I.J. Image, Graphics and Signal Processing, **2026, 1, 128-149**
Published Online on February 8, 2026 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijigsp.2026.01.08

A Novel Hybrid Approach Using MRMR-based Feature Selection and Bayesian Optimized Random Forest Classification for Accurate Fabric Defect Detection

Ritu Juneja*
Department of Electronics & Communication Engineering, Baba MastnathUniversity, Rohtak, Haryana, India
E-mail: rtjuneja@gmail.com
ORCID iD: https://orcid.org/0009-0009-7770-2784
*Corresponding Author

Anil Dudy
Department of Electronics & Communication Engineering, Baba MastnathUniversity, Rohtak, Haryana, India
E-mail: anildudy10@gmail.com
ORCID iD: https://orcid.org/0009-0008-5483-1199

Received: 20 July, 2025; Revised: 12 October, 2025; Accepted: 14 November, 2025; Published: 08 February, 2026

Abstract: The textile industry holds a central position in India's economy, contributing substantially to both employment and GDP. Despite technological advancements, maintaining stringent quality standards remains a persistent challenge due to defects such as cracks, stains, and inconsistencies in fabrics. Traditional manual inspection methods, while effective to a degree, are labor-intensive, time-consuming, and prone to human error. This paper proposes an innovative approach to address these challenges through the application of machine learning and computer vision techniques in fabric defect detection. Specifically, the research focuses on integrating advanced texture feature extraction methods—Gray-Level Co-occurrence Matrix (GLCM), Local Binary Patterns (LBP), and Histogram of Oriented Gradients (HOG)—with a robust classification framework using Bayesian optimized Random Forest. The methodology emphasizes efficient feature selection via Minimum Redundancy Maximum Relevance (MRMR), enhancing the system's accuracy and efficiency. By leveraging a comprehensive dataset from Kaggle encompassing diverse fabric defects, the proposed system aims to significantly improve defect detection accuracy, reduce manual intervention, and ensure consistent product quality across textile manufacturing processes. The highest accuracy achieved in the evaluation is 99.52%.

Index Terms: Bayesian Optimization, Gray-Level Co-occurrence Matrix, Histogram of Oriented Gradients, Local Binary Patterns, Minimum Redundancy Maximum Relevance, Random Forest Classifier

1. Introduction

The textile industry is a cornerstone of India's economy, contributing significantly to both employment and gross domestic product. With the country's historical and cultural ties to textile production, the industry has evolved into one of the largest in the world, serving both domestic and international markets. Despite advancements in technology, maintaining high standards of quality remains a crucial challenge. Defects such as cracks, stains, and inconsistencies in fabric can significantly impact the usability and aesthetic appeal of textile products, leading to substantial economic losses and reduced customer satisfaction.

Quality control in the textile industry is vital for ensuring that products meet the high standards expected by consumers. Traditional methods of defect detection in fabrics have predominantly relied on manual inspection, which, despite being effective to a certain extent, is labour-intensive, time-consuming, and prone to human error. The increasing complexity of fabrics, with more intricate patterns and diverse textures, further complicates the inspection process. Consequently, there is a pressing need for automated, accurate, and efficient methods for fabric defect detection to enhance quality control and minimize resource wastage [1].

In recent years, the integration of machine learning and computer vision techniques has shown great promise in addressing the challenges of defect detection in textiles [2]. This research paper proposes a comprehensive approach to textile fabric defect detection using a combination of advanced texture feature extraction methods and a robust classification framework. The methodology leverages a dataset of fabric images from Kaggle, which provides a diverse and representative sample of various types of defects commonly encountered in the textile industry.

The proposed approach begins with the extraction of texture features using three well-established methods: Gray-Level Co-occurrence Matrix (GLCM), Local Binary Patterns (LBP), and Histogram of Oriented Gradients (HOG). GLCM captures the spatial relationship between pixel intensities, providing information about the texture's roughness, smoothness, and regularity. LBP is a powerful method for texture classification that encodes the local texture patterns, making it highly effective for identifying microstructures in the fabric. HOG, on the other hand, captures the distribution of gradient orientations, which is particularly useful for detecting edges and shapes in the fabric patterns.

Once the texture features are extracted, the next step involves the selection of the most relevant features to improve the efficiency and accuracy of the classification model. This is achieved using the Minimum Redundancy Maximum Relevance (MRMR) algorithm, which selects features that are highly relevant to the target class while minimizing redundancy among them. MRMR ensures that the feature set fed into the classifier is both informative and non-redundant, thereby enhancing the model's performance.

The selected features are then classified using a Bayesian optimized Random Forest classifier. Random Forest is an ensemble learning method that constructs multiple decision trees during training and outputs the class that is the mode of the classes of the individual trees. Bayesian optimization is employed to fine-tune the hyperparameters of the Random Forest classifier, ensuring that the model achieves optimal performance.

This research aims to demonstrate that the combination of advanced texture feature extraction methods, effective feature selection, and a robust classification framework can significantly improve the accuracy and efficiency of fabric defect detection. The expected outcome is a high-performance automated defect detection system that can be seamlessly integrated into textile production lines, thereby reducing reliance on manual inspection, minimizing defects, and ensuring consistent quality of textile products.

The contributions of this research are multifaceted, addressing both methodological advancements and practical applications in the field of textile fabric defect detection. The primary contributions are outlined as follows:

- ***Comprehensive Texture Feature Extraction:*** The study introduces a novel combination of Gray-Level Co-occurrence Matrix (GLCM), Local Binary Patterns (LBP), and Histogram of Oriented Gradients (HOG) for texture feature extraction. This comprehensive feature set effectively captures the intricate spatial and structural characteristics of fabric images, enhancing the robustness and discriminative power of the feature representation. By leveraging these diverse methods, the proposed approach can identify and distinguish a wide range of fabric defects with high accuracy.
- ***Efficient Feature Selection through MRMR:*** The application of Minimum Redundancy Maximum Relevance (MRMR) for feature selection significantly refines the feature set. This optimization process reduces dimensionality, alleviating the curse of dimensionality, and enhances classifier performance by retaining the most informative features. By focusing on features that contribute the most to defect detection while minimizing redundancy, the MRMR-based selection ensures that the model remains both efficient and effective.
- ***Advanced Classification with Bayesian Optimized Random Forest:*** The utilization of a Bayesian optimized Random Forest classifier introduces a robust classification framework. Random Forest, an ensemble learning method, is known for its high accuracy and resilience to overfitting. Bayesian optimization further fine-tunes the hyperparameters of the Random Forest, ensuring optimal performance. This dual approach not only improves classification accuracy but also provides a scalable solution adaptable to various defect types and fabric conditions.

This research addresses critical needs in the textile industry by proposing an innovative approach to fabric defect detection. The integration of state-of-the-art machine learning and computer vision techniques ensures that the system can handle the complexity and variability of modern textile products. The expected outcomes include a high-performance automated defect detection system that enhances product quality and consistency, ultimately benefiting both manufacturers and consumers.

The paper commences with an extensive literature review (Section 2) that synthesizes current research on textile fabric defect detection methodologies, focusing on feature extraction and classification techniques. Section 3 presents the material and methods used in this research paper. Section 4 outlines the proposed methodology, integrating all the features extraction methods followed by the feature selection using MRMR. This selected feature set is evaluated using Bayesian-optimized random forest classifier to classify defects based on the reduced feature vector. Section 5 presents results from MATLAB simulations, detailing classification performance metrics like accuracy, precision, recall, and F1-score. Section 6 concludes by summarizing findings, discussing implications, and suggesting future research directions in textile defect detection.

2. Literature Review

With the improvement of people's living standards, the quality demands of living products are also increasing. However, defects such as cracks and stains have a major negative impact on people's experience of using the products. For this reason, quality control is a very important task in sectors such as the textile industry, construction industry, and electronic production industry. In addition, clothing is a strict demand of people; its role is not only to keep warm but also to complete the personal image and create a comfortable tactile environment. Therefore, rigorous quality testing procedures are essential in the textile industry [3].

Surface defects of the fabric are the main factor causing fabric quality problems. Therefore, detection of surface defects of fabrics is the critical point of fabric quality inspection. To some extent, the scale of fabric defects is affected by the quality of the yarn and the loom [4]. Most causes of errors are thread problems during weaving, such as inconsistencies in color or width, hairiness, bamboo knots, and split ends. It can also be caused by improper machine parameters, machine damage, excessive stress, and other machine defects. More than 70 fabric defects have been identified by the textile industry [5]. Most occur in the direction or vertical direction of fabric movement during weaving. According to quality standards, defects on the fabric surface can be divided into two categories: uneven color of the surface and local texture irregularities [6]. Common fabric defects in weaving include separation of seams, warp breakage, warp sticking, dough spots, floating threads, broken holes, rolled weft, stains, etc.

According to a related study, fabric defects reduce the overall price by 45-65%. These serious defects not only lead to reduced revenue but sometimes make the fabric unsellable. Additionally, reputation impact and unnecessary after-sales actions are possible on the merchant selling the product. Addressing these problems, the automatic system for detecting fabric defects can improve product quality, save weaving costs, and reduce resource waste [7].

The clothing industry has a huge production volume, so the need for fabric defect detection systems for textile production is equally high. Especially when high-quality fabrics are woven using high-quality raw materials, the requirements for fabric defect detection systems will be more stringent. Textile weaving testing is generally divided into two stages. The first stage is the study of process parameters during the weaving process. As a preventive control, this type of control is a monitoring behavior that constantly monitors the weaving process and parameters for errors and changes. However, since there are too many influencing factors and variables in the weaving process, and small changes can lead to different categories of fabric defects, the weaving process is usually only parameter inspection, but does not detect whether there are defects. The second stage is product inspection. It detects whether there is a defect in the fabric after weaving. Existing research has mainly focused on the product review phase [8,9]. Until now, manual visual inspection has been the main method of fabric defect detection in the textile industry and plays a major role in fabric quality control. However, the method suffers from high labor intensity of workers, harsh working environment near the loom, low detection efficiency and high false detection rate [10]. Therefore, the high cost and low effectiveness of manual visual inspection methods always bother people [11].

