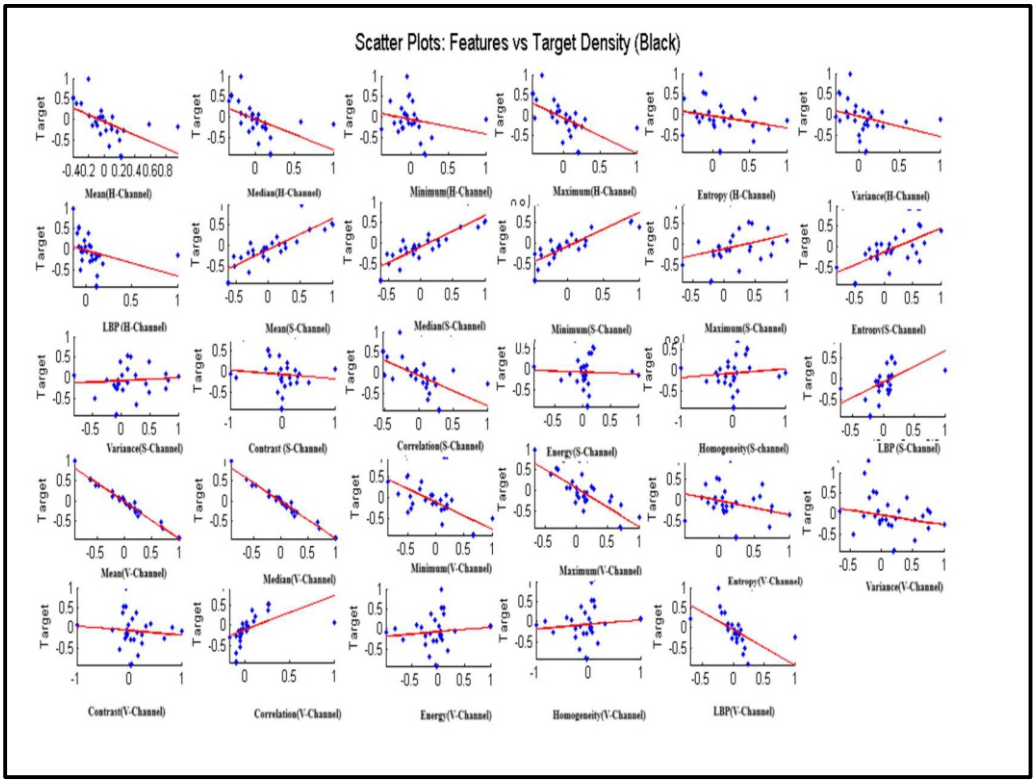

Figure 7.28 Correlation Scatterplot Between Black Color Target Density and Features

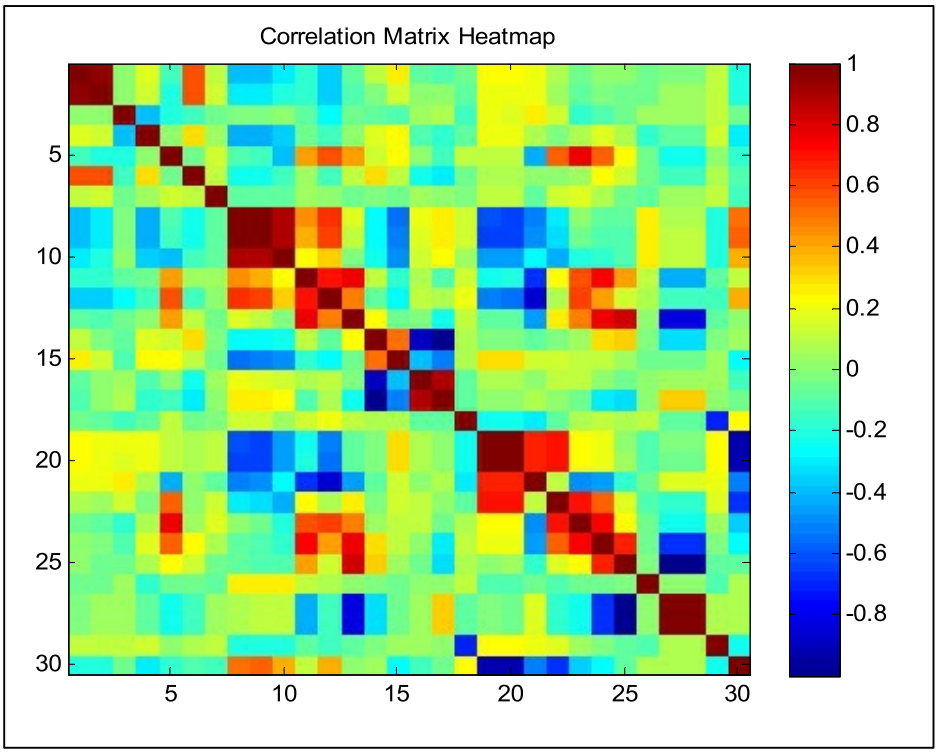

Figure 7.29 Heatmap for Black correlation matrix (Density Prediction)

In Figure 7.28 the correlation plot between target density and black color features are depicted. For this particular color from the scatter plot it is identified that features like Mean in saturation channel, Median in saturation channel, Minimum in saturation channel, Maximum in saturation channel, Entropy in saturation channel, LBP in saturation channel and Correlation in value channel are positively correlated with target density whereas features like Mean in hue channel, Median in hue channel, Minimum in hue channel, Maximum in hue channel, Entropy in hue channel, Variance in hue channel, LBP in hue channel, Correlation in saturation channel, Mean in value channel, Median in value channel, Minimum in value channel, Maximum in value channel and LBP in value channel are negatively correlated. Whether it is positive or negative these features have some impact to the target density. Therefore, by applying thresholding 0.2 in heatmap for black correlation matrix (Figure 7.29) after trial and error, strongly correlated features are selected, which are: Mean in hue channel, Median in hue channel, Minimum in saturation channel, Entropy in saturation channel and LBP in saturation channel.

7.4.2.2.2 Result of Density Prediction

For density prediction the same prediction model is used with the metrics to evaluate the performance of the particular model. Figure 7.30, 7.31, 7.32 and 7.33 shows cyan, magenta, yellow and black density prediction performance graph for the prediction model respectively.

In case of cyan density prediction (Figure 7.30), the area plots of R^2, RMSE, and MAE reveal that, RF demonstrates a strong ability to fit the training data with high R^2 values in terms of area and low error measurement (RMSE and MAE). Conversely, the Support Vector Regression (SVR) and Linear Regression (LR) models, characterized by lower area in R^2 and higher area in error measurements indicate that, they are not well-suited for predicting the density of cyan.

In Magenta density prediction (Figure 7.31) the area plots indicate that the Random Forest (RF) model outperforms the other two models. RF achieves the highest area under R^2 and the smallest area in RMSE and MAE, demonstrating its superior predictive performance. While the Support Vector Regression (SVR) model performs better than Linear Regression (LR). The area of R^2 is higher and area of error metrics is low compare to LR. But overall, the RF model clearly establishes its efficiency and reliability for predicting magenta density.

The area plot graph (Figure 7.32) for yellow density prediction shows that RF gives better result for R^2 , RMSE and MAE. Where R^2 value is high as expected and RMSE, MAE shows lower value indicating minimizing the error. Whereas the other two prediction model does not fit to the truth.

For black density RF gives better result showing the higher value for R^2 and lower for the error. SVR and LR does not appropriate for the prediction of density (Figure 7.33).

