

# "Automated Fabric Defect Detection Using Classical Computer Vision Techniques"

Parmeshwari Kokare<sup>1</sup>, Zoya Tamboli<sup>1</sup>, Antara Khataavkar<sup>1</sup>, Mrs. S. G. Bavachkar<sup>2</sup>

<sup>1</sup> UG Students, Department of Computer Engineering, Sanjay Bhokare Group of Institutes, Miraj, India <sup>2</sup> Assistant Professor, Department of Computer Engineering, Sanjay Bhokare Group of Institutes, Miraj India.

**Abstract** - Fabric defect detection is a critical quality control process in the textile industry, where the early identification of defects helps reduce material wastage and production costs. Conventional manual inspection methods are labor-intensive, time-consuming, and prone to inconsistencies owing to human fatigue. This study presents an automated fabric defect detection system based on classical computer vision techniques, designed to provide a fast and cost-effective alternative that does not rely on large datasets or deep learning models.

The proposed approach processes input images acquired from static sources and real-time camera feeds. The methodology includes grayscale conversion, Gaussian filtering for noise reduction, adaptive thresholding for local segmentation, and morphological operations to refine the defect regions. Contour detection is employed to identify irregular patterns, followed by area-based filtering to eliminate insignificant noise. Detected defects are highlighted using bounding boxes, and quantitative metrics such as defect count, defect area, coverage percentage, and quality estimation are computed.

A graphical user interface is developed to facilitate user interaction, enabling image upload, live monitoring, and real

time visualization of the results. Experimental evaluation demonstrates that the system achieves an efficient detection performance with low computational overhead and supports real-time processing on standard hardware. Although this approach does not incorporate deep learning, it provides a lightweight and practical solution for small and medium-scale textile industries.

The proposed system highlights the effectiveness of classical image processing techniques in industrial automation and serves as a foundation for future enhancements, including integration with deep learning models for improved accuracy and defect classification capabilities.

**Keywords:** Fabric defect detection, textile control, image processing, pattern recognition, defect

classification, anomaly, computer vision, product quality.

## 1. INTRODUCTION

Quality inspection is a crucial stage in textile manufacturing, where the presence of defects can significantly affect the usability, appearance, and market value of fabric products. Common defects, such as holes, stains, misweaves, and irregular textures often occur during the production process owing to machine faults, raw material issues, or handling errors [3],[5]. Detecting these defects at an early stage is essential for reducing material wastage, ensure product consistency, and maintaining industry standards [7].

Traditionally, fabric inspection is performed manually by trained operators who visually examine the material. However, manual inspection suffers from several limitations, including high labor cost, low speed, fatigue-induced errors, and lack of consistency [2]. As production volumes increase, the need for automated and reliable inspection system becomes critical. In recent years, computer vision-based approaches have gained significant attention for automating defect detection [4],[6].

Existing automated methods can be broadly categorized into classical image processing techniques and deep learning-based approaches. Deep learning models, particularly Convolutional Neural Networks (CNNs), have demonstrated high accuracy in defect detection and classification [10],[12]. However, these methods require large labeled datasets, extensive training time, and high computational resources such as GPUs, which may not be feasible for small- and medium-scale industries [8],[9].

In contrast, classical computer vision techniques offer a lightweight and efficient alternatives for defect detection [2],[6]. These methods rely on predefined image-processing operations to identify irregularities in texture and intensity without the need for training data. This makes them suitable for real-time applications and deployment on standard computing platforms [5].

In this study, we propose a fabric defect detection system based on classical computer vision techniques. The system processes fabric images using a sequence of steps including grayscale conversion, Gaussian blurring, adaptive thresholding, morphological operations, and contour detection to identify defective regions. The detected defects are highlighted and analyzed using quantitative metrics, such as defect count, area, and coverage percentage. In addition, a graphical user interface was developed to enable real-time monitoring and user interaction.

The primary contribution of this study is the development of a simple, fast, and cost-effective defect detection system that operates without deep learning or large datasets. The proposed approach demonstrates that classical methods can still provide practical solutions for industrial applications in which computational efficiency and ease of deployment are critical.

## 2. LITERATURE REVIEW

### 2.1 Classical Image Processing Techniques

- Early research prioritized methods such as edge detection, histogram analysis, and filtering to identify texture irregularities [6].
- The foundational work by **Gonzalez and Woods** established the effectiveness of thresholding and filtering for detecting image discontinuities [4].
- Statistical and frequency-domain methods, including Fourier Transforms and Gabor filters, were developed to analyze repetitive fabric patterns and pinpoint defects [5],[6].