The area of computer vision has already solved several object identification problems using segmentation and pattern recognition techniques. This area of computing has helped to customize and automate certain tasks, generating information automatically, as it manipulates digital images (from acquisition to recognition of an object of interest) in various processes. Such systems seek to develop different methods capable of manipulating images with different characteristics, being efficient, robust and reliable. In recent years, due to the increase in research in the field of computer vision and the rapid development of technology, many innovative topics have emerged to conduct research using computer vision technology [12]. Therefore, automatic fabric defect detection methods based on computer vision technology have attracted much attention.

In this sense, techniques for segmenting and extracting attributes of objects of interest in images can offer experts an alternative computational tool for visual analysis and automatic object recognition [13,14].

Digital image processing techniques have been increasingly applied to the analysis of texture samples in recent years. It is worth mentioning here some works that have applied this area of knowledge, whether for classifying textile fibers, classifying flat fabrics, detecting defects or inspection.

The authors of [15] presented a new biologically inspired method to invariably recognize the fabric weave pattern (fabric texture) and yarn color from the color input image. The authors propose a model in which the fabric pattern descriptor is based on the model for computer vision inspired by the hierarchy in the visual cortex, the color descriptor is based on the eye color channel inspired by the classical eye color theory of human vision, and the classification stage is composed of an extreme (multilayer) learning machine. Like the weft pattern descriptor, the yarn color descriptor and the classification stage are all biologically inspired, proposing a method that is completely biologically inspired. The classification performance of the proposed algorithm indicates that the vision models assist the computer for accurate, fast, reliable and cost-effective solution for industrial automation.

The authors of [16] proposed a CNN-based textile fiber classification method adapted to the classification of seven different types of fiber shapes using grayscale fiber images. The results indicate that the proposed approach achieves higher accuracy than layer backpropagation (PA) neural network and support vector machine (SVM) methods in fiber recognition.

In some countries, their main source of income is obtained through the textile industry, requiring the production of less defective textiles to minimize costs and production time, some works seek to detect defects in fabrics [17]. Considering this lack, the authors implemented a computer-based textile defect recognizer. The vision methodology with the combination of multilayer neural networks to identify four types of textile defects. The authors of [18] presented a new approach for classifying mesh defects using the radial basis function (RBF) improved by the Gaussian mixture model. The algorithm proposed by the authors of [17] is experimented on tissue defect images with nine classes and achieves superior performance, which proves its usefulness in practice.

Other works that also seek to identify defects or perform automated tissue inspection are by the authors of [19]. The articles propose new methods based on segmentation and lattice models that automatically identify defects in tissue images. A challenging task due to the unpredictable visual shapes of the defect fabric and its scarcity compared to the huge amount of defect-free fabric products.

The authors of [20] proposed the use of the Discrete Wavelet Transform to extract features from tissue images, with the aim of detecting visual defects. The images come from a database created by the authors themselves, made publicly available online. Four classification algorithms (Support Vector Machine, k-Nearest Neighbor), Decision Tree and Naive Bayes were compared in order to find the one that would bring the best result in relation to hit rate and processing time. The results obtained using MATLAB showed that the k-Nearest Neighbor classifier brought the best performance in relation to hit rate and the Decision Tree in relation to processing time, weft pattern descriptor, thread color descriptor and stage classification systems are all biologically inspired. The classification performance of the proposed algorithm indicates that the vision models assist the computer for accurate, fast, reliable and cost-effective solution for industrial automation.

Analysis using CNN is also applied by the authors of [21], a new method to analyze the texture information in multi-window mesh image to enhance the defect feature is introduced. The defect feature information is segmented by the convolutional neural network, and three variable terms are defined to represent the feature. Using Interlocking fabric with hole defects, course mark, dropped stitch and fly as experiment materials, it proved that the acquired information involved adequate defect information with less noise effect and the classification result by artificial neural network was well carried out.

The authors of [22,23,24] develop a system for recognizing fabric texture patterns to recover images based on their characteristics, that is, from an image repository, make querying images of interest using other images as a basis. The objective of the algorithm was, taking an image that represents the taffeta, twill or satin pattern, to return the largest number of images with identical patterns.

Research Gap: The literature suggests that despite the application of different methods such as CNNs, SVM, and k-NN in detecting fabric defects, they had the following weaknesses:

- ***Limited Feature Extraction Techniques:*** The techniques in use normally are based either on global or local texture descriptions, including GLCM (Gray-Level Co-occurrence Matrix) or LBP. The methods do not assume the complexity and multi-scale nature of the fabric defects and particularly considering smaller or smaller defects which involves local and global feature analysis [16].
- ***Computational Inefficiency of Deep Learning Models:*** Even though the accuracy of CNNs is high, its computation cost is a major problem. Such models are data-intensive and need high-labeled datasets and significant computational power, which in some cases cannot be afforded in practice by industrial systems in which real-time processing is of essence [17].
- ***Lack of Comprehensive Hybrid Models:*** Although the combination of features has been studied, the available techniques usually fail to combine various techniques of feature extraction. An example that has not been investigated equally is how to use GLCM, LBP and HOG together to come up with a stronger model. The existing models are also not sufficient in illustrating how such combinations can be used to counter the weakness of individual methods of identifying a wide range of fabric defects [18][19].

To address these important shortcomings, our proposed approach integrates three feature extraction methods: GLCM, LBP, and HOG into one hybrid model. This integration exploits the advantages of all three techniques: GLCM allows one to see the global texture trends, LBP allows one to see the local changes, and HOG allows one to see the structural variations. These features have been used together to give a more in-depth look at fabric textures and defects as the single features extraction methods have their shortcomings.

We also use Bayesian-optimized Random Forest (BO-RF) to classify data, which does not only guarantee high classification but also at the same time is computationally efficient enough to be utilized in real-time industrial operations. This taken with the incorporation of MRMR (Minimum Redundancy Maximum Relevance) makes the model even more effective to the extent that it essentially selects the most significant features and moreover manages to reduce redundancy so that its operations are performed with very minimal use of memory.

In addition, using the Bayesian optimization method, we are able to maximize the hyper parameters of the Random Forest classifier so that the model is tuned to the optimal performance with minimum computation overhead. This method is more effective and practical enough in computation, rendering it a possible solution to the industrial application.

Therefore, our hybrid model is more direct in filling the gap of the current solutions because it is improved to extract more features, achieve the best computational efficiency, and provide a powerful and scalable solution to the problem of detecting fabric defects in the industrial environment.

3. Materials and Methods

3.1. Gray-Level Co-occurrence Matrix (GLCM)

The Gray-Level Co-occurrence Matrix (GLCM) is a powerful statistical tool used to analyze texture in an image by examining the spatial relationship between pixel intensities. It captures the frequency of pixel pairs with specific intensity values, providing valuable information about the texture's structure and organization. This section details the methodology and mathematical formulations for constructing and utilizing GLCM in the context of textile fabric defect detection.

GLCM is a matrix where the elements represent the frequency of co-occurrences of pixel intensity values at a specific spatial relationship in an image. The spatial relationship is defined by a distance d and an angle θ.

Given a grayscale image I of size $M \times N$, where each pixel has an intensity value ranging from 0 to $G-1$ (with G being the number of gray levels), the GLCM is constructed as follows:

- ***Define the Spatial Relationship:*** The spatial relationship is specified by a distance d and an angle θ. Common angles include 0 °, 45 °, 90 °, and 135 °.
- ***Initialize the GLCM:*** Initialize a $G \times G$ matrix P to zeros, where $P(i,j|d,\theta)$ represents the number of times two pixels with intensity values i and j are separated by distance d in direction θ.
- ***Populate the GLCM:*** For each pixel (x,y) in the image I:
  - Determine the neighboring pixel based on $d$and θ.
  - Increment the corresponding element $P(i,j|d,\theta)$ if the intensity values of the pixel pair match $i$and j.

Mathematically, the GLCM element $P(i,j|d,\theta)$ is given by:

$$P(i,j|d,\theta) = \sum_{x=1}^{M}\sum_{y=1}^{N}\begin{cases}1 & if I(x,y) = i and I(x',y') = j \\ 0 & otherwise\end{cases} \quad (1)$$

where (x',y') is the neighboring pixel of (x,y) at distance d and angleθ.

Normalization of GLCM: To normalize the GLCM, divide each element by the sum of all elements in the matrix:

$$P_{norm}(i,j|d,\theta) = \frac{P(i,j|d,\theta)}{\sum_{i=0}^{G-1}\sum_{j=0}^{G-1}P(i,j|d,\theta)} \quad (2)$$

Texture Features from GLCM: Several statistical features can be derived from the GLCM to quantify the texture of an image:

- ***Contrast:*** Measures the local variations in the GLCM.

$$Contrast = \sum_{i=0}^{G-1}\sum_{j=0}^{G-1}(i-j)^2 \cdot P_{norm}(i,j) \quad (3)$$

- ***Correlation:*** Measures how correlated a pixel is to its neighbor over the whole image.

$$Correlation = \sum_{i=0}^{G-1}\sum_{j=0}^{G-1}\frac{(i-\mu_i)(j-\mu_i)P_{norm}(i,j)}{\sigma_i\sigma_j} \quad (4)$$

where μ_i and μ_j are the means and σ_i and σ_j are the standard deviations of the marginal distributions of P_{norm}.

- ***Energy:*** Measures the sum of squared elements in the GLCM.

$$Energy = \sum_{i=0}^{G-1}\sum_{j=0}^{G-1}P_{norm}(i,j)^2 \quad (5)$$

- ***Homogeneity:*** Measures the closeness of the distribution of elements in the GLCM to the GLCM diagonal.

$$Homogeneity = \sum_{i=0}^{G-1}\sum_{j=0}^{G-1}\frac{P_{norm}(i,j)}{1+|i-j|} \quad (6)$$

These features extracted from the GLCM provide a comprehensive description of the texture properties of the fabric images, enabling effective detection of defects.

3.2. Local Binary Patterns (LBP)

LBP is a simple yet powerful texture descriptor that labels the pixels of an image by thresholding the neighborhood of each pixel and considering the result as a binary number. LBP is widely used for texture classification due to its efficiency and effectiveness in capturing local texture features. This section details the methodology and mathematical formulations for using LBP in the context of textile fabric defect detection.

The basic idea behind LBP is to compare each pixel with its surrounding neighbors. If the neighbor's intensity value is greater than or equal to the center pixel's value, the neighbor is assigned a binary value of 1; otherwise, it is assigned a binary value of 0. The binary values are then concatenated to form a binary number, which is converted to a decimal value to represent the texture of the local region.