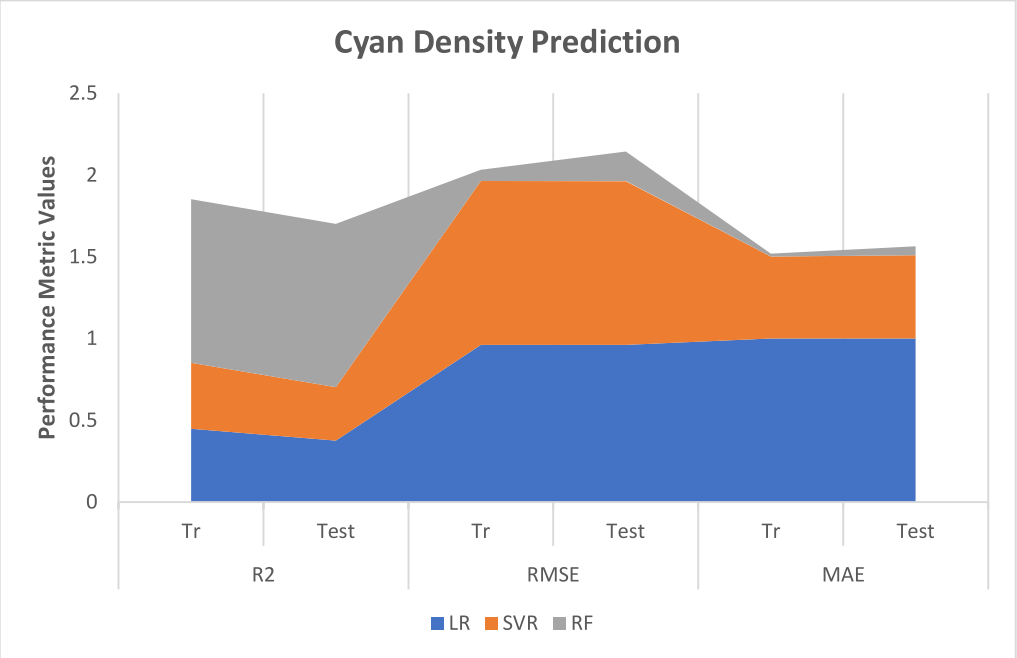

Figure 7.30 Area Plot of Performance Metrics for Density Prediction (Cyan)

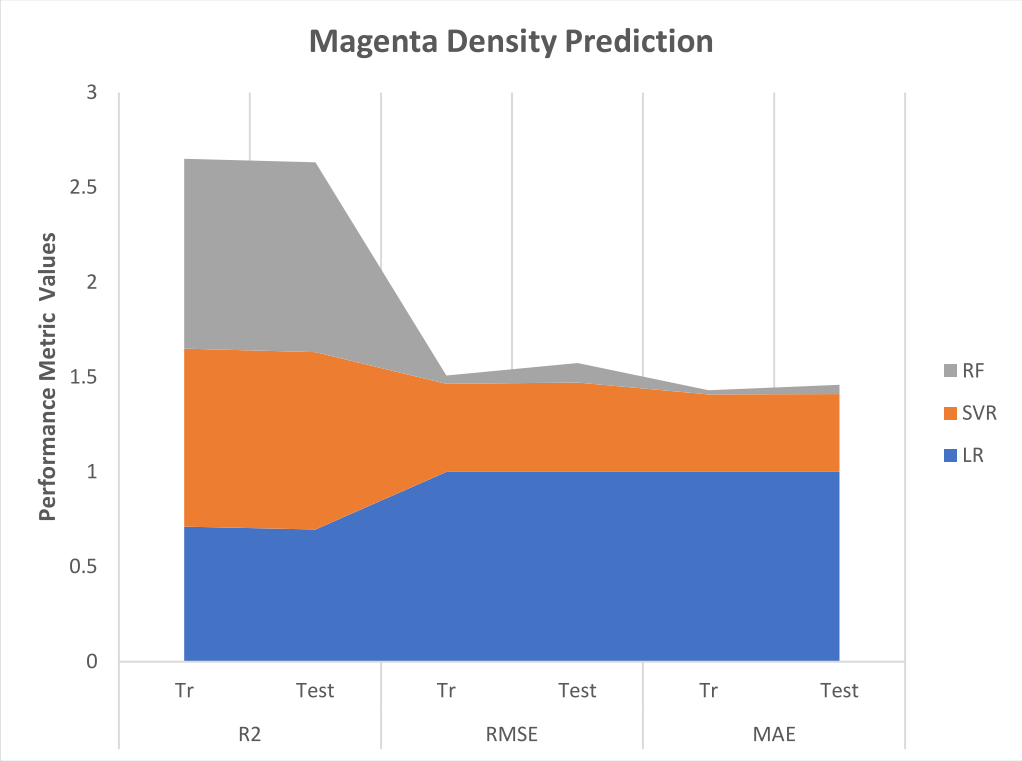

Figure 7.31 Area Plot of Performance Metrics for Density Prediction (Magenta)

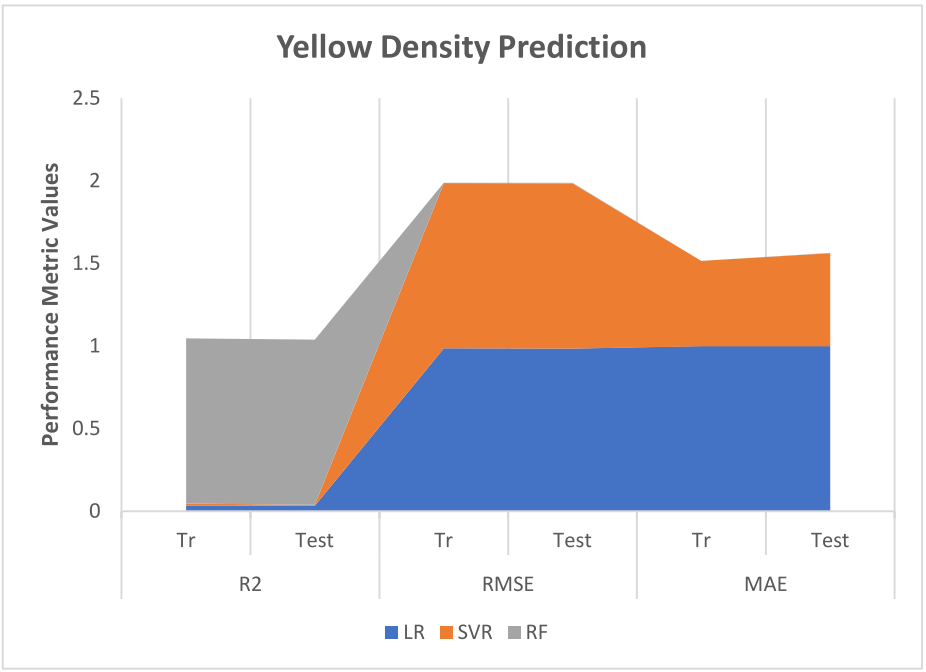

Figure 7.32 Area Plot of Performance Metrics for Density Prediction (Yellow)

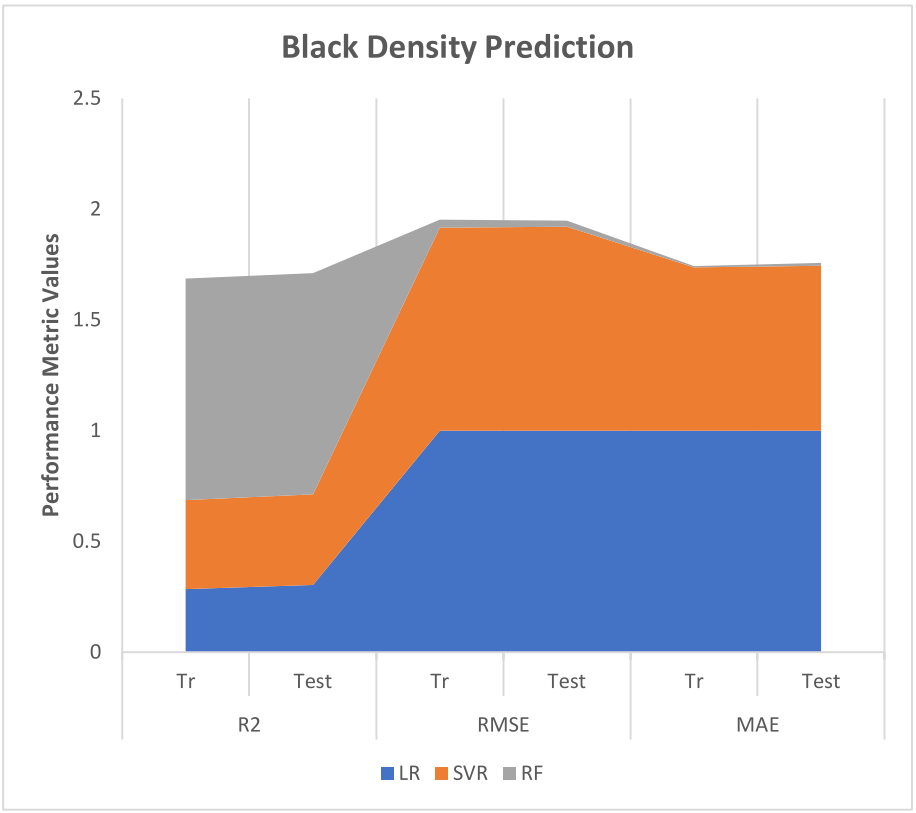

Figure 7.33 Area Plot of Performance Metrics for Density Prediction (Black)

7.5 Conclusion

This study aims to predict density and contrast from unknown color patches using a machine learning approach. The Inception V3 model is employed as a classifier, demonstrating excellent performance in predicting cyan, yellow, and black colors with high accuracy. However, the model exhibits slightly lower prediction performance for the magenta color. The accuracy and loss graph for training and testing dataset proves the efficiency of classification with minor deviation. For cyan, the model achieves 99% accuracy rate whereas for magenta there is 9% of misclassification. Yellow is classified truly 97% with minimal misclassification. Black is perfectly predicted, with 100% accuracy without any misclassification. Overall, the classifier performed well in classifying the cyan, yellow and black with perfect recall except magenta with lower recall.