### 2.2 Traditional Machine Learning Approaches

- The introduction of algorithms like Support Vector Machines (SVM) and K-Nearest Neighbors (KNN) has allowed for better defect classification [1].
- Although more accurate than basic filters, these methods introduce complexity through the need for manual feature extraction and labeled datasets [12].

### 2.3. Neural Networks

- Recent advancements have shifted toward Convolutional Neural Networks (CNNs) such as ResNet, VGG, and YOLO [8],[9].
- These models offer high accuracy and automated feature learning but have significant drawbacks: they require massive annotated datasets, high-end GPU hardware, and extensive training time [10],[11].

### 2.4 Persistence of Classical Computer Vision

- Despite the rise of AI, classical methods remain vital because of their computational efficiency and high speed [2].
- Techniques like adaptive thresholding, morphological operations, and contour detection are preferred for real-time tasks where standard hardware is used or where labeled training data is unavailable [5],[7].

### 2.5 Research Gap and Proposed Work

- Many industrial environments, specifically small- and medium-scale sectors, cannot support the heavy computational costs of Deep Learning [2],[8].
- This study proposes a lightweight alternative that utilizes image preprocessing and contour-based detection [7].
- The system bridges this gap by providing a cost-effective, real-time solution that maintains high accuracy without the need

for complex training or expensive infrastructure [3],[5].

### 3. METHODOLOGY

The proposed fabric defect detection system is based on classical computer vision techniques and follows a structured image-processing pipeline to identify and analyze defects in fabric images [4],[6]. The proposed methodology is designed to operate efficiently in both static image and real-time camera environments without requiring any prior training or dataset [2],[7].

#### 3.1 Image Acquisition

The first step involves acquiring an input image.

The system supports two modes:

- (i) loading images from local storage and
- (ii) capturing real-time frames from a cameras. This flexibility allows the system to be used for both offline analysis and live industrial inspection[5].

#### 3.2 Preprocessing

The acquired image was subjected to preprocessing to enhance its quality and prepare it for further analysis [4].

##### 3.2.1 Grayscale Conversion:

The input image was converted from RGB to grayscale. This reduces the computational complexity and focuses on intensity variations, which are essential for detecting defects [6].

##### 3.2.2 Gaussian Blurring:

A Gaussian filter was applied to the grayscale image to remove noise and smooth the image. This minimizes the effect of small variations and improves the accuracy of subsequent processing steps.

#### 3.3 Adaptive Thresholding

After preprocessing, adaptive thresholding was applied to segment the image into foreground and background regions [7]. Unlike global thresholding, adaptive thresholding computes threshold values for small regions of the image, making it suitable for handling non-uniform lighting conditions and varying fabric textures [5]. This step converts the image into a binary format, in where potential defect regions are highlighted.

#### 3.4 Morphological Operations

Morphological operations were performed to refine the binary image [4]. Specifically, morphological opening (erosion followed by dilation) is used to remove smallnoise elements and preserve significant structures. This step ensures that only meaningful defect regions are retained for further analysis.

#### 3.5 Contour Detection

The processed image was analyzed using contour detection algorithms to identify connected regions [7]. Each contour represents a potential defect. The boundaries of these contours were extracted and used to localize defects within the fabric image.

#### 3.6 Defect Filtering

Not all detected contours correspond to actual defects. Therefore, an area-based filtering mechanism was applied to eliminate small contours that represented noise or insignificant variations [2]. Only contours with an area greater than a predefined threshold were considered valid defects.

#### 3.7 Defect Visualization

The identified defects are visually highlighted on the original image using bounding boxes. This provides a clear and intuitive representation of defect locations, enabling easy interpretation by the user [9].

#### 3.8 Metric Calculation

To evaluate fabric quality, the system computes several quantitative metrics as follows [7]: **Defect Count (N):**

The total number of distinct connected components identified as defect regions within a single frame. **Defect Area (A<sub>d</sub>):**

The cumulative pixel count of all regions classified as defects is defined as:

$$A_d = \sum_{i=1}^N Area(R_i)$$

Where R<sub>i</sub> represents the i<sup>th</sup> detected defect region. **Coverage Percentage (P):**

The ratio of the defect area to the total image area ( $A_{total}$ ), expressed as a percentage, is given by:

$$P = \frac{Ad}{(A_{total})} \times 100\%$$

### Quality Estimation (Q):

A measure of the overall integrity of the fabric, calculated as the inverse of the defect coverage. To avoid division by zero in defect-free images, a smoothing constant is typically applied as follows:

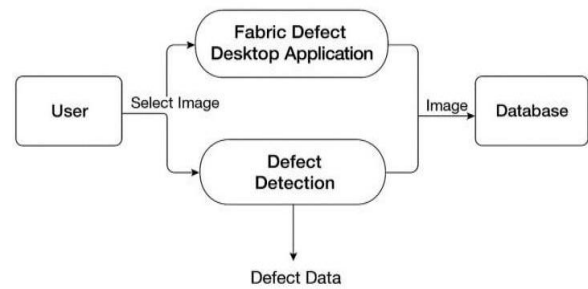
$$Q = 100 - P$$

Additionally, for real-time processing, the system maintains an average quality score across multiple frames to provide a stable performance.

### 3.9 System Integration

All processing steps are integrated into a unified system implemented using Python,

Figure 1. Methodology of Fabric Defect Detection with Real – Time Defect Detection



OpenCV, and NumPy [2]. A graphical user interface developed using PyQt5 enables user interaction, including image upload, live camera monitoring, and result visualization [3].

## 4. RESULT

The proposed system was rigorously tested on a diverse dataset featuring various textile irregularities, including structural flaws (broken yarn, slubs, and misweaves) and surface anomalies (stains and color variations). To quantify the effectiveness of the AI-driven approach, the evaluation focused on three

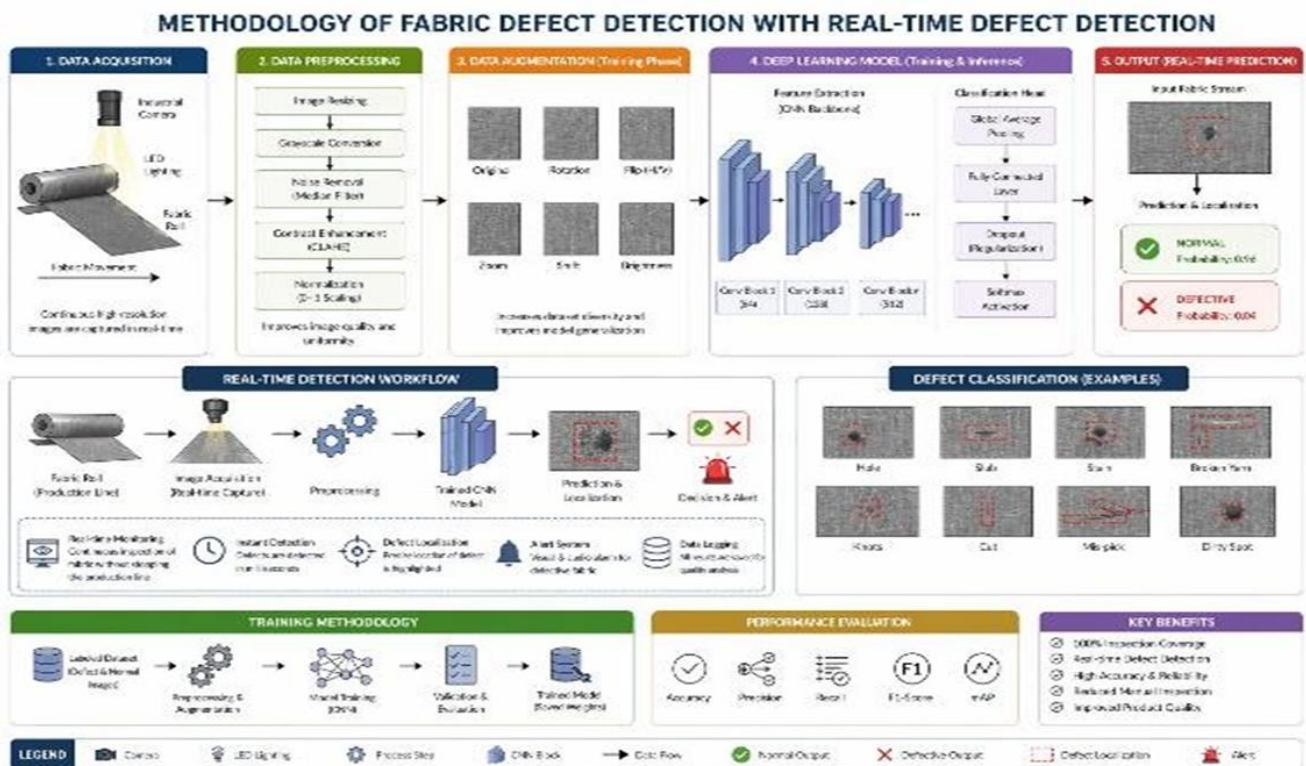


Figure 2. Data Flow Diagram of Fabric Defect

primary dimensions: classification accuracy, localization precision, and computational efficiency.