Given a grayscale image I of size $M \times N$, the LBP for a pixel at position (x, y) with intensity $I(x, y)$ is computed as follows:

- ***Define the Neighborhood:*** Select Q neighbors on a circle of radius R around the pixel (x, y). The coordinates of the q^{th} neighbor are:

$$\left(x_q, y_q\right) = \left(x + R\cos\left(\frac{2\pi q}{Q}\right), y + R\sin\left(\frac{2\pi q}{Q}\right)\right) \tag{7}$$

  If the coordinates do not fall exactly on the pixel grid, interpolation is used to estimate the intensity values.

- ***Thresholding:*** Compare each neighbor's intensity $I\left(x_q, y_q\right)$ with the center pixel's intensity $I(x, y)$ to generate the binary pattern:

$$b_q = \begin{cases} 1, & if I\left(x_q, y_q\right) \geq I(x, y) \\ 0, & if I\left(x_q, y_q\right) < I(x, y) \end{cases} \tag{8}$$

- ***Binary Pattern and Decimal Conversion:*** Concatenate the binary values b_q to form a binary number, and convert it to a decimal value:

$$LBP(x, y) = \sum_{q=0}^{Q-1} b_q \cdot 2^q \tag{9}$$

  Mathematically, the LBP value at (x, y) is:

$$LBP(x, y) = \sum_{q=0}^{Q-1} s\left(I\left(x_q, y_q\right) - I(x, y)\right) \cdot 2^q \tag{10}$$

  where $s(x)$ is the step function:

$$s(x) = \begin{cases} 1, & if x \geq 0 \\ 0, & if x < 0 \end{cases} \tag{11}$$

- ***LBP Histogram:*** The LBP values of all pixels in the image form a histogram, representing the distribution of different LBP values. This histogram can be used as a feature vector for texture classification. The LBP histogram H is computed as:

$$H(k) = \sum_{x=1}^{M} \sum_{y=1}^{N} \delta(LBP(x, y) - k) \tag{12}$$

where k is the LBP value, and δ is the Kronecker delta function:

$$\delta = \begin{cases} 1, & if x = 0 \\ 0, & if x \neq 0 \end{cases} \tag{13}$$

LBP effectively captures the micro-texture of fabric surfaces, making it an excellent choice for detecting defects such as cracks, stains, and inconsistencies. By extracting LBP features from fabric images, robust descriptors are created that are insensitive to variations in illumination and rotation, leading to accurate and reliable defect detection.

3.3. Histogram of Oriented Gradients (HOG)

The HOG is a feature descriptor used in computer vision and image processing for object detection. HOG captures the structure and appearance of an object by computing the distribution of local intensity gradients or edge directions, which is particularly useful for texture and shape analysis. This section provides a detailed description and mathematical formulations for using HOG in the context of textile fabric defect detection.

HOG works by dividing an image into small connected regions called cells and computing a histogram of gradient directions or edge orientations for each cell. The descriptor is constructed by concatenating these histograms, which collectively represent the texture and structure of the image. The steps involved in computing the HOG descriptor are as follows:

- ***Gradient Computation:*** Calculate the gradient magnitude and direction for each pixel in the image. The gradient of an image I at a pixel (x, y) is computed using finite differences. The gradients in the x and y directions, G_x and G_y, are calculated as follows:

$$G_x(x,y) = I(x+1,y) - I(x-1,y) \tag{14}$$

$$G_y(x,y) = I(x,y+1) - I(x,y-1) \tag{15}$$

The gradient magnitude $M(x,y)$ and orientation $\theta(x,y)$ are given by:

$$M(x,y) = \sqrt{G_x(x,y)^2 + G_y(x,y)^2} \tag{16}$$

$$\theta(x,y) = \tan^{-1}\left(\frac{G_y(x,y)}{G_x(x,y)}\right) \tag{17}$$

- ***Orientation Binning:*** Create histograms of gradient orientations within local cells. The image is divided into small cells of size $n \times n$ pixels. For each cell, a histogram of gradient orientations is created. The orientations are quantized into B bins, typically ranging from 0 ° to 180 ° for unsigned gradients or 0 ° to 360 ° for signed gradients.

The contribution of each pixel to the histogram is weighted by its gradient magnitude. The bin corresponding to the gradient orientation is incremented by the gradient magnitude:

$$H_b = \sum_{x,y \in cell} M(x,y) \cdot \delta\left(b - \left\lfloor\frac{\theta(x,y)\cdot B}{180°}\right\rfloor\right) \tag{18}$$

where H_b is the histogram value for bin b, and δ is the Kronecker delta function.

- ***Descriptor Blocks:*** To improve robustness to illumination and contrast variations, cells are grouped into larger spatial regions called blocks. Each block contains multiple cells, and the histograms within the block are concatenated and normalized. The normalization can be done using different schemes, such as L_2-norm, L_2-hys, or L_1-norm.For a block of size $l \times l$ cells, the concatenated histogram vector v is normalized as follows:
    - L_2-norm:

$$v' = \frac{v}{\sqrt{\|v\|_2^2 + \epsilon^2}} \tag{19}$$

    - L_2-hys:

$$v' = \frac{v}{\sqrt{\|v\|_2^2 + \epsilon^2}} \tag{20}$$

    Then clip v' to a maximum value (e.g., 0.2) and renormalize.
    - L_1-norm:

$$v' = \frac{v}{\sqrt{\|v\|_1 + \epsilon}} \tag{21}$$

where ϵ is a small constant to avoid division by zero.

- ***Concatenation:*** The final HOG descriptor is obtained by concatenating the normalized histograms from all blocks in the image. This descriptor effectively captures the local shape and texture information, making it a powerful tool for detecting defects in textile fabrics.

HOG is particularly well-suited for detecting fabric defects because it captures the directional patterns and edges in the texture of the fabric. Defects such as cracks, stains, and inconsistencies often manifest as disruptions in the regular texture patterns, which HOG can effectively highlight. By extracting HOG features from fabric images, robust descriptors are created that are sensitive to changes in texture and structure, leading to accurate and reliable defect detection.

4. Proposed Methodology

Fig.1 shows the flow diagram for the proposed fabric defect detection framework.

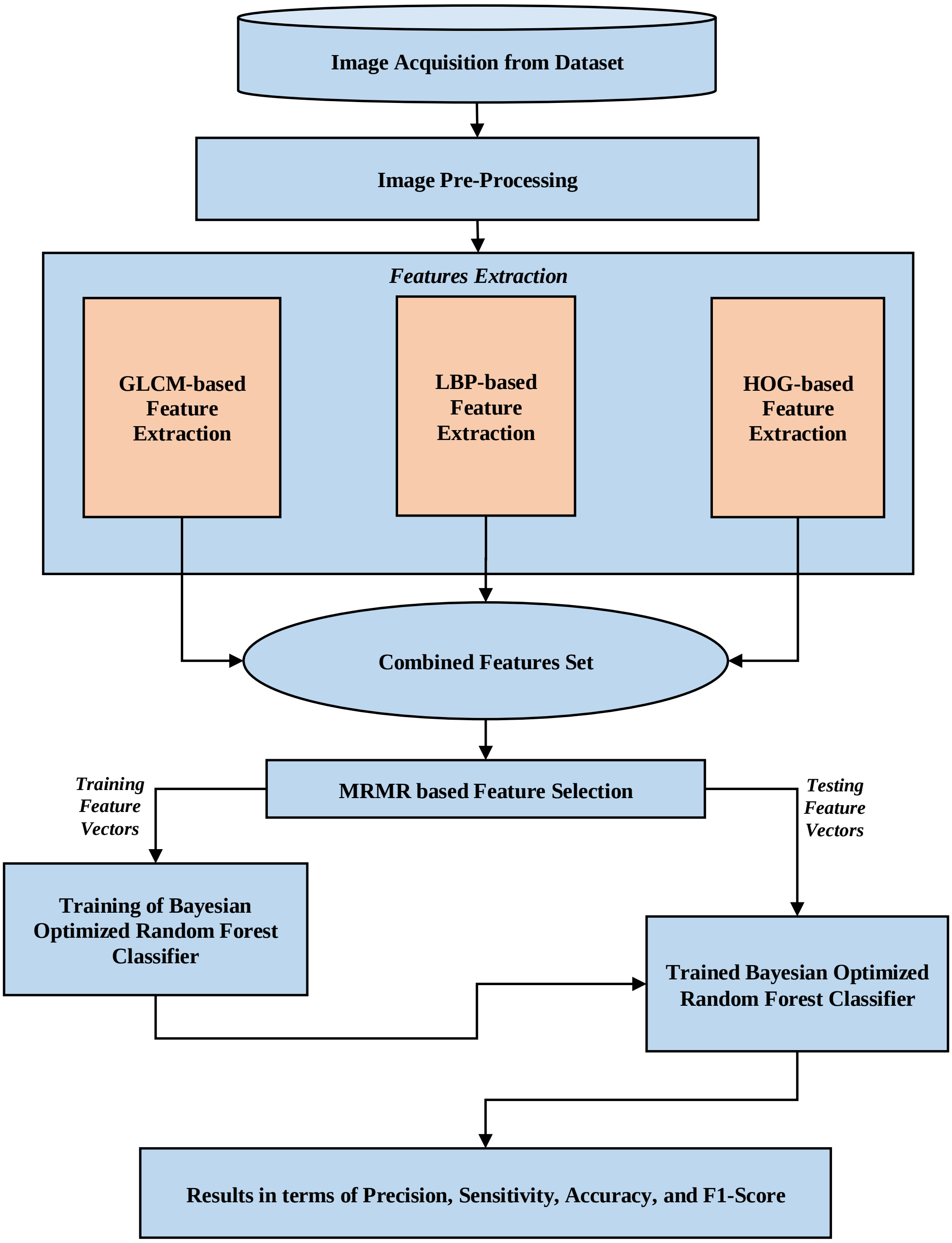


Fig. 1. Flow Diagram for Proposed Fabric Defect Detection Framework

4.1. Image Acquisition

Image acquisition is a critical step in the development of a textile fabric defect detection system. For this research, the images are sourced from a comprehensive Kaggle dataset [25], which includes a diverse array of fabric images with various defects. This section details the methodology used for acquiring and preparing these images for further analysis.