Although there are minor misclassifications in the Magenta class, the model delivered outstanding performance with high and balanced metrics across all classes. It showcases excellent accuracy (98%) in distinguishing between Cyan, Magenta, and Yellow, with negligible confusion. Notably, its flawless prediction of Black underscores the model's robust classification capabilities.

For density and contrast prediction of colors, three performance metrics (R^2, RMSE and MAE) are used to compare the LR, SVR and RF regression model's prediction efficiency. Random Forest (RF) model outperformed the Logistic Regression (LR) and Support Vector Regression (SVR) models by evaluating their performance metrics for all the four colors (Cyan, Magenta, Yellow and Black). In performance metrics, area under error metrics should be lower and area under R^2 should be higher. This fact is also depicted in area plot graph for all four colors, where it shows Random Forest (RF) model achieves the highest area under R^2 and lower for error measurement, compared to other two regression model. Although SVR showed moderate performance in predicting contrast for black and predicting density for magenta, the RF model achieved the best overall results for both density and contrast prediction. The normalized data table also supports this fact and proves the strength of RF model for contrast and density prediction.

This study aims to perform the prediction of print color contrast and density with a mobile based approach by reducing the procurement and maintenance cost of machinery. It makes the procedure cost-effective as well as remotely controlled. The future scope of this work may be incorporation of combination of colors to achieve better prediction of color from unknown

patches. Furthermore, leveraging more complex regression models, such as neural network-based regressors, or Gaussian process regression, could provide superior predictive accuracy and robustness in estimating print contrast and density.

CHAPTER - 8

Discussion and Conclusion

CHAPTER- 8

Discussion and Conclusion

8.1 Introduction

Printing technologies have evolved significantly, driven by the digital revolution, necessitating advanced quality monitoring to meet customer expectations. Print quality, defined by visual and functional excellence, depends on factors such as substrate properties, ink formulation, process parameters, and environmental conditions. Traditional quality assessment methods, reliant on manual inspection and costly tools. Manual inspection has limitations in accuracy, efficiency, and scalability, especially in high-speed, large-scale production and costly instruments need huge initial investment and maintenance cost.

This study explores the application of computer vision (CV) techniques for automated print quality assessment and defect detection. Maintaining high-quality standards requires manufacturers to identify and rectify defects promptly during print production. Traditional quality control methods, often fall short of meeting the accuracy, speed, and consistency demanded by modern production environments. To address these challenges, advanced technologies like computer vision and deep learning have become indispensable, enabling printers to achieve precise, efficient, and reliable defect detection and correction.

Computer Vision leverages advanced algorithms to analyse visual data, offering objective, consistent, and repeatable evaluations while reducing human intervention. It efficiently identifies defects, such as scumming, showthrough, misregistration etc. and, and provides

insights into their correlation with substrate and ink properties. Compared to conventional methods, Computer Vision systems enhance processing speed, reduce operational costs, and deliver reliable quality control, representing a transformative solution for the printing industry.

8.1.1 Discussion on Scumming Printing Defect Detection

In offset lithography, the scumming print defect occurs when the non-image areas of the image carrier accept ink and transfer it to the blanket. This defect compromises the quality of the printed image, negatively impacting readability. Additionally, the presence of scumming pixels can interfere with archival processes, such as scanning and optical character recognition (OCR). Various press conditions contribute to this phenomenon. Typically, such printing issues are identified manually, and preventive measures are implemented based on the specific factors causing the defect. However, there is no standardized procedure for detecting scumming defects apart from visual assessment.

This study introduces a novel approach to address this limitation by proposing an automated defect detection system, which minimizes human error. The method employs a computer vision-based technique to evaluate scumming defects and quantify their occurrence, offering significant improvements over existing manual detection methods. For this study the real-life samples are used collected from offset presses and mobile camera is used for acquisition of those collected printed samples making the procedure a mobile based approach. Identifying scum pixels is challenging because the density difference between scum and print pixels is often minimal. Furthermore, pixel intensity variations across different types of images complicate the segregation process. To overcome this challenge here in this study a proposed range of intensity is introduced tailored for high-key, low key and mid key images which helps to segregate scumming pixel from print pixel more accurately. Moreover, application of bit plane slicing and automated bit plane selection helps to extract most relevant information from the bit plane for better identification minimizing the manual intervention. This study finds the suitable application of Discrete Cosine Transform and Inverse Discrete Cosine Transform to identify the print pixel discarding unwanted frequency components [105]. This combination of DCT and IDCT ensures a clean and accurate reconstruction of the print, enhancing accuracy of defect detection and maintaining the integrity of the printed content. Besides identification of scumming printing defect, quantitative measure of scumming in the printed samples, ensures consistency and accuracy in defect evaluation. The results demonstrate the effectiveness of the

proposed algorithm compared to current subjective methods ensuring that the proposed method is less time consuming in terms of computational load and is cost effective.

8.1.2 Discussion on Showthrough and Strikethrough Printing Defect Detection

A relatively simple approach [109] for detecting showthrough and strikethrough print defects using a computer vision method is presented. Showthrough and strikethrough are common printing issues, typically related to the opacity and porosity of paper. Under normal lighting conditions, showthrough refers to the visibility of printing on the reverse side of printed paper, while strikethrough occurs when ink penetrates through the paper to the other side. Mostly in printing presses this kind of defects are identified manually which depends on the operator's ability of identification and it can vary from man to man.

The intensity of showthrough pixels is extremely low, making them difficult to distinguish from printed areas. In contrast, strikethrough results from ink penetration and depends on the paper's absorption properties, has the similar intensity of printed pixels, causes them to overlap in the image foreground. These print defects can degrade both image and print quality.

In this study, these two print defects are detected using a histogram equalization technique which enhance the contrast between foreground and background pixels to identify defect pixels. It helps to distinguish the showthrough or strikethrough pixels from print pixels. A global thresholding algorithm is applied to the histogram-equalized image to segment the printed area from the background. This algorithm does not require any prior knowledge for segmentation; instead, it relies on pixel intensity to differentiate between the foreground and background. Since pixel intensity plays a crucial role in detecting these printing defects, the study evaluates the effectiveness of the global thresholding method. Image subtraction method proves its strength for segregation of defected pixels by utilizing the pixel intensity while preserving the other information Additionally, the properties of the substrate significantly influence the occurrence of these defects. The study reveals that when paper thickness is lower, showthrough increases—even if the paper is less porous [109]. This automated approach helps to discard manual intervention and ensures more consistent results in defect identification.

8.1.3 Discussion on Doubling Printing Defect Detection

A computer vision-based approach for print quality assessment, focusing on the detection of doubling print defects using the Structural Similarity Index Measure (SSIM) algorithm is presented [114]. Doubling is a common printing issue characterized by the appearance of a non-directional double image. Since doubling is an undesirable defect that impacts print quality, its detection is crucial for ensuring quality print production.

In this study, the doubling defect is identified using a computer vision-based method where the SSIM algorithm is applied to analyse print samples containing both text and images. Detecting the overlapped regions caused by doubling is challenging as overlapped element has similar intensity. The proposed approach compares the structural similarity of the overlapped edges to pinpoint the double impressions. To identify the overlapped region, pixel connectivity by using 8- connectivity is applied. It ensures that no overlapped regions are missed and helps to distinguish between well-printed regions and those affected by doubling defects.