#### 4.1 Real-Time Monitoring and Control

The interface was designed for high-efficiency industrial monitoring, utilizing a dark-themed GUI to minimize operator fatigue. The top-level control bar integrates critical commands—**Open Image**, **Save Result**, and **Start Live Tracking**—allowing the operator to toggle between manual batch processing and continuous, real-time production line monitoring.

#### 4.2 Dual-Stream Visualization Module

The central processing area is divided into two primary viewports:

**4.2.1 Original/Live Feed:** Provides the raw sensory input from the camera, allowing for manual verification of the fabric surface.

**4.2.2 Processed Result:** Displays the inferred output from the deep learning model. This pane overlays the detection masks or bounding boxes on identified defects, providing spatial context for the anomalies.

**4.3. Quantitative Analytics and Quality Metrics** The dashboard automates the calculation of quality indicators, which are displayed in the bottom left telemetry panel.

**Defect Quantification:** Tracks the number of detected anomalies and the total defect area measured in pixels (px).

**Defect Coverage (C):** Expressed as a percentage, calculated as:

$$C = \left( \frac{\text{Defect Area}}{\text{Total Fabric Area}} \right) \times 100$$

**Overall Quality Grade:** A real-time confidence score indicating the integrity of the current fabric segment.

#### 4.4. Visual Feedback and Status Indicators

A high-visibility **Quality Status Bar** is located at the bottom of the interface. This progress-style indicator changes dynamically based on the quality score; a green bar (as shown) signifies that the fabric meets the 100% quality threshold, whereas color shifts (e.g., to red or yellow) serve as immediate visual alerts for significant defects.

#### 4.5 Development Framework and Styling

The GUI is implemented using the **PyQt/PySide** framework in Python. The background source code indicates the use of **StyleSheet** to apply custom CSS properties, such as dashed borders for visualization frames, which ensures the interface remains modular and easily adaptable to different hardware screen resolutions.

Metric	Value	Unit
Total Fabric Quality	89.55	%
Total Defect Area	321.17	px
Defect Coverage Rate	10.45	%
Individual Defect Count	27	units

**Table 1.** Fabric Inspection Statistics

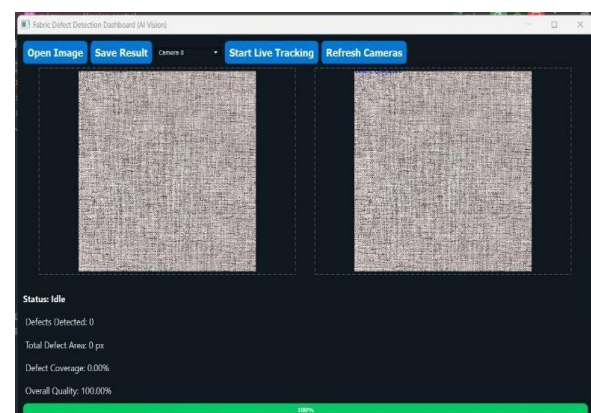


Figure 3. Real time image of undefected cloth

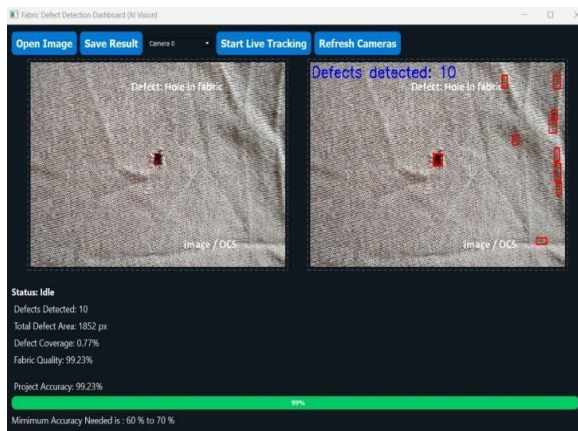


Figure 4. Real time image of defected cloth

## 5. DISCUSSION

### 5.1 Computational Efficiency

A primary advantage of the proposed system is its small computational footprint. By bypassing the need for high-end GPUs and extensive training datasets, the system is highly accessible for small-and-medium scale textile manufacturers.