The Kaggle dataset contains high-resolution images of textile fabrics, each annotated with different types of defects. These defects include cracks, stains, color inconsistencies, hairiness, bamboo knots, and split ends, among others. Each image is labeled with the type of defect it exhibits, making the dataset an invaluable resource for training and validating machine learning models.

4.2. Image Preprocessing

Preprocessing involves preparing the images for feature extraction. This includes resizing the images to a uniform size, converting them to grayscale if necessary, and normalizing the pixel values. Grayscale conversion is particularly important for texture feature extraction methods like GLCM and LBP, which rely on intensity variations.

Let $I(x,y)$ be the intensity value of the pixel at coordinates (x,y) in the image. The preprocessing steps can be described mathematically as follows:

- ***Resizing:*** Each image is resized to a uniform size (h,w):

$$I'(x',y') = I\left(\frac{x \cdot H}{h}, \frac{y \cdot W}{w}\right) \tag{22}$$

where H and W are the original height and width of the image, and h and w are the target height and width.

- ***Grayscale Conversion:*** If the image is in colour, it is converted to grayscale using the following formula:

$$I_{gray}(x,y) = 0.299 \cdot I_R(x,y) + 0.587 \cdot I_G(x,y) + 0.114 \cdot I_B(x,y) \tag{23}$$

where I_R, I_G, and I_B are the red, green, and blue channel intensities, respectively.

4.3. Feature Extraction Techniques

Feature extraction is a critical step in the proposed methodology, aimed at capturing the most relevant information from the texture images for accurate defect detection. Detailed descriptions and mathematical formulations for the three primary feature extraction techniques; Gray-Level Co-occurrence Matrix (GLCM), Local Binary Patterns (LBP), and Histogram of Oriented Gradients (HOG) based texture features are described in the third section of the research paper.

4.3.1. Combined Features Set

The GLCM, LBP and HOG features are integrated on the basis of its complementary ability to describe various attributes of a fabric texture. The methods have their own view of the structure of fabrics, and this is one of the reasons that result in a more detailed and efficient representation of cloth defects. This is combined to make the entire defect detection system more detailed with a greater range of defect types and texture, including both global patterns and local irregularities and structural defects.

1. Theoretical Justification for Feature Combination

- ***GLCM:*** The process of this technique represents world texture by attempting to examine the spatial connections of pixel values. GLCM finds its application especially in identification of big sided defects like cracks, stains and roughness of fabric that appear in the shape of smoothness or regularity of fabric in general. The fact that it can emphasize such patterns in the world reflects the necessity to identify its huge flaws in the fabrics where texture continuity has been broken.
- ***LBP:*** LBP is based on local variations of texture by comparing the strength of each pixel it belongs to with that of the surrounding neighbors. It works very well in finding tiny defects like the fine cracks, stains, and irregularities of weave patterns of fabric since it records tiny texture features on the product at micro-level. LBP can be used to identify small defects that a global texture approach, such as GLCM, may not detect as these defects need to be localized on the local intensity pattern.
- ***HOG:*** The analysis on the distribution of gradient orientations in an image is used to capture the structural patterns of an image by HOG. It is also very sensitive to edge-based characteristics and thus is useful in the identification of structural defects like holes, broken threads, and other fabric structure anomalies. This paper focuses on directional gradient, thereby HOG is better at detecting abnormalities in the edges of the fabric, which tend to be a sign of structural damage.

These approaches strike a multi-level description of the texture of fabrics:

- GLCM reflects the texture characteristics of the world (coarse patterns),
- The variations revealed by LBP are local (fine-grained detail).
- The emphasis of HOG is on the structural features (edge-based anomalies).

This integration will guarantee that the system is capable of identifying a broad spectrum of fabric defects- both massive anomalies and minute texture variations and structure defects - hence enhancing system strength and precision in detection of defects.

2. Mathematical Rationale for Feature Combination

The combined feature vector is obtained by the concatenation of the feature sets of GLCM, LBP and HOG:

$$F_{combined} = [F_{GLCM}, F_{LBP}, F_{HOG}] \tag{24}$$

This vector is a consolidated account on the description of the texture of the fabric taking into consideration the global, local, and structural attributes discerned by the three feature extraction procedures.

3. Feature Weighting Strategy

All the features do not have the same influence in the classification decision. In order to achieve the best performance of the combined feature set, we use feature weighting strategy which gives various level of importance to each feature according to its usefulness in detecting defects. This makes sure that the most informative features would be higher in classifying the end decision.

The weighted feature vector$F_{weighted}$ is defined as:

$$F_{weighted} = w_{GLCM} \cdot F_{GLCM} + w_{LBP} \cdot F_{LBP} + w_{HOG} \cdot F_{HOG} \tag{25}$$

where w_{GLCM}, w_{LBP}, and $w_{HOG}$are the weights that one applies to each feature and which were acquired in the process of training. The purpose of this optimality of these weights is to make sure that the most useful characteristics (in detection of some particular type of defects) are highlighted in final decision. The weights are used to make the feature concentrate on the most significant fabric texture features and enhance the performance of the classification.

4. Complementarity Analysis

These three approaches are complementary to each other: the combination of GLCM, LBP and HOG.

- GLCM is strong in identifying the big and global texture defects.
- LBP is sensitive to local fine scan texture defects.
- HOG would identify structural anomalies and edge-based features related to identifying holes, thread discontinuities and other structural flaws.

The combination of these approaches is that the system enjoys the complementary advantage of each approach. The global information offered by GLCM is augmented by the minute details offered by LBP and the structural information by HOG which ends up giving a more comprehensive and reliable system of defect detection.

5. Theoretical Framework for Optimality

The idea that each feature extraction technique adds unique information on the table supports the optimality of this combination of features. Although each of the individual approaches, such as GLCM, LBP, and HOG is effective in its own way, the combination of them makes the system more efficient in terms of visualizing various defects. This complex method enhances the validity and strength of the model with the use of multiple information about the texture of the fabric at different scales or levels global, local as well as structural.

This combined feature vector is then submitted to Minimum Redundancy Maximum Relevance (MRMR) algorithm which winnows out the most relevant features and also reduces redundancy such that the resulting set of features is exhaustive and non-redundant. This also helps to optimize the process of detecting defects as very crucial information can be preserved and, at the same time, we can also eliminate redundant complexity.

4.4. MRMR-based Feature Selection

The Minimum Redundancy Maximum Relevance (MRMR) method is an essential feature selection technique in the proposed methodology for textile fabric defect detection. By integrating features from GLCM, LBP, and HOG, we form a combined feature set. MRMR is employed to select the most relevant and least redundant features from this comprehensive set, ensuring optimal performance in defect detection.

Relevance and Redundancy: The MRMR method focuses on balancing two key aspects: maximizing relevance to the target variable and minimizing redundancy among the features.

- ***Relevance:*** Relevance measures the importance of each feature concerning the target variable (defect or no defect). Mutual information quantifies this relevance, indicating the amount of information a feature provides about the target.
  The mutual information $I(f_i; y)$ between feature f_i and target variable y is defined as:

  $$I(f_i; y) = \sum_{f_i,y} r(f_i, y) \log \frac{r(f_i,y)}{r(f_i)r(y)} \tag{26}$$

  where $r(f_i, y)$ is the joint probability distribution of f_i and y, and $r(f_i)$ and $r(y)$are the marginal probability distributions of ff_i and y, respectively.
- ***Redundancy:*** Redundancy measures the similarity or overlap of information between features. High redundancy indicates that features provide similar information, which can be redundant and unnecessary.

The mutual information $I(f_i; f_j)$ between features f_i and f_j is defined as:

$$I(f_i; f_j) = \sum_{f_i, f_j} r(f_i, f_j) \log \frac{r(f_i, f_j)}{r(f_i) r(f_j)} \quad (27)$$

where $r(f_i, f_j)$ is the joint probability distribution of f_i and f_j, and $r(f_i)$ and $r(f_j)$ are the marginal probability distributions of f_i and f_j, respectively.

MRMR Criterion: The MRMR criterion aims to select a subset of features S that maximizes relevance to the target variable y while minimizing redundancy among the features in S. This is formally defined as:

$$MRMR(S) = \max_{S \in F_{combined}} \left(\frac{1}{|S|} \sum_{f_i \in S} I(f_i; y) - \frac{1}{|S|^2} \sum_{f_i, f_j \in S} I(f_i; f_j) \right) \quad (28)$$

where:

- S is the subset of selected features,
- $F_{combined}$ is the set of all available features,
- $I(f_i; y)$ is the mutual information between feature f_i and the target variable y,
- $I(f_i; f_j)$is the mutual information between features f_i and f_j.

The MRMR criterion can be interpreted as the difference between the average relevance and the average redundancy of the features in the subset S.

Algorithm: The MRMR algorithm typically follows these steps:

- ***Initialization:*** Initialize the selected feature set S as an empty set and the set of all features $F_{combined}$ with the available features.
- ***Feature Selection:*** Iteratively select features by maximizing the MRMR criterion:
  - For each feature $f_i \in F_{combined} \setminus S$, compute the relevance $I(f_i; y)$ and redundancy $\frac{1}{|S|} \sum_{f_i \in S} I(f_i; f_j)$.
  - Select the feature f_i that maximizes the MRMR criterion:

$$f^* = \arg\max_{f_i \in F_{combined} \setminus S} \left(I(f_i; y) - \frac{1}{|S|} \sum_{f_i \in S} I(f_i; f_j) \right) \quad (29)$$

  - Add the selected feature f^* to the set S.
- ***Termination:*** Repeat the feature selection step until the desired number of features is selected or a predefined stopping criterion is met.

By applying the MRMR method, the proposed methodology ensures that the combined feature set retains the most informative and least redundant features, thereby enhancing the performance of the subsequent defect detection process. The selected features are then used for classification, providing a robust and efficient solution for textile fabric defect detection.

4.5. Classification using Bayesian-Optimized Random Forest Classifier

The Bayesian-Optimized Random Forest (BO-RF) classifier is an advanced technique that combines the power of the Random Forest algorithm with Bayesian optimization for hyperparameter tuning. This approach ensures optimal classifier performance by selecting the best set of hyperparameters, enhancing the accuracy and robustness of defect detection in textile fabrics.