Additionally, a comparative analysis is conducted with print quality assessment metrics, such as Structural Similarity Index Measure (SSIM), Mean Square Error (MSE) and Feature Similarity Indexing Method (FSIM). SSIM considers luminance, contrast and structural correlation of double printed edges to check the similarity. Whereas FSIM only finds feature-based similarity and less sensitive to contrast and luminance. MSE provides only the numerical measure of absolute error, does not reflect the perceptual quality or structural alignment of the images. Therefore, this study finds the effectiveness of SSIM algorithm, as the proper identification of doubling printing defects need to consider contrast and luminance information besides structural correlation. This study finds that if the similarity percentage of two edges of printed image is greater than or equal to 90% [114], they are considered structurally identical, for text-based image the structural similarity exceeded 95%, demonstrating high accuracy in detecting doubling defects. However, for image based doubling defects less accuracy is found in terms of similarity.

8.1.4 Discussion on Misregistration Printing Defect Detection

Misregistration, a common print defect in offset printing, impacts the visual quality of printed images. Registration ensures the accurate alignment of CMYK colors, while improper alignment of or shifting of colors results in blurred images. Traditionally, registration marks on

printed sheets are used as manual detection by linen glass of such issues. This study introduces a novel approach [117] by applying a computer vision-based technique to detect and quantify misregistration. The quantification of color shifting is achieved using Euclidean and Manhattan distance measurement methods. The result depicts not only the detection of misregistration also how much color is shifted is quantified. This innovative method offers a reliable alternative to conventional human perception-based techniques in the printing industry which makes the process cost effective, less time consuming and a mobile based approach.

In misregistration, mainly the shifting of colors is happened in CMYK color space. Therefore, to quantify and identify the misregistration RGB image has to be converted into CMYK color space. Now the challenge arises because direct translation between the two models is not feasible due to differences in their color gamut's, or the range of colors they can represent. RGB can produce more vivid and diverse colors than CMYK, making conversion essential for adapting digital designs for printing while maintaining as much color accuracy as possible. To address these challenges, ICC (International Color Consortium) profiles are employed to standardize color reproduction across different devices and ensure consistency. Otherwise, the converted CMYK color spaced image data may lose some information related to color information. The study finds that if there is a pixel-to-pixel distance between separated color channels in the CMYK color space, misregistration is present, even if it is sometimes undetectable to the naked eye. However, if there is no distance between the color channels, the colors are considered superimposed, indicating no misregistration in the printed image. Therefore, this automated approach [117] of detection and quantification of misregistration print defect is more reliable than manual identification. Though in registered sample, a minimal occurrence of misregistration is sometimes found but it is acceptable to some extent.

8.1.5 Discussion on Prediction of Print Contrast and Density Quality Parameter

Image quality depends on the ability to distinguish difference between density values within the image. When there are greater differences in density between midtones, highlights, and shadow tones, it becomes easier to see the details. The difference in tonal density is referred to as print contrast. Print contrast is determined by the whiteness and brightness of the paper, as well as the density of the ink. Insufficient paper opacity and the resulting excessive show-through can diminish print contrast that can affect overall print quality.

This study aims to predict density and contrast from unknown color patches using a machine learning approach. The Inception V3 model is utilized as a color classifier, demonstrating excellent accuracy in predicting cyan, yellow, and black colors. However, its performance is slightly lower for magenta. The accuracy and loss graphs for the training and testing datasets confirm the model's classification efficiency, with only minor deviations. The model achieves a 99% accuracy rate for cyan, while magenta has a 9% misclassification rate. Yellow is correctly classified 97% of the time with minimal errors, and black is perfectly predicted with 100% accuracy and no misclassification. Overall, the classifier performs exceptionally well with 98% accuracy in identifying cyan, yellow, and black, achieving perfect recall, except for magenta, which has a lower recall.

For predicting the print contrast and density, three performance metrics—R^2, RMSE, and MAE—are used to compare the prediction efficiency of the LR, SVR, and RF regression models. The Random Forest (RF) model outperformed both Logistic Regression (LR) and Support Vector Regression (SVR) based on performance metrics for all four colors: Cyan, Magenta, Yellow, and Black. In these metrics, a lower value for error measurements and a higher value for R^2 indicate better performance. This trend is also reflected in the area plot graph for all four colors, where the RF model achieves the highest area under R^2 and the lowest for error measurements compared to the other two regression models. Although SVR demonstrated moderate performance in predicting contrast for black and density for magenta, the RF model delivered the best overall results for both density and contrast prediction.

8.2 Scope of Future work

Future research can explore several directions to further explore the application of computer vision (CV) techniques in print quality assessment and defect detection. By addressing these areas, future work can further revolutionize quality control in the printing industry, making it more efficient, accurate, and sustainable.

These include:

I. **Integration of Machine Learning**: Incorporating advanced machine learning algorithms can significantly enhance defect classification accuracy by allowing systems to learn from diverse datasets and adapt dynamically to new printing scenarios. These algorithms enable

the identification of subtle patterns and complex relationships in data, improving the precision and reliability of defect detection in varied production environments.

II. **Real-Time Monitoring**: Developing CV systems capable of real-time analysis can optimize quality control during high-speed production processes, minimizing waste and improving efficiency.

III. **Expanded Defect Detection**: Broadening the scope of detectable defects to include complex issues like microstructural inconsistencies, ink adhesion failures, substrate deformation that may compromise print quality.

IV. **Integration with IoT**: Combining CV systems with IoT technology for remote monitoring and centralized quality control, enhancing operational flexibility.

V. **Multivariable Optimization**: Using CV data to optimize printing parameters, ink formulations, and substrate properties simultaneously, creating a comprehensive framework for process improvement.

VI. **User-Friendly Interfaces**: Designing intuitive user interfaces for CV systems to simplify operation and interpretation for non-experts.

8.3 Conclusion

This research primarily aims to enhance the efficiency and accuracy of print quality assessment and defect detection by leveraging computer vision techniques discarding manual detection process. When a printed sample is evaluated using this procedure and no defects detected, it can be concluded that the sample meets the quality standards for output printed product. Additionally, incorporating contrast and density prediction it enhances the accuracy of checking quality parameters, ensuring a more comprehensive quality assessment. By automating the conventional defect detection process, the study seeks to minimize human intervention, resulting in faster and more reliable evaluations of real-life print problems. This automation promises to boost productivity, reduce human error, and deliver consistent, repeatable results.

A key focus of the research is to develop methods that are both cost-effective and time-efficient while maintaining high standards of precision in print assessments. By prioritizing the computer vision-based techniques that require minimal resources, the study aims to make

advanced quality control methods accessible to a broader range of printing operations, fostering industry-wide optimization.

Moreover, the integration of a mobile-based approach introduces greater flexibility, enabling on-the-go monitoring of print quality. This innovation simplifies deployment in diverse environments and allows remote quality control, providing significant benefits in terms of convenience and scalability.

By addressing these goals, the research contributes to advancing print production quality control, showcasing the potential of computer vision technologies as transformative tools for the printing industry. The study underscores the feasibility and effectiveness of these technologies in enhancing efficiency, reducing costs, and improving print quality across various applications. It represents a pivotal step toward modernizing traditional practices, ensuring consistency and reliability in print production, and driving innovation in quality assessment methodologies.

REFERENCES

1. H. Kipphan, Handbook of Print Media (Springer Ed., 2001).
2. Barnard, M. “The Print and Production Manual.” 11th ed.
3. Leach, R. and Pierce, R. (2007). “The Printing Ink Manual. Springer Science & Business Media.”
4. Eldered, “Chemistry for The Graphic Arts”, GATF Press.,(2007)
5. indiamart.com.(2016). Disc Micrometer Available at: https://www.indiamart.com/proddetail/disc-micrometer-20399037130.html [Accessed 29 Dec. 2024. 9.25PM]
6. Rycobel. (n.d.). Gurley Densometer for measuring the porosity of materials. Available at: https://www.rycobel.com/products/gurley-densometer. [Accessed 29 Dec. 2024. 9.25PM]
7. Heilmann, J. and Lindqvist, U. (1999). Significance of paper properties on print quality in continuous ink jet printing. Journal of Imaging Science and Technology, 44(6), pp. 495-499.
8. Y. Tu, Q. Chen, L. Liao and W. Deng, "Research on Print Quality Assessment and Identification: Evaluation of Print Edge Roughness," 2009 International Conference on Computational Intelligence and Software Engineering, Wuhan, China, 2009, pp. 1-4, doi: 10.1109/CISE.2009.5364481.
9. Thung, K.-H. & Raveendran, P., 2010. A Survey of Image Quality Measures. In Proceedings of the 2009 International Conference for Technical Postgraduates (TECHPOS). IEEE. doi: 10.1109/TECHPOS.2009.5412098.
10. Y. Xu, N. Lin, Jiang Guiping and Guo Ge, "Research on a model for evaluating the perceived sharpness of black text in digital prints," 2011 4th International Congress on Image and Signal Processing, Shanghai, 2011, pp. 1954-1957, doi: 10.1109/CISP.2011.6100563.