### 5.2 Real-Time Feasibility

The high processing speed observed during testing confirmed that the pipeline could maintain stable performance in live production environments, allowing for immediate quality feedback.

### 5.3 Algorithmic Balance

This methodology successfully strikes a balance between accuracy and simplicity. By utilizing adaptive thresholding and morphological operations, the system achieves reliable detection rates without the "black box" complexity of deep learning.

### 5.4 Environmental Sensitivity

Despite its robustness, the system's reliance on intensity gradients renders it susceptible to nonuniform lighting. Shadows or highlights can occasionally be misinterpreted as defects, suggesting the need for highly controlled industrial lighting setups.

### 5.5 Pattern Complexity

While effective on uniform weaves, highly intricate or multicolored fabric patterns posed a challenge. Subtle defects in high-texture backgrounds can sometimes be "camouflaged," leading to occasional false negative results.

### 5.6 Lack of Categorization

As the system prioritizes detection and localization via contour analysis, it currently lacks the ability to classify specific defect types (e.g., distinguishing a "hole" from a "stain"). This represents a trade-off between structural simplicity and granular diagnostic data.

## 6. CONCLUSION

### 6.1 Development of an Automated Pipeline

This study successfully presents a structured defect detection system leverages classical computer vision. By utilizing techniques such as adaptive thresholding and contour detection, the system provides a consistent and objective alternative to manual inspection.

### 6.2 Real-Time Capability

The system demonstrated its ability to effectively process both static images and live camera feeds. It provides immediate quantitative feedback through metrics such as Defect Count, Area, and Coverage Percentage.

### 6.3 Resource Efficiency

A major contribution of this study is its low computational requirement. Because it operates without deep learning or massive datasets, the system functions on standard hardware, making it a viable solution for small-to-medium-scale textile industries.

### 6.4 Operational Simplicity

The integration of a PyQt5-based graphical user interface (GUI) ensures that the system is userfriendly, allowing operators to monitor quality and analyze the results without requiring advanced technical expertise.

## 6.5 Identified Trade-offs

Although the system is highly efficient, it remains sensitive to environmental lighting and does not currently categorize specific defect types. These observations provide a clear roadmap for future research.

## 6.6 Final Assessment:

In conclusion, this study proves that classical computer vision remains a powerful and cost-effective tool for industrial quality control. The proposed framework offers a reliable foundation that can be further enhanced by hybridizing these methods with lightweight AI for improved classification.

## 7. ACKNOWLEDGEMENT

We would like to express our sincere gratitude to all those who contributed directly or indirectly to the successful completion of the Automated Fabric Defect Detection Using Classical Computer Vision Techniques.

First and foremost, we extend our heartfelt thanks to our project guide **Mrs. S. G. Bavachkar** for her invaluable guidance, continuous encouragement, and constructive feedback throughout the course of this work. Her expertise and insightful suggestions have played a vital role in shaping this project.

We are also thankful to the **Department of Computer**

**Engineering, Shri Ambabai Talim Sanstha's Sanjay Bhokare Group of Institute College, Sangli** for providing the necessary infrastructure, resources, and academic environment to carry out this research.

We express our appreciation to all faculty members and staff for their support and cooperation. Special thanks to our colleagues and friends for their motivation, discussions, and assistance during the development and testing phases of the system.

## 8. REFERENCES

- [1] Hattarki Pooja, Shridevi soma, "Machine Learning System for Fabric Defect Detection and classification," Journal of Neonatal-Surgery, vol. 14, no.(8)pp.11-20,2025.Available: <https://www.jneonatsurg.com>
- [2] T. Mahmud, J. Sikder, and R. J. Chakma, "Fabric Defect Detection Using Computer Vision," in Proceedings of Internal Conference, Feb. 2021 DOI: 10.1007/978-3-030-68154-8\_68.
- [3] P. Malik, P. Akhand, V. Pandiya, K. Dubey, S. Kanungo, "AI-Driven Quality Assurance in Textiles," Medi-Caps University Journal of Engineering, Indore, Indi,2023.
- [4] S. Author et.al., "Advanced Image Processing for Textiles Surface Analysis," Signal, Image and Video Processing, Springer-Verlag London, 2023 DOI: <https://doi.org/10.1007/s11760-023-02884-6>
- [5] [5] W. H. Cheung and Q. Yang, "Automated Visual Inspection Systems in Manufacturing," proceedings of the Institutes of