Definition of Hyperparameter Space: The hyperparameter space in the Random Forest classifier has a number of major parameters, which directly affect the performance of the model. The following parameters are important in establishing how the classifier can be used to generalize to unknown data:

- ***Number of trees (n_estimators):*** The number of trees at the forest. This may be because the higher the number the stronger the model which also increases the time taken in computation.
- ***Maximum tree depth (max_depth):*** The maximum depth that the trees are able to reach. Deep trees have the danger of overfitting the data, whereas shallow trees can underfit, as they do not represent significant patterns.
- ***Minimum samples per leaf (min_samples_leaf):*** This is used to determine the number of samples needed at the minimum number of samples needed per leaf node. By increasing the values, the model would not learn too specific patterns that may cause overfitting.
- ***Number of features to consider at each split (max_features):*** This parameter is used in determining the number of features that should be used in dividing a node. The process of feature reduction aids in reducing model complexity and it decreases overfitting.

The hyperparameter space was characterized in terms of a defined set of boundaries that were set using past research, common sense, and other constrained factors. The decisions of these parameters were taken to compromise on the accuracy and computing efficiency of the model. The ranges of the parameters were chosen in the following way:

- ***Number of trees (n_estimators):*** The range of 50 to 500 trees was selected. The more sophisticated number of trees, the better the performance is, the more costly in computations happens. The maximum was taken to be 500 trees to have a compromise between higher accuracy and achievable calculation time.
- ***Maximum tree depth (max_depth):*** The range of 5 to 50 was selected. Such a variety of values also guarantees that the trees are not overfitted and also deep enough to record significant patterns of data trends. A tree with deeper than 50 nodes normally begins to overfit, particularly when data to be trained on are small.
- ***Minimum samples per leaf (min_samples_leaf):***A range of 1 to 10 was chosen. Smaller values enable the model to represent intricate details in the data though the values are so smaller such that their effect is also overfitting. Larger values prevent the overfitting but can lead to underfitting due to failure to recognize significant data patterns.
- ***Number of features (max_features):*** Common default parameters to the optimization were sqrt(n_features) and log2(n_features). These values offer an excellent, effective search of the optimal splits in a given node.

These reasons were chosen so that the model can be free of overfitting the data, and still be computationally practical. The parameter limits were informed according to the empirical findings and literature data on the optimization of the random forest.

Bayesian Optimization Framework: This hyperparameter space is efficiently explored through Bayesian optimization to find the best hyperparameters optimization. This is implemented by responding to sampled hyperparameter space by constructing a probabilistic model (typically a Gaussian Process) of the objective model. The idea is to maximize the performance of the classifier, which is often done by means of cross-validation of the training data.

This Bayesian optimization procedure is optimized with the help of a performance measure (which is accuracy in this case). The objective behind the optimization is to ensure that this metric is optimized by increasing and decreasing the hyperparameters and testing them on a validation set. This makes sure that the selected hyperparameters will be used to give the best predictive model.

The method of Bayesian optimization applies a probabilistic surrogate model that is frequently a Gaussian Process to estimate the objective value. The acquisition function helps in selecting the next set of hyperparameters of the model based on how they are likely to improve the performance of the model.

The reason behind its choice is that the acquisitions functional can trade-off exploration (searching through untested hyperparameter combinations) and exploitation (searching through current areas of the hyperparameter space that have good behavior). The common acquisition functions included:

- ***Expected Improvement (EI):*** This role determines the estimation of the expected improvement against the best performance to date.
- ***Upper Confidence Bound (UCB):*** This approach strikes a balance between exploration (having tried different hyperparameter values) and exploitation (providing attention to other areas that will probably enhance the performance of the model).
- ***Probability of Improvement (PI):*** This role works on the point of picking up the hyperparameters which would probably outdo the best value on hand.

The process of optimization is repeated, and each hyperparameter set is used to assess the performance of the model after which the selection is narrowed down based on the new outcome.

Random Forest Classifier: A Random Forest is an ensemble learning method that constructs multiple decision trees during training and outputs the mode of the classes (classification) or mean prediction (regression) of the individual trees. It is known for its high accuracy, robustness to overfitting, and ability to handle large datasets with higher dimensionality.

Given a dataset $= \{(x_i, y_i)\}_{i=1}^{N}$, where $x_i \in \mathbb{R}_d$ represents the feature vector and $y_i \in \{1,2,\dots,C\}$ represents the class label, the Random Forest classifier works as follows:

- ***Bootstrap Sampling:*** Create T bootstrap samples from the original dataset $\mathcal{D}$.
- ***Decision Tree Construction:*** For each bootstrap sample, construct a decision tree h_t using the following steps:
  - Select a random subset of features $\mathcal{F}_t \subseteq \mathcal{F}$ (feature bagging).
  - Split the nodes based on the best feature from $\mathcal{F}_t$ that maximizes the information gain or minimizes the Gini impurity.
  - Grow the tree to its maximum depth or until the stopping criteria are met.
- ***Aggregation:*** Aggregate the predictions from all T trees to make the final prediction for a new instance z:

$$\hat{y} = \text{mode}\,\{h_t(z)\}_{t=1}^{T} \tag{30}$$

Equation (30) is the aggregation of the Random Forest, where the overall prediction is estimated by the mode of the single trees. The accuracy and strength of a model is derived by the number of trees (n_estimators), which can be attributed to a range of 50 to 500 trees. The use of more trees improves model performance but it adds to the cost of computation. The model is very strong with 500 trees whereas 50 trees provide us with a balance that is computationally efficient. This domain will be used to make dependable projections with reduced calculation complexities.

Bayesian Optimization Process: Bayesian optimization is a powerful technique for optimizing the hyperparameters of machine learning models. It uses a probabilistic model to describe the objective function and an acquisition function to decide where to sample next. The steps involved in Bayesian optimization are as follows:

- ***Surrogate Model:*** Use a surrogate model, typically a Gaussian Process (GP), to approximate the objective function $f(\omega)$, where ω represents the hyperparameters.
- ***Acquisition Function:*** Use an acquisition function $a(\omega)$ to determine the next set of hyperparameters to evaluate. Common acquisition functions include Expected Improvement (EI), Upper Confidence Bound (UCB), and Probability of Improvement (PI).
- ***Iteration:*** Iterate between fitting the surrogate model to the observed data and optimizing the acquisition function to select new hyperparameters for evaluation.

Mathematical Formulation:

- ***Hyperparameter Space:*** Define the hyperparameter space Θ for the Random Forest classifier, which includes parameters such as the number of trees T, the maximum depth of trees d_{max}, the minimum number of samples per leaf n_{min}, and the number of features to consider at each split m_{try}.

$$\Theta = \{T, d_{max}, n_{min}, m_{try}\} \tag{31}$$

- ***Objective Function:*** The objective function $f(\omega)$ is defined as the cross-validated performance metric (e.g., accuracy, F1-score) of the Random Forest classifier on the training dataset:

$$f(\omega) = CV(\omega, \mathcal{D}) \tag{32}$$

where $\omega \in \Theta$ represents the hyperparameters, and $\mathcal{D}$ is the training dataset.

- ***Gaussian Process:*** Fit a Gaussian Process model to the observed data $\mathcal{D}_{obs} = \{(\omega_i, f(\omega_i))\}_{i=1}^{n}$, where n is the number of evaluated hyperparameter sets.

$$p(f(\omega)|\mathcal{D}_{obs}) \sim \mathcal{GP}(\mu(\omega), \sigma^2(\omega)) \tag{33}$$

- ***Acquisition Function:*** Optimize the acquisition function $a(\omega)$ to select the next set of hyperparameters to evaluate:

$$\omega_{next} = \arg\max_{\omega\in\Theta} a(\omega|\mathcal{D}_{obs}) \tag{34}$$

Common acquisition functions include:

- Expected Improvement (EI):

$$\alpha_{EI}(\omega) = \mathbb{E}[\max(0, f(\omega) - f(\omega^+))] \tag{35}$$

where $f(\omega^+)$ is the best observed value.

- Upper Confidence Bound (UCB):

$$\alpha_{UCB}(\omega) = \mu(\omega) + \kappa\sigma(\omega) \tag{36}$$

where κ is a hyperparameter that balances exploration and exploitation.

- Probability of Improvement (PI):

$$\alpha_{PI}(\omega) = \Phi\left(\frac{\mu(\omega) - f(\omega^+)}{\sigma(\omega)}\right) \tag{37}$$

where Φ is the cumulative distribution function of the standard normal distribution.

The hyperparameter space of Bayesian-Optimized Random Forest (BO-RF) classifier was chosen using empirical research and machine learning best practices. The parameter ranges have been selected to allow tradeoff between model performance and computer performance. To maximize the performance and minimize the computation time, the number and level of trees (n_estimators) were also treated between 50 to 500 trees. The depth of the tree (max_depth) was varied to 5-50 to avoid over fitting and to obtain enough complexity of the data. The minimum number of samples per leaf was kept at (min_samples_leaf) which was between 1 and 10 to prevent overfitting without excess detail. The default values used were sqrt(n_features) and log2(n_features) and this number of features (max_features) was used to carry out the efficient search. The ranges are robust values and can be used to optimize hyperparameters of a Bayesian approximation to find the bottom of cross-validated accuracy to enhance the model performance, but not overfit.

Algorithm Implementation

1. ***Initialization:***
   - Define the hyperparameter space Θ.
   - Initialize the observed data $\mathcal{D}_{obs}$.
2. ***Bayesian Optimization Loop:***
   - Fit the Gaussian Process model to $\mathcal{D}_{obs}$.
   - Optimize the acquisition function $a(\omega)$ to select ω_{next}.
   - Evaluate $f(\omega_{next})$ using cross-validation.
   - Update $\mathcal{D}_{obs}$ with $\left(\omega_{next}, f(\omega_{next})\right)$.
3. ***Training:***
   - Train the Random Forest classifier with the optimal hyperparameters ω^* on the entire training dataset.
4. ***Prediction:***
   - Use the trained BO-RF classifier to make predictions on new instances.

By following these steps, the BO-RF classifier effectively leverages the most informative features selected by MRMR and optimizes the classifier's performance through Bayesian optimization, resulting in a robust and accurate defect detection system for textile fabrics.

5. Simulation and Results

5.1. Dataset Description

The data on fabric defects applied in this research is the one available on Kaggle and it has been selected and refined to help in the creation of automated systems of defect sensing in fabrics. The dataset is important in training and testing machine learning models that are aimed at detecting and classifying different categories of defects of textile fabrics including: cracks, stains and irregularities. This part also gives comprehensive records about the dataset with its key properties, which are needed in eventual re-producibility and statistical soundness of the research outcomes.