11. H. Liu, M. Huang, B. Wu, Y. Liu, X. Li and Y. Xu, "Ink feeding control based on measured ink density," 2012 5th International Congress on Image and Signal Processing, Chongqing, China, 2012, pp. 1376-1380, doi: 10.1109/CISP.2012.6469888.
12. Zhang, D. and Tang, W. (2012). The new testing and standardized solutions of printing quality — PressSIGN intelligent printing production. pp.181–185. doi:https://doi.org/10.1109/icade.2012.6330123.
13. Baar, T., Brettel, H. and Ortiz, M.V. (2013). A survey of 3D image quality metrics for relief print evaluation. pp.1–6. doi:https://doi.org/10.1109/cvcs.2013.6626283.
14. Kumar, R. & Moyal, V., 2013. Visual Image Quality Assessment Technique Using FSIM. International Journal of Computer Applications Technology and Research, 2(3), pp.250-254. ISSN: 2319–8656.
15. W. -J. Zhang, F. Xu, X. -Y. Li, H. Xiao, S. -H. Peng and H. -D. Nam, "Automatic printing plate defect detection based on a simplified homocentric square filter," 2016 9th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Datong, China, 2016, pp. 336-340, doi: 10.1109/CISP-BMEI.2016.7852732.
16. Schirmer, J., Jewgeni Roudenko, Reichenberger, M., Neermann, S. and Franke, J. (2018). Print Quality Assessment by Image Processing Methods for Printed Electronics Applications. pp.1–6. doi:https://doi.org/10.1109/isse.2018.8443617.
17. Y. Wang and Y. Ai, "Research on the Influence of Digital Printing Quality," 2019 2nd World Conference on Mechanical Engineering and Intelligent Manufacturing (WCMEIM), Shanghai, China, 2019, pp. 392-395, doi: 10.1109/WCMEIM48965.2019.00084.
18. Wang, Y. and Cai, B. (2021). Research on Standardization of Printing Image Quality Control Based on G7 and GATF. 2021 IEEE 21st International Conference on Communication Technology (ICCT), pp.1202–1205. doi:https://doi.org/10.1109/icct52962.2021.9658093.
19. Angelov, S. and Milena Lazarova (2022). Convolutional Autoencoders for Image Comparison in Printing Industry Quality Control. pp.1–5. doi:https://doi.org/10.1109/comsci55378.2022.9912573.
20. S. Kumar and A. K. Baral, "Critical Analysis of Print Quality Factors (Hue Error and Print Contrast) in Piezoelectric Inkjet Press on Matt Coated and Gloss Coated Cellulosic Substrates," 2022 2nd International Conference on Advance Computing and

Innovative Technologies in Engineering (ICACITE), Greater Noida, India, 2022, pp. 1948-1952, doi: 10.1109/ICACITE53722.2022.9823474.

21. Wang, A., Wei, A.-T., Wang, H. and Chi, H. (2024). Enhancing 3D Printing Infill Quality through Advanced Machine Learning. 2022 14th International Conference on Electronics, Computers and Artificial Intelligence (ECAI), pp.1–4. doi:https://doi.org/10.1109/ecai61503.2024.10607535.
22. J. R. G. Townshend, C. O. Justice, C. Gurney and J. McManus, "The impact of misregistration on change detection," in IEEE Transactions on Geoscience and Remote Sensing, vol. 30, no. 5, pp. 1054-1060, Sept. 1992, doi: 10.1109/36.175340.
23. Xiaolong Dai and S. Khorram, "The effects of image misregistration on the accuracy of remotely sensed change detection," in IEEE Transactions on Geoscience and Remote Sensing, vol. 36, no. 5, pp. 1566-1577, Sept. 1998, doi: 10.1109/36.718860.
24. Mitropulos, P., Koulamas, C., Stojanovic, R., Koubias, S., Papadopoulos, G., & Karayanis, G. (1999). A real-time vision system for defect detection and neural classification of web textile fabric. Proceedings of SPIE - The International Society for Optical Engineering, [online] March. DOI: 10.1117/12.341126.
25. A. Kumar and H. C. Shen, "Texture inspection for defects using neural networks and support vector machines," Proceedings. International Conference on Image Processing, Rochester, NY, USA, 2002, pp. III-III, doi: 10.1109/ICIP.2002.1038978.
26. Wang, H. and Ellis, E.C. (2005). Image misregistration error in change measurements. Photogrammetric Engineering & Remote Sensing, 71(9), pp. 1037-1044. DOI: 10.14358/PERS.71.9.1037.
27. Oztan, B., Sharma, G. and Loce, R.P. (2004). Quantitative evaluation of misregistration-induced color shifts in color halftones. Proceedings of SPIE - The International Society for Optical Engineering, 5667, pp. 501-512. DOI: 10.1117/12.585072.
28. Van Niel, T.G., McVicar, T.R., Li, L.T., Gallant, J.C. and Yang, Q.K. (2008). The impact of misregistration on SRTM and DEM image differences. Remote Sensing of Environment, 112(5), pp. 2430-2442. DOI: 10.1016/j.rse.2007.11.003.
29. You Fucheng, Zhang Lifan and Zhang Yongbin, "The research of printing's image defect inspection based on machine vision," 2009 International Conference on Mechatronics and Automation, Changchun, China, 2009, pp. 2404-2408, doi: 10.1109/ICMA.2009.5246130.

30. X. Yong, W. Yaonan and Peng Tao, "Influences of color spaces on printing defect classification," 2009 Chinese Control and Decision Conference, Guilin, China, 2009, pp. 3354-3357, doi: 10.1109/CCDC.2009.5191931.
31. L. Liu, Y. Wang and Y. Wang, "Tie points-based pixel-level compensation of misregistration for change detection," 2010 IEEE International Geoscience and Remote Sensing Symposium, Honolulu, HI, USA, 2010, pp. 1015-1018, doi: 10.1109/IGARSS.2010.5653838.
32. S. H. Indera Putera and Z. Ibrahim, "Printed circuit board defect detection using mathematical morphology and MATLAB image processing tools," 2010 2nd International Conference on Education Technology and Computer, Shanghai, China, 2010, pp. V5-359-V5-363, doi: 10.1109/ICETC.2010.5530052.
33. Bouchot, J.-L., Stübl, G., and Moser, B. (2011). A template matching approach based on the discrepancy norm for defect detection on regularly textured surfaces. Proceedings of SPIE - The International Society for Optical Engineering, 8000. DOI: 10.1117/12.889865.
34. Y. Ding, T. Wang and X. Fu, "Reducing misregistration based on feature image mosaicing," 2012 IEEE International Conference on Information Science and Technology, Wuhan, China, 2012, pp. 535-538, doi: 10.1109/ICIST.2012.6221704.
35. Hsu, Q.-C. and Lin, C.-M. (2011). Image-based inspection system for detection of glass substrate's edge defects. Applied Mechanics and Materials, 121-126, pp. 3970-3974. DOI: 10.4028/www.scientific.net/AMM.121-126.3970
36. B. Kaur, G. Kaur and A. Kaur, "Detection and classification of Printed circuit board defects using image subtraction method," 2014 Recent Advances in Engineering and Computational Sciences (RAECS), Chandigarh, India, 2014, pp. 1-5, doi: 10.1109/RAECS.2014.6799537
37. Chauhan, S.K., Jangra, V. and Kumar, A. (2016). Comparative analysis of print defects in newspaper printing
38. S. Langkam and A. K. Deb, "A dual estimation approach for removing the show-through effect in the scanned documents," 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, 2017, pp. 1712-1716, doi: 10.1109/ICASSP.2017.7952449
39. M. Zhou, G. Wang, J. Wang, C. Hui and W. Yang, "Defect detection of printing images on cans based on SSIM and chromatism," 2017 3rd IEEE International Conference on

Computer and Communications (ICCC), Chengdu, China, 2017, pp. 2127-2131, doi: 10.1109/CompComm.2017.8322912.