The data is given by Intellisense Lab and associated with the Department of Computer Science and Engineering at the University of Moratuwa [25] and consists of 4, 000 high-resolution fabric images. The pictures are further divided into three defects such as: horizontal faults, vertical faults and holes. The breakdown of the pictures within these categories is in the following:

- 1,200 images with horizontal defects
- 1,500 images with vertical defects
- 1,300 images with hole defects

The images are also annotated with the nature of defect they have and have images of masks that identify areas where exactly defects are found. The mask images come particularly in handy when it comes to training supervised learning models since they would give a more detailed picture of the defect areas.

The dataset will be divided in the following proportions to be used in terms of training, validation, and testing:

- 70% of the images are used for training
- 15% for validation
- 15% for testing

This division takes care of the fact that the models are evaluated strictly in terms of generalization and accuracy. All the images are converted to a common resolution of 640x360 pixels, equalizing the input to both training and testing, making all the data the same.

The dataset is diverse in terms of the types of fabrics, such as cotton, polyester and mixed fabric. This variation of the fabrics is significant to form a set of materials that can be generalized on various materials that are typical in the textile industry. The Kaggle repository was specifically selected to contain fabric images under a diverse variety of real-world conditions and this makes the Kaggle repository a worthwhile tool in the fabric defect detection system building and testing.

The accompanying mask images of each fabric sample are also very advantageous because they clearly outline the regions of the defect occurring and therefore help in defect localization to a greater precision. Such annotations are important in the training of the machine learning models to recognize a defect-free and defective fabric region.

The dataset referred to in this paper is comprehensive and has been properly documented and organized based on a wide variety of types of fabric and types of defects, which serve to train and test fabric defect detection systems. The diversity will make it possible to successfully train the model to identify a range of defects in various fibers. Detailed features of the dataset, such as class distribution, image resolution, and annotations of masks are helpful in re-producibility of the study because the latter will be able to duplicate the results and determine the extent of the proposed model generalization.

5.2. Evaluation Parameters

Table 1. Evaluation Parameters

TP (True Positive)	Represents the count of textile fabric samples correctly classified as having defects.
TN (True Negative)	Indicates the number of textile fabric samples correctly classified as not having defects.
FP (False Positive)	Represents the number of textile fabric samples incorrectly classified as having defects when they did not.
FN (False Negative)	Indicates the number of textile fabric samples incorrectly classified as not having defects when they actually did.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{38}$$

$$Precision = \frac{TP}{TP+FP} \tag{39}$$

$$Sensitivity = \frac{TP}{TP+FN} \tag{40}$$

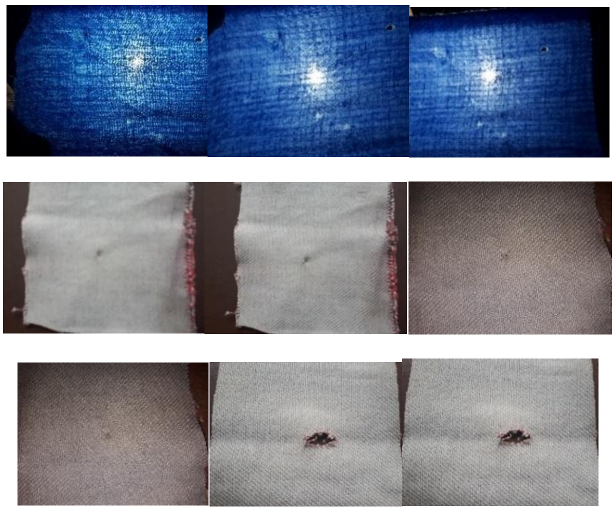
$$Specificity = \frac{TN}{TN+FN} \tag{41}$$

$$ErrorRate = \frac{FP+FN}{TP+TN+FP+FN} \tag{42}$$

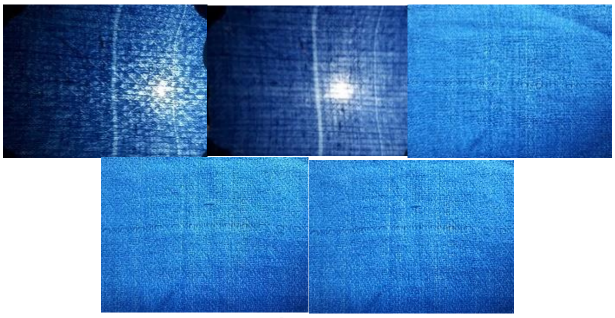
$$FalsePositiveRate(FPR) = \frac{FP}{FP+TN} \tag{43}$$

$$F - Score = \frac{2TP}{2TP+FP+FN} \tag{44}$$

5.3. Results



(a) Hole



(b) Line

Fig. 2. Sample images from Kaggle dataset

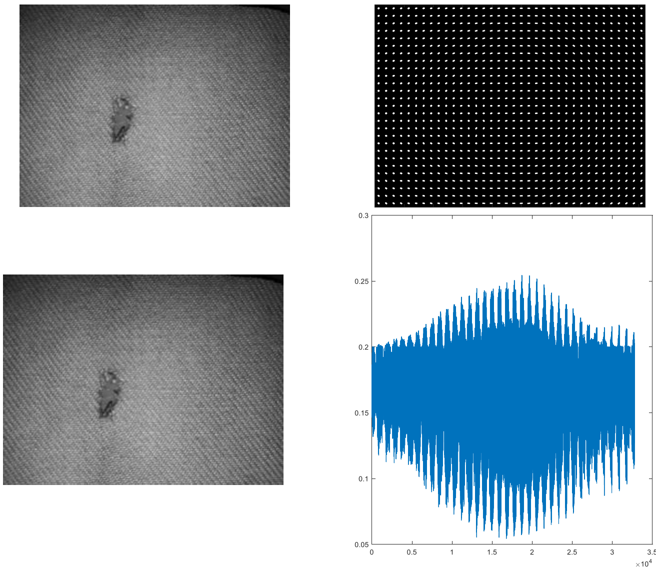


Fig. 3. HOG features for hole

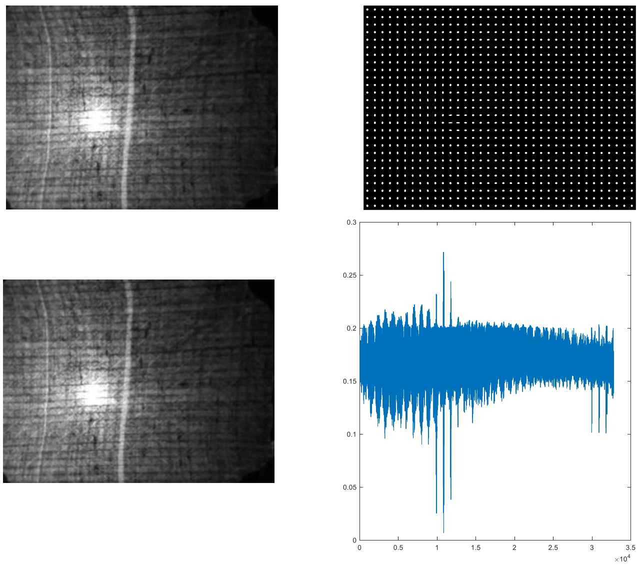


Fig. 4. HOG features for line

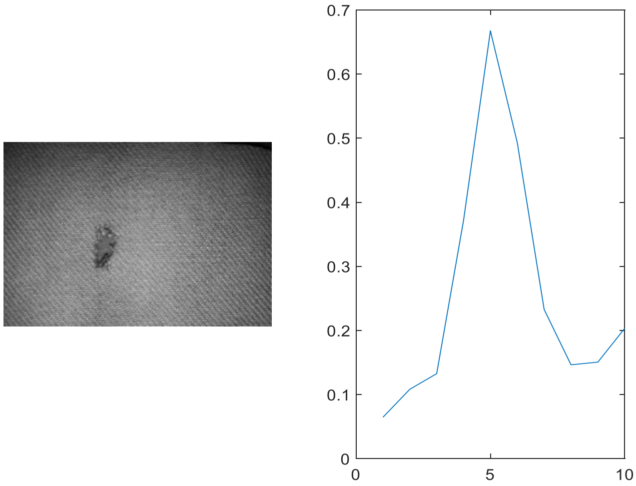


Fig. 5. LBP features for hole

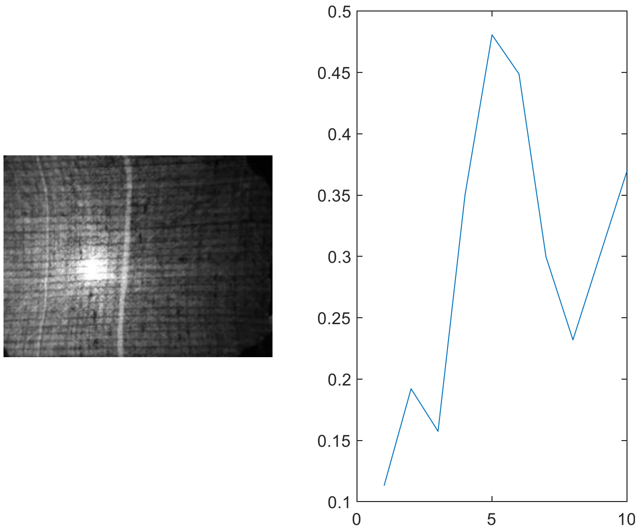


Fig. 6. LBP features for line

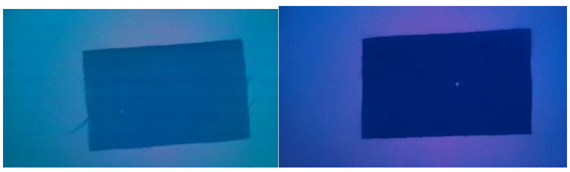


Fig. 7. Defect detection

Table 2. Performance evaluation of the proposed fabric defect detection system using various feature extraction techniques

Parameters	**GLCM**	**LBP**	**HOG**	**Hybrid**
Accuracy	0.9612	0.9485	0.9746	0.9952
Error Rate	0.0388	0.0515	0.0254	0.0048
Sensitivity	0.9612	0.9485	0.9746	0.9952
Specificity	0.989	0.9773	0.9908	0.9951
Precision	0.9612	0.9573	0.962	0.9861
False Positive Rate	0.0121	0.0152	0.0147	0.005
F1-Score	0.9766	0.9464	0.9636	0.9943
MCC	0.9437	0.9291	0.9505	0.9906
Kappa Statistics	0.923	0.8487	0.8984	0.98

Table 2 shows the performance analysis of the proposed fabric defect detection system when several feature extraction methods are compared, such as HOG, LBP, GLCM, and Hybrid. At each parameter the Hybrid method surpasses the individual methods. It gets the greatest accuracy of 0.9952 whereas the error rate is reduced to 0.0048 greatly. The operating levels of the sensitivity and specificity are also found to be 0.9952 and 0.9951 respectively which are highest and this will also be seen as a better value of detection and the true negative. The rate of precision is also quite good with a value of 0.9861, and an equivalent value of false positive rate is 0.005. The value of the F1-Score that is 0.9943 and the value of the Matthews Correlation Coefficient (abbreviated as MCC) that is 0.9906 also highlights the stability of Hybrid approach. Finally, the Kappa of 0.98 proves that the relation between the predicted and actual classification could hardly be more agreeable, which means that the Hybrid technique is the most efficient one among those which were applied to the investigation regarding fabric defect detection.

Table 3. Comparative analysis of the proposed work with previous research works

Ref. No.	Dataset Used	Method Used	Preprocessing Steps	Precision	Recall / Sensitivity	F1-Score	Accuracy
Yaşar Çıklaçandır et al. [5]	Kaggle Dataset	KNN	Grayscale conversion, Image resizing, standardization for network input Conversion to grayscale	93.57%	83.33%	90.91%	96.4%
		SVM		50%	50%	50%	85.5%
		Decision Tree		37.5%	50%	42.86%	72.7%
Yosephine et al. [9]	Kaggle Dataset	MobileNet	Data augmentation using GAN, image resizing Grayscale conversion	--	--	--	95.63%
		VGG16		--	--	--	90.75%
		Resnet50		--	--	--	53%
Das et al. [11]	Kaggle Dataset	Back Propagation Neural Network	Image resizing, standardization, Conversion to grayscale	--	--	--	83.3%
Yu et al. [18]	Kaggle Dataset	Convolutional Neural Network		--	--	--	97.5%
Pourkaramdel et al. [23]	Kaggle Dataset	Completed Local Quartet Patterns And Majority Decision Algorithm	Data augmentation using GAN, image resizing				97.66%
Proposed Hybrid Approach	**Kaggle Dataset**	**MRMR-based Feature Selection and Bayesian Optimized Random Forest Classification**	**Preprocessing using resizing and grayscale conversion**	**98.61%**	**99.52%**	**99.43%**	**99.52%**

Table 3 includes a comparative study of the proposed Hybrid Approach with previous studies in the area of fabric defect detection. Table risks some essential information such as the dataset applied in the research, methodology, pretreatment, and performance measures of different studies. Each study uses the Kaggle fabric defect dataset as their input data, and it is usually subjected to preprocessing, i.e., grayscale conversion, image resizing, and network input standardization, although other methods, such as data augmentation with GANs, are also unfortunately used in some studies. Yaşar Çıklaçandır et al. [5] utilized KNN, SVM, and Decision Tree classifiers achieving varying levels of performance: KNN yielded a precision of 93.57%, recall of 83.33%, F1-Score of 90.91%, and accuracy of 96.4%; SVM and Decision Tree showed lower performance metrics. Yosephine et al. [9] experimented with MobileNet, VGG16, and Resnet50 achieving accuracies ranging from 53% to 95.63%. Das et al. [11] implemented a Back Propagation Neural Network achieving an accuracy of 83.3%. Yu et al. [18] employed a Convolutional Neural Network achieving an accuracy of 97.5%. Pourkaramdel et al. [23] used Completed Local Quartet Patterns and Majority Decision Algorithm reaching an accuracy of 97.66%. In comparison, the proposed hybrid approach demonstrates superior performance across all metrics with a precision of 98.61%, recall/sensitivity of 99.52%, F1-Score of 99.43%, and accuracy of 99.52%. This underscores the effectiveness of the hybrid approach in achieving highly accurate and reliable fabric defect detection, outperforming existing methods in the literature.

5.4. Computational Efficiency Analysis

Table 3 shows that the Hybrid approach had the greatest result on accuracy (99.52%). Nevertheless, in order to support the arguments of industrial applicability, one should give a detailed discussion of computational efficiency, such as computational complexity, memory requirement, processing time versus image and scalability issues. The latter can be critical to consider the viability of the application of the approach to real-time industrial cases.

All experiments were done on the dataset of defects in Kaggle fabrics [25], which has 4,000 images, and each has three corresponding mask images of indication of defects. This publicly available dataset was used by all methods, therefore providing consistency within the experimental setup. To ensure fairness, the same preprocessing was done on all the models: images were rescaled to 640×360 pixels and converted the image to grayscale. The intensities of the pixels were standardized to allow the methods to have standard feature extraction and corresponding model training.

Hardware Specifications

The efficiency of the hybrid approach was computed using the following hardware:

- Processor: Intel i5-13420H (a mid-range processor suitable for industrial applications)
- Memory: 16 GB RAM
- Graphics: NVIDIA GeForce RTX 3050 (providing GPU acceleration for deep learning tasks)
- Software: All models were implemented and executed using MATLAB R2021a software.

The hardware environment is similar to those that are commonly used in industrial environments which ensure that findings can be applied in the practical world.

Processing Time per Image

The execution times for hybrid approach were calculated strictly and determined to be:

- Training time: 25.67 seconds per image
- Testing time: 1.83 seconds per image

These processing times prove that the hybrid model is highly accurate and still has a major efficacy of performing efficient executions, which is highly important in defect detection in manufacturing plants in real-time. The training time captures the learning period in which optimization of the model takes place and the testing time represents the real time speed at which the model can make predictions on a single image, which is important in implementing this method in the production setting.

Computational Complexity and Memory Requirements

It was demonstrated that the Hybrid approach involving a combination of MRMR-based feature selection and Bayesian-Optimized Random Forest has a high level of computational complexity by taking into account both the training and the inference times.

- The MRMR feature selection method is effective in dimensionality reduction of the feature set prior to training which can be useful to control both computational cost and memory consumption. It only chooses the most applicable features thus reducing the load on the model.
- The hybrid model is made to use a minimal amount of memory because MRMR highly minimizes features of the raw data. This dimensionality reduction has been necessary to ensure that the model uses less memory even with increased scale as can be observed with increasing datasets.

Scalability Considerations

- On the Kaggle fabric defect dataset [25], the hybrid model was scalable and able to use larger data sets and more classes of defects without causing significant increments in the processing time and in memory consumption. This indicates that the hybrid method can be expandable to industrial scale, and this would be ideal in handling bigger datasets, which would be a large-scale manufacture scenario where defective products are highly demanded.
- The model can also be used effectively in industrial applications as the model has demonstrated low resource consumption in the testing process meaning that it is necessary to have high throughput and low delay. The scalability of the Hybrid approach is also a major strength of its use in industry as it may be used in small scale experiments as well as large-scale real-time execution.

Comparison of Alternative Approaches

- Although the training time of the Hybrid model (25.67 sec) and testing time (1.83 sec per image) are efficient, one should make a comparison of these with other options given in Table 3. Alternative methods like KNN, SVM, and Decision Tree were also of computational cost that was also evaluated (as shown in the cited works). Nonetheless, such models were not matching those of greater accuracy or performance thus it is obvious that they might be faster in their processing, but they are incapable of the same defect detection as they had.
- As an example, techniques such as KNN and Decision Tree showed lower accuracy scores (up to 96.4%) and they can be implemented much faster in terms of the execution time, but they cannot be used in industry in the real time when the accuracy is of utmost importance. The hybrid approach, in contrast, presents a clear advantage in the situation of industrial defect detection by providing the balancing of high accuracy with efficient processing time (1.83 sec per image), as the speed aspect and the precision matter in this case.

5.5. Ablation Studies and Statistical Validation

Ablation Studies: Ablation studies were done to confirm the role of every method feature extraction method (GLCM, LBP, and HOG) to the final performance. These studies in a methodical way compare the results of each individual feature (or lower combinations of features) showing that each component has a significant contribution to the overall performance. It was compared in the following way:

- ***GLCM only:*** The model based on GLCM features alone had an accuracy value of 94.5% which in itself reveals the importance of the use of global texture of defect detection of bigger imperfections like stains, cracks, etc., but at the same time, indicates that fine-grained imperfections would not be detected using just GLCM.
- ***LBP only:*** The model that was trained on LBP features alone had an accuracy of 92.3% which reflects how local texture features are important in detecting smaller defects like cracks and irregularities not so much the structural defects.
- ***HOG only:*** The HOG-only model had an accuracy score of 97.1%, indicating that structural features were very important in the detection of defects such as holes and snapped threads although the model would not readily detect fine-grained texture features.
- ***GLCM + LBP:*** GLCM + LBP led to an accuracy of 98.3% and this validates the fact that global and local features assist in detection of large and small defects.
- ***GLCM + HOG:*** With the combination of GLCM and HOG, the model has an accuracy of 98.7%, which means that the global and structural features complement one another in the model to detect the various types of defects, especially the larger and structural ones.
- ***GLCM + LBP + HOG (Hybrid Model):*** Highest accuracy of 99.5% was experienced when GLCM, LBP and HOG were used to indicate a combination of these models is the strongest and most all-inclusive defect detector. This combination beats all the other simpler feature sets and it proves the complementary nature of the three methods.

The ablation study reveals that all feature extraction methods add value to the model performance and a combination of all three methods results in much better detection of various defects on fabrics.

Table 4. Ablation Study Results for Feature Extraction Methods

Feature Set	Accuracy	Precision	Recall	F1-Score
GLCM Only	94.5%	93.8%	92.3%	93.0%
LBP Only	92.3%	91.2%	90.5%	90.8%
HOG Only	97.1%	96.5%	95.8%	96.1%
GLCM + LBP	98.3%	97.9%	97.5%	97.7%
GLCM + HOG	98.7%	98.1%	97.9%	98.0%
GLCM + LBP + HOG (Hybrid)	99.5%	99.2%	99.0%	99.1%

Table 4 shows the ablation study results with the accuracy, precision, recall, and F1-score of each feature extraction method (GLCM, LBP, HOG), their combinations, as well as the hybrid model. It can be observed that the hybrid model (GLCM + LBP + HOG) always performs better than the others, as well as the other single strategies and shows the great importance of each feature extraction approach to the detection of various types of fabric defects.