40. Mei Zhang, Jinglan Wu, Huifeng Lin, Peng Yuan, Yanan Song, The Application of One-Class Classifier Based on CNN in Image Defect Detection, Procedia Computer Science,Volume 114, 2017, Pages 341-348, ISSN 1877-0509,https://doi.org/10.1016/j.procs.2017.09.040
41. Ghosh, B., Manas Kamal Bhuyan, Pradipta Sasmal, Yuji Iwahori and Gadde, P. (2018). Defect Classification of Printed Circuit Boards based on Transfer Learning. 2018 IEEE Applied Signal Processing Conference (ASPCON). doi:https://doi.org/10.1109/aspcon.2018.8748670.
42. Karthik, D., Vijayarekha, K., & Arun, A.R. (2018). Printing defect identification in pharmaceutical blisters using image processing. Asian Journal of Pharmaceutical and Clinical Research, 11(3), pp. 1-5. DOI: 10.22159/ajpcr.2018.v11i3.23407.
43. S. Langkam and A. K. Deb, "Multiple Model Kalman Filter Approach for Show-through Cancellation," 2018 IEEE 2nd International Workshop on Arabic and Derived Script Analysis and Recognition (ASAR), London, UK, 2018, pp. 78-83, doi: 10.1109/ASAR.2018.8480311.
44. Y. Wang and M. Yang, "Research on Online Monitoring of Print Image Quality Detection Based on Image Technology," 2021 IEEE 21st International Conference on Communication Technology (ICCT), Tianjin, China, 2021, pp. 1108-1111, doi: 10.1109/ICCT52962.2021.9657983.
45. J. Li, X. Bai, J. Pan, Q. Tian, W. Fu and Z. Jing, "A Deep Learning Method for Printing Defect Detection," 2022 IEEE 4th International Conference on Power, Intelligent Computing and Systems (ICPICS), Shenyang, China, 2022, pp. 246-249, doi: 10.1109/ICPICS55264.2022.9873703.
46. J. Li, J. Pan and Q. Zhang, "A Printing Defect Recognition Method Based on Class-imbalanced Learning," 2023 IEEE 3rd International Conference on Power, Electronics and Computer Applications (ICPECA), Shenyang, China, 2023, pp. 370-373, doi: 10.1109/ICPECA56706.2023.10075748.
47. J. Liu et al., "An Improved Printing Defect Detection Method Based on YOLOv5s," 2023 5th International Conference on Communications, Information System and Computer Engineering (CISCE), Guangzhou, China, 2023, pp. 253-257, doi: 10.1109/CISCE58541.2023.10142568.

48. W. Qi, J. Liu, Y. Li and J. Liu, "Research on Defect Detection in Printing Barcode Area Based on Machine Learning," 2024 20th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), Guangzhou, China, 2024 pp. 1-7, doi: 10.1109/ICNC-FSKD64080.2024.10702241.
49. Lee, S.U., Chung, S.Y. and Park, R.H. (1990). A comparative performance study of several global thresholding techniques for segmentation. Computer Vision, Graphics, and Image Processing, 52(2), pp. 171-190. DOI: 10.1016/0734-189X(90)90053-X.
50. Beauchemin, M. and Fung, K.B. (2005). An adaptive filter for the reduction of artifacts caused by image misregistration. Proceedings of the International Workshop on the Analysis of Multi-Temporal Remote Sensing Images, June. DOI: 10.1109/AMTRSI.2005.1469865.
51. Du, W., Tian, X. and Sun, Y. (2011). A dynamic threshold edge-preserving smoothing segmentation algorithm for anterior chamber OCT images based on modified histogram. Proceedings of the 4th International Congress on Image and Signal Processing (CISP), DOI: 10.1109/CISP.2011.6100288.
52. T. Kiran and Goo-Rak Kwon, "An advanced segmentation using bit-plane slicing technique in extraction of lungs region," 2011 Second Asian Himalayas International Conference on Internet (AH-ICI), Kathmundu, Nepal, 2011, pp. 1-5, doi: 10.1109/AHICI.2011.6113949
53. Manish I. Patel, Vishvjit K. Thakar, Shishir K. Shah,Image Registration of Satellite Images with Varying Illumination Level Using HOG Descriptor Based SURF,Procedia Computer Science,Volume 93,2016,Pages 382-388, ISSN 1877-0509,https://doi.org/10.1016/j.procs.2016.07.224.
54. Boda, R., Yezerla, S.K. and Naik, B.R. (2016) 'Performance analysis of image segmentation methods for noisy MRI images', 2016 International Conference on Communication and Signal Processing (ICCSP), April 2016. doi:10.1109/ICCSP.2016.7754286.
55. Cheng, Y. and Li, B. (2021) 'Image segmentation technology and its application in digital image processing', 2021 IEEE Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC), April 2021. doi:10.1109/IPEC51340.2021.9421206.
56. Jianqing, H., Qi, Y., Debing, L. and Hailing, F. (2022) 'An image segmentation method for banana leaf disease image with complex background', 2022 5th International

Conference on Data Science and Information Technology (DSIT), July 2022. doi:10.1109/DSIT55514.2022.9943850.

57. A. Sharma, N. Mishra and P. Sharma, "Image Retrieval Algorithm Based on Quantized DCT Coefficients," 2011 International Conference on Computational Intelligence and Communication Networks, Gwalior, India, 2011, pp. 431-434, doi: 10.1109/CICN.2011.90.
58. Frid, N., Mlinaric, H., & Knezovic, J. (2013) 'Acceleration of DCT transformation in JPEG image conversion', 36th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO 2013), May 2013.
59. Abu, N.A. and Ernawan, F. (2015) 'A novel psychovisual threshold on large DCT for image compression', The Scientific World Journal, 2015(3), pp. 001–011. doi:10.1155/2015/821497.
60. Fu, X. and Wan, Y. (2015) 'Accurate image rotation using DCT transformation', 2015 IEEE Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), December 2015. doi:10.1109/IAEAC.2015.7428611.
61. Chao-Yang Pang, Ri-Gui Zhou, Ben-Qiong Hu, WenWen Hu, Ahmed El-Rafei, Signal and image compression using quantum discrete cosine transform,Information Sciences,Volume 473,2019,Pages 121-141,ISSN 0020-0255,https://doi.org/10.1016/j.ins.2018.08.067.
62. S. Muniyappan, A. Allirani and S. Saraswathi, "A novel approach for image enhancement by using contrast limited adaptive histogram equalization method," 2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT), Tiruchengode, India, 2013, pp. 1-6, doi: 10.1109/ICCCNT.2013.6726470.
63. Kong, N.S.P., Ibrahim, H. and Hoo, S.C. (2013). A literature review on histogram equalization and its variations for digital image enhancement. International Journal of Innovation, Management and Technology, 4(4)
64. K. Santhi and R. S. D. WahidaBanu, "Contrast enhancement of bi-histogram equalization with neighborhood metrics," 2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT), Tiruchengode, India, 2013, pp. 1-5, doi: 10.1109/ICCCNT.2013.6726796.
65. Patel, O., Maravi, Y.P.S. and Sharma, S. (2013). A comparative study of histogram equalization-based image enhancement techniques for brightness preservation and

contrast enhancement. Signal & Image Processing: An International Journal, 4(5), pp. 15-25. DOI: 10.5121/sipij.2013.4502.

66. N. Senthilkumaran and J. Thimmiaraja, "Histogram Equalization for Image Enhancement Using MRI Brain Images," 2014 World Congress on Computing and Communication Technologies, Trichirappalli, India, 2014, pp. 80-83, doi: 10.1109/WCCCT.2014.45.
67. Mustafa, W.A. and Abdul Kader, M.M. (2018). A review of histogram equalization techniques in image enhancement application. Journal of Physics Conference Series, 1019(1), p. 012026. DOI: 10.1088/1742-6596/1019/1/012026.
68. Zhou Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," in IEEE Transactions on Image Processing, vol. 13, no. 4, pp. 600-612, April 2004, doi:10.1109/TIP.2003.819861.
69. A. Horé and D. Ziou, "Image Quality Metrics: PSNR vs. SSIM," 2010 20th International Conference on Pattern Recognition, Istanbul, Turkey, 2010, pp. 2366-2369, doi: 10.1109/ICPR.2010.579.
70. Dauphin, G. and Viaris de Lesegno, P. (2010). Analysis and comparison of quality metrics with reference based on uniform colour spaces. Visual Information Processing (EUVIP), 2010. DOI: 10.1109/EUVIP.2010.5699122.
71. Sara, U., Akter, M. & Uddin, M.S., 2019. Image Quality Assessment through FSIM, SSIM, MSE and PSNR—A Comparative Study. Journal of Computer and Communications, 7(3), pp.8-18. doi: 10.4236/jcc.2019.73002.
72. Ahmed, Z., Sayadi, M. & Faniech, F., 2019. Satellite Images Features Extraction Using Phase Congruency Model. IJCSNS International Journal of Computer Science and Network Security, 9(2). Manuscript received February 5, 2009; Manuscript revised February 20, 2019.
73. P. Anandababu and M. Kamarasan, "Structural Similarity Measurement with Metaheuristic Algorithm for Content based Image Retrieval," 2019 International Conference on Smart Systems and Inventive Technology (ICSSIT), Tirunelveli, India, 2019, pp. 73-77, doi: 10.1109/ICSSIT46314.2019.8987784.
74. B. Xia, J. Cao and C. Wang, "SSIM-NET: Real-Time PCB Defect Detection Based on SSIM and MobileNet-V3," 2019 2nd World Conference on Mechanical Engineering and Intelligent Manufacturing (WCMEIM), Shanghai, China, 2019, pp. 756-759, doi: 10.1109/WCMEIM48965.2019.00159.

75. M. D. Malkauthekar, "Analysis of euclidean distance and Manhattan Distance measure in face recognition," Third International Conference on Computational Intelligence and Information Technology (CIIT 2013), Mumbai, 2013, pp. 503-507, doi: 10.1049/cp.2013.2636.
76. Peng, C.-Y.J., 2002. An Introduction to Logistic Regression Analysis and Reporting. The Journal of Educational Research, 96(1), pp.3-14. Taylor & Francis. doi: 10.1080/00220670209598786.
77. Maalouf, M. 2011. Logistic Regression in Data Analysis: An Overview. International Journal of Data Analysis Techniques and Strategies, 3(3), pp.281-299. doi: 10.1504/IJDATS.2011.041335.
78. Xiaoling Xia, Cui Xu and Bing Nan (2017). Inception-v3 for flower classification. 2017 2nd International Conference on Image, Vision and Computing (ICIVC). doi:https://doi.org/10.1109/icivc.2017.7984661.
79. Mittal, S., Kaushik, P., Hashmi, S. and Kumar, K. (2018). Robust Real Time Breaking of Image CAPTCHAs Using Inception v3 Model. pp.1–5. doi:https://doi.org/10.1109/ic3.2018.8530607.
80. Chen, G., Chen, Y., Yuan, Z. and Lu, X. (2019) 'Breast cancer image classification based on CNN and bit-plane slicing', 2019 International Conference on Medical Imaging Physics and Engineering (ICMIPE), November 2019. doi:10.1109/ICMIPE47306.2019.9098216.
81. Li, G., Lin, X., Zhang, J., Yan, J., Xie, B. and Qin, J. (2019). Fresh Tea Leaves Classification Using Inception-V3. 2019 IEEE 2nd International Conference on Information Communication and Signal Processing (ICICSP). doi:https://doi.org/10.1109/icicsp48821.2019.8958529.
82. Hsieh, Y.-C., Chin, C.-L., Wei, C.-S., Chen, I-Miao., Yeh, P.-Y. and Tseng, R.-J. (2020). Combining VGG16, Mask R-CNN and Inception V3 to identify the benign and malignant of breast microcalcification clusters. pp.1–4. doi:https://doi.org/10.1109/ifuzzy50310.2020.9297809..
83. J. Bae, M. Kim and J. S. Lim, "Feature Extraction Model Based on Inception V3 to Distinguish Normal Heart Sound from Systolic Murmur," 2020 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Korea (South), 2020, pp. 460-463, doi: 10.1109/ICTC49870.2020.9289317.
84. Z. Yuan, G. Chen, Q. Chen, W. Li and S. Long, "Breast Cancer Image Classification Based on CNN Classifier," 2020 International Conferences on Internet of Things

(iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData) and IEEE Congress on Cybermatics (Cybermatics), Rhodes, Greece, 2020, pp. 921-925, doi: 10.1109/iThings-GreenCom-CPSCom-SmartData-Cybermatics50389.2020.00143.

85. P. Yin, R. Yuan, Y. Cheng and Q. Wu, "Deep Guidance Network for Biomedical Image Segmentation," in IEEE Access, vol. 8, pp. 116106-116116, 2020, doi: 10.1109/ACCESS.2020.3002835.
86. Nabil, M., Rady, M., Moussa, K., Mahmoud Wessam, Hossam, M., Retaj Yousri and M. Saeed Darweesh (2021). A Preprocessing Approach to Improve the Performance of Inception v3-based Face Shape Classification. doi:https://doi.org/10.1109/niles53778.2021.9600535.
87. Saeed, Z., Khan, M.U., Raza, A., Khan, H., Javed, J. and Arshad, A. (2021). Classification of Pulmonary Viruses X-ray and Detection of COVID-19 Based on Invariant of Inception-V 3 Deep Learning Model. pp.1–6. doi:https://doi.org/10.1109/icecube53880.2021.9628338.
88. R. Thangaraj, P. Pandiyan, T. Pavithra, V. K. Manavalasundaram, R. Sivaramakrishnan and V. K. Kaliappan, "Deep Learning based Real-Time Face Detection and Gender Classification using OpenCV and Inception v3," 2021 International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation (ICAECA), Coimbatore, India, 2021, pp. 1-5, doi: 10.1109/ICAECA52838.2021.9675635
89. M. Balipa and A. Castalino, "Alstonia Tree Detection using CNN and Inception V3 Algorithms," 2022 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), Greater Noida, India, 2022, pp. 318-321, doi: 10.1109/ICCCIS56430.2022.10037697.
90. Likhitha, S. & Baskar, R., 2022. Skin Cancer Segmentation Using R-CNN Comparing with Inception V3 for Better Accuracy. In Proceedings of the 2022 11th International Conference on System Modeling & Advancement in Research Trends (SMART). IEEE. doi: 10.1109/SMART55829.2022.10047686.
91. G. Priyadharshini and D. R. Judie Dolly, "Comparative Investigations on Tomato Leaf Disease Detection and Classification Using CNN, R-CNN, Fast R-CNN and Faster R-CNN," 2023 9th International Conference on Advanced Computing and

Communication Systems (ICACCS), Coimbatore, India, 2023, pp. 1540-1545, doi: 10.1109/ICACCS57279.2023.10112860.

92. M. Mohammed Iqbal, G. Tamilmani, C. H. Tej Pavan and P. Sri Pranav, "COVID-19 Classification using CNN, Inception V3 and Transfer Learning," 2023 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES), Chennai, India, 2023, pp. 1-6, doi: 10.1109/ICSES60034.2023.10465433.
93. N. Joshi, D. Kumar, V. Kukreja and P. Sarangi, "Automated Tea Leaves Recognition: Multi-Classification Using YOLOv5 and Inception V3 Model," 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), Delhi, India, 2023, pp. 1-6, doi: 10.1109/ICCCNT56998.2023.10306542.
94. T Kalpana, R Thamilselvan, Saravanan, T.M., M Pyingkodi, S Nandhakumar, V Priyanka and M Sowndharya (2023). An Image Based Classification and Prediction of Diseases on Cotton Leaves Using Deep Learning Techniques. doi:https://doi.org/10.1109/iccci56745.2023.10128306.
95. R. Chavan and D. Pete, "Classification of Retinal Fundus Images using VGG16 and Inception V3," 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), Delhi, India, 2023, pp. 1-5, doi: 10.1109/ICCCNT56998.2023.10306593
96. Susanti, E., Cahyo, E.N., Sutanta, E., Ariyana, R.Y. & Kumalasanti, R.A., 2023. Beef Image Classification Using the Inception V3 Transfer Learning Model. In Proceedings of the 2023 IEEE 9th Information Technology International Seminar (ITIS). IEEE. doi: 10.1109/ITIS59651.2023.10420013.
97. Verma, S., Singh, G., None Warishpatel, Salman, H.M., A. Pavani and Srikanth, S. (2023). Detection of Traffic Sign using Inception V3 in Comparison with VGG-19 to Measure Accuracy. pp.730–733. doi:https://doi.org/10.1109/icacite57410.2023.10183243.
98. Shruti Kankariya, Kanak Thakre, Solanki, U., Mali, S. and Abhishek Chunawale (2024). Sign Language Gestures Recognition using CNN and Inception v3. 2021 International Conference on Emerging Smart Computing and Informatics (ESCI). doi:https://doi.org/10.1109/esci59607.2024.10497401.