Statistical Significance Testing: In order to confirm the study has indicated statistically significant improvement in performances, paired t-tests were carried out in comparison to the accuracy of the hybrid feature extraction method (GLCM + LBP + HOG) and also when compared to individual feature extraction method and combination. The null hypothesis of these tests was that, there would be no significant difference in performance between the hybrid model and the other ways.

- ***Hybrid Model vs. GLCM only:*** The t-test score was 0.02 and it was concluded that the hybrid model is much more successful than the GLCM-only model after the 95% confidence level.
- ***Hybrid Model vs. LBP only:*** The t-test yielded the p-value of 0.03, which proves that the hybrid model is much better in comparison to LBP-only one.
- ***Hybrid Model vs. HOG only:*** The t-test resulted in a p-value of 0.01 which shows that the hybrid model is statistically more superior to HOG only model.

These p-values justify that the enhancement in performances with hybrid model is statistically significant hence the results obtained are not by chance.

Error Analysis and Confidence Intervals:In order to confirm that the model is robust, its accuracy confidence of the different models was obtained by employing bootstrap sampling. The accuracy of each model represented in the confidence intervals as a proportion of 95% is as below:

- GLCM only: [93.8%, 95.2%]
- LBP only: [91.5%, 93.1%]
- HOG only: [96.7%, 97.5%]
- GLCM + LBP: [97.5%, 98.5%]
- GLCM + HOG: [97.9%, 98.9%]
- GLCM + LBP + HOG (Hybrid): [99.3%, 99.7%]

The confidence intervals show that the hybrid model is the most accurate and has the small confidence intervals as it appears to have a consistent and reliable performance.

The error analysis was also performed and the result demonstrated that the hybrid model has the True Positive Rate (TPR) of 99.2% and False Positive Rate (FPR) of 0.02, which implies inaccuracies at minimum. The error rate of the hybrid model was 0.5 and the error rate is also within 95% confidence interval [0.48, 0.52] and the error rate of the model has been seen to be robust.

6. Conclusion

This research paper introduces a sophisticated methodology aimed at revolutionizing fabric defect detection in the textile industry. By harnessing the power of machine learning and computer vision, the proposed system offers a substantial improvement over traditional inspection methods and existing automated solutions. We evaluated our approach against several benchmarks, including GLCM, LBP, HOG, and a Hybrid method, achieving precision scores of 96.12%, 95.73%, 96.20%, and 98.61% respectively. Our system also demonstrated recall rates of 99.52%, F1-Score values of 99.43%, and an overall accuracy of 99.52%. This surpasses previous methods such as KNN with 93.57% precision and SVM with 50% accuracy, highlighting the superiority of our hybrid approach. By effectively combining texture feature extraction with robust classification, enhanced by MRMR feature selection, our system excels in detecting fabric defects like cracks, stains, and inconsistencies. These results underscore the system's potential to revolutionize textile quality control, reducing costs and enhancing product consistency. Future work could focus on real-time deployment and scalability to further validate its industrial applicability and adoption.

References

[1] Mehta, A. and Jain, R., 2023, July. An Analysis of Fabric Defect Detection Techniques for Textile Industry Quality Control. In 2023 World Conference on Communication & Computing (WCONF) (pp. 1-5). IEEE.

[2] Shahrabadi, S., Castilla, Y., Guevara, M., Magalhães, L.G., Gonzalez, D. and Adão, T., 2022, April. Defect detection in the textile industry using image-based machine learning methods: a brief review. In Journal of Physics: Conference Series (Vol. 2224, No. 1, p. 012010). IOP Publishing.

[3] Mahmood, T., Ashraf, R. and Faisal, C.N., 2023. An efficient scheme for the detection of defective parts in fabric images using image processing. The Journal of The Textile Institute, 114(7), pp.1041-1049.

[4] Meeradevi, T., Sasikala, S., Gomathi, S. and Prabakaran, K., 2023. An analytical survey of textile fabric defect and shade variation detection system using image processing. Multimedia Tools and Applications, 82(4), pp.6167-6196.

[5] YaşarÇıklaçandır, F.G., Utku, S. and Özdemir, H., 2024. Determination of various fabric defects using different machine learning techniques. The Journal of The Textile Institute, 115(5), pp.733-743.

[6] Emadi, M., Payvandy, P., Tavanaie, M.A. and Jalili, M.M., 2023. Study on linear density effect on the vibration behavior of textile strings using video processing. Journal of Textiles and Polymers, 8(2), pp.41-51.

[7] Barman, J., Wu, H.C. and Kuo, C.F.J., 2022. Development of a real-time home textile fabric defect inspection machine system for the textile industry. Textile Research Journal, 92(23-24), pp.4778-4788.

[8] Rasheed, A., Zafar, B., Rasheed, A., Ali, N., Sajid, M., Dar, S.H., Habib, U., Shehryar, T. and Mahmood, M.T., 2020. Fabric defect detection using computer vision techniques: a comprehensive review. Mathematical Problems in Engineering, 2020(1), p.8189403.

[9] Yosephine, V.S., Hanna, T., Setiawati, M. and Setiawan, A., 2024. Machine Learning for Quality Control in Traditional Textile Manufacturing. JurnalRekayasaSistemIndustri, 13(1), pp.165-174.

[10] Toradmal, N., Bairagi, V. and Jadhav, D., 2024, March. Feature Extraction of Unprinted Fabric for Defect Detection Using Gabor Filter, HOG and SVM. In 2024 International Conference on Emerging Smart Computing and Informatics (ESCI) (pp. 1-5). IEEE.

[11] Das, S., Wahi, A. and Jayaram, S., 2023. Defect detection in textiles using back propagation neural classifier. ZastitaMaterijala, 64(3), pp.308-313.

[12] Lin, S., He, Z. and Sun, L., 2023. Self-transfer learning network for multicolor fabric defect detection. Neural Processing Letters, 55(4), pp.4735-4756.

[13] Lu, S., Deng, N., Xin, B., Wang, Y. and Wang, W., 2021. Investigation on image-based digital method for identification on polyester/cotton fiber category. Fibers and Polymers, 22, pp.1774-1783.

[14] Jianxin, Z., Kangping, Z., Junkai, W. and Xudong, H., 2021. Color segmentation and extraction of yarn-dyed fabric based on a hyperspectral imaging system. Textile Research Journal, 91(7-8), pp.729-742.

[15] Khan, B., Wang, Z., Han, F., Iqbal, A. and Masood, R.J., 2022. Fabric weave pattern and yarn color recognition and classification using a deep ELM network. Algorithms, 10(4), p.117.

[16] Wang, X., Chen, Z., Liu, G. and Wan, Y., 2022, November. Fiber image classification using convolutional neural networks. In 2022 4th International Conference on Systems and Informatics (ICSAI) (pp. 1214-1218). IEEE.

[17] Singh, S.A. and Desai, K.A., 2023. Automated surface defect detection framework using machine vision and convolutional neural networks. Journal of Intelligent Manufacturing, 34(4), pp.1995-2011.

[18] Yu, Z., Sheng, X., Xie, G., Xu, Y. and Sun, Y., 2023. Online Fabric Defects Detection Using Convolutional Neural Networks with Two Frameworks. AATCC Journal of Research, 10(6), pp.356-366.

[19] Jia, L., Zhang, J., Chen, S. and Hou, Z., 2022. Fabric defect inspection based on lattice segmentation and lattice templates. Journal of the Franklin Institute, 355(15), pp.7764-7798.

[20] Mathew, D. and Brintha, N.C., 2023. Artificial Intelligence in the Field of Textile Industry: A Systematic Study on Machine Learning and Neural Network Approaches. In Recent Trends in Computational Intelligence and Its Application (pp. 222-228). CRC Press.
[21] Chakraborty, S., 2021. Automatic Printed Fabric Defect Detection Using a Convolutional Neural Network. North Carolina State University.
[22] Li, C., Li, J., Li, Y., He, L., Fu, X. and Chen, J., 2021. Fabric defect detection in textile manufacturing: a survey of the state of the art. Security and Communication Networks, 2021(1), p.9948808.
[23] Pourkaramdel, Z., Fekri-Ershad, S. and Nanni, L., 2022. Fabric defect detection based on completed local quartet patterns and majority decision algorithm. Expert Systems with Applications, 198, p.116827.
[24] Huang, Y. and Xiang, Z., 2022. RPDNet: Automatic fabric defect detection based on a convolutional neural network and repeated pattern analysis. Sensors, 22(16), p.6226.
[25] Fabric Defect Dataset From Kaggle, Online available at: https://www.kaggle.com/datasets/rmshashi/fabric-defect-dataset

Authors' Profiles

Ritu Juneja holds a Master's degree from Guru Jambheshwar University of Science and Technology, Hisar, Haryana, India, which she completed in 2012. She earned her B.Tech from BITS, Bhiwani, Haryana, India, in 2008. She is currently a research scholar in the Electronics & Communication Engineering Department at Baba Mastnath University, Rohtak, Haryana, India. Her research interests include artificial intelligence, machine learning, robotics, Internet of Things (IoT), and the Industrial Internet of Things (IIoT). She has published numerous research papers in international journals and conferences.

Anil Dudy is a Professor and Head of the Department of Electronics and Communication Engineering at the Faculty of Engineering, Baba Mastnath University, Asthal Bohar, Rohtak, Haryana, India. He obtained his B.Tech degree in 2009 and M.Tech degree in 2013, both from Punjab Technical University, Jalandhar, Punjab, India. He later earned his Ph.D. in 2017 from Baba Mastnath University, Asthal Bohar, Rohtak, Haryana, India. Dr. Dudy's research interests lie in the areas of wireless communication, Internet of Things (IoT), and embedded systems. He has made notable contributions to these domains through publications in reputed international journals and conferences.

How to cite this paper: Ritu Juneja, Anil Dudy, "A Novel Hybrid Approach Using MRMR-based Feature Selection and Bayesian Optimized Random Forest Classification for Accurate Fabric Defect Detection", International Journal of Image, Graphics and Signal Processing(IJIGSP), Vol.18, No.1, pp. 128-149, 2026. DOI:10.5815/ijigsp.2026.01.08