99. Kaur, A., Sharma, R., Chattopadhyay, S. and Verma, A. (2024). Automated Multiclass Classification of Groundnut Leaf Diseases Using Fine-Tuned InceptionV3 Model. pp.1–4. doi:https://doi.org/10.1109/wconf61366.2024.10692187.

100. Rani, K.S., Thamizhazhakan K, Babu, S.S., Barath Kumar S, Ebi Kumar A and Harish D (2024). Implementation of Inception V3-Based CNN Classification for Malware Detection. 22, pp.1–6. doi:https://doi.org/10.1109/raeeucci61380.2024.10547935.

101. Soelinto et al**.** Pengendalian Scumming pada Mesin Lithrone 440 di PT. Kartika Naya (Studi Kasus Kemasan Pigeon Peristaltic Nipple L), Jurnal Magenta, STMK Trisakti - Vol. 3 | No. 02 | Juli 2019

102. Cohen, M., Afzali, A., Simonyi, E., & Pennington, K., 1991. Making negatives and plates for printing by electroerosion: III. Use of the direct negative and direct plate. IBM J. Res. Dev., 35, pp. 512-534. https://doi.org/10.1147/rd.354.0512.

103. Gonzalez, R.C. and Woods, R.E. “Digital image processing”. 2018,New York, Ny: Pearson

104. Pang, C.-Y., Zhou, R.-G., Hu, B.-Q., Hu, W. and El-Rafei, A. (2019). “Signal and image compression using quantum discrete cosine transform. Information Sciences,” 473, pp.121–141. doi:10.1016/j.ins.2018.08.067

105. Saha, J., Naskar, S., Chatterjee, A. and Paul, K.C. (2018). “Print Scum Identification using DCT based Computer Vision Method.” 2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN). doi:10.1109/icrcicn.2018.8718734

106. L. A. Wilson, What the printer should know about Paper, 3rd ed.

107. Todd , R.E. (n.d.). Printing Inks formulation principals, manufacture and quality control testing procedures. ISBN: 85802 027 1

108. J.P.Casey, Pulp and paper Chemistry and Chemical Technology, 3rd ed, Vol-IV

109. Saha, J., Naskar, S. and Maiti, S. (2023). Showthrough and Strikethrough print defect detection using histogram equalization based computer vision method. Journal of Graphic Engineering and Design, 14(2), pp.15–21. doi:https://doi.org/10.24867/jged-2023-2-015.

110. Green, P. and MacDonald, L. (2011). Colour Engineering. John Wiley & Sons.

111. Al bovik, The essential guide to Image Processing, 2nd ed.

112. Macphee , J.M. (1998.). fundamental of lithographic printing:mechanics of printing

113. Brehm, P.V. (1996). Introduction to Color Bars. Graphic Communications Assn

114. Saha, J., Naskar, S. (2022). An approach toward detection of doubling print defect using SSIM algorithm, Proceedings of International Conference on Data, Electronics and Computing, Algorithms for Intelligent Systems, 2022. DOI: 10.1007/978-99-1509-5_15.
115. Berns, R. S. (2000). Principles of Color Technology.
116. Shapiro, C. (1974). The Lithographers Manual.
117. Saha*, et al.* (2023). An Approach towards Measurement of Color Shifting in Misregistration Print Defect using Euclidean and Manhattan Distance Metrics. International Journal on Recent and Innovation Trends in Computing and Communication, 11(9), pp.3676–3680. doi: https://doi.org/10.17762/ijritcc.v11i9.9590.
118. P. Green,Color Management understanding and using ICC Profiles, 1st edition, 2010.
119. Giorgianni, E.J. and Madden, T.E. (2008). Digital color management: encoding solutions. Chichester, U.K.: J. Wiley.
120. B. Chakravarty, Digital Color Printing Technology, 1st edition 2002.
121. Breede, M.H. (2006). Handbook of Graphic Arts Equations. Gatfpress
122. MacDonald, L. and Luo, M.R. (2002). Colour Image Science. John Wiley & Sons.
123. Komorowski, M., Marshall, D.C., Salciccioli, J.D. and Crutain, Y. (2016). Exploratory Data Analysis. Secondary Analysis of Electronic Health Records, [online] pp.185–203. doi:https://doi.org/10.1007/978-3-319-43742-2_15.
124. Rao, A.S., Vardhan, B.V. and Shaik, H. (2021). Role of Exploratory Data Analysis in Data Science. [online] IEEE Xplore. doi:https://doi.org/10.1109/ICCES51350.2021.9488986
125. Joshi, N., Kumar, D., Kukreja, V. and Pradeepta Sarangi (2023). Automated Tea Leaves Recognition: Multi-Classification Using YOLOv5 and Inception V3 Model. doi:https://doi.org/10.1109/icccnt56998.2023.10306542
126. PrintPlanet.com. (2019). X-Rite (Gretag) SpectroEye. [online] Available at: https://printplanet.com/threads/x-rite-gretag-spectroeye.277696/ [Accessed 28 Dec. 2024]
127. Shorten, C., Khoshgoftaar, T.M. A survey on Image Data Augmentation for Deep Learning. J Big Data 6, 60 (2019). https://doi.org/10.1186/s40537-019-0197-0
128. Luke T, Geoff N. Improving deep learning using generic data augmentation. arXiv preprint. 2017

129. Fawzi, A., Fawzi, O. and Frossard, P. (2018). Analysis of Data Augmentation in Machine Learning Frameworks. New York: Springer.
130. Xu, M., Yoon, S., Fuentes, A. and Park, D.S. (2023). A Comprehensive Survey of Image Augmentation Techniques for Deep Learning. Pattern Recognition, 137, p.109347. doi:https://doi.org/10.1016/j.patcog.2023.109347.
131. Dadhich, A. (2018). Practical computer vision.
132. Aboul Ella Hassanien (2024). Artificial Intelligence for Environmental Sustainability and Green Initiatives. Springer Nature.
133. Tukey, J.W. (1977) Exploratory Data Analysis. Reading, MA: Addison-Wesley
134. Dudáš, A. Graphical representation of data prediction potential: correlation graphs and correlation chains. Vis Comput 40, 6969–6982 (2024). https://doi.org/10.1007/s00371-023-03240-y
135. Richardson, A. (2011). Logistic Regression: A Self-Learning Text, Third Edition by David G. Kleinbaum, Mitchel Klein. International Statistical Review, 79(2), pp.296–296. doi:https://doi.org/10.1111/j.1751-5823.2011.00149_22.x
136. Bishop, C.M. (2006) Pattern Recognition and Machine Learning. New York: Springer
137. Kleinbaum, D.G., Klein, M. (2010) Logistic Regression: A Self-Learning Text. 3rd edn. New York: Springer ISBN : 978-1-4419-1741-6
138. Nobel, S.M.N.; Sultana, S.; Singha, S.P.; Chaki, S.; Mahi, M.J.N.; Jan, T.; Barros, A.; Whaiduzzaman, M.Unmasking Banking Fraud:Unleashing the Power of Machine Learning and Explainable AI (XAI) on Imbalanced Data. Information 2024, 15,298. https://doi.org/10.3390/
139. Cha Zhang and Ma, Y. (2012). Ensemble Machine Learning. New York: Springer.
140. Genuer, R. and Jean-Michel Poggi (2020). Random forests with R. Cham, Switzerland: Springer.
141. Smith, C. (2017). Decision trees and random forests: a visual introduction for beginners. Canada Blue Windmill Media.
142. roadmaps.mit.edu. Random Forest in Data Analytics - MIT Technology Roadmapping. Available at: https://roadmaps.mit.edu/index.php/Random_Forest_in_Data_Analytics
143. Awad, M. and Khanna, R. (2015). Efficient Learning Machines : Theories, Concepts, and Applications for Engineers and System Designers. Berkeley, Ca: Apress.
144. Ming-Yang Kao (2008). Encyclopedia of algorithms. Berlin: Springer.

145. Hastie, T., Tibshirani, R. and Friedman, J. (2009) The Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd ed. New York: Springer

146. Santra AK, Christy CJ (2012) Genetic algorithm and confusion matrixfor document clustering. Int J Comput Sci 9(1):322–328

147. Nagelkerke, N. J. D. (2005). Coefficient of Determination. In Encyclopedia of Biostatistics. John Wiley & Sons

148. Chai, T. and Draxler, R.R., 2014. "Root mean square error (RMSE) or mean absolute error (MAE)? – Arguments against avoiding RMSE in the literature." Geoscientific Model Development, 7(3), pp.1247–1250

Jayeeta Saha
16/09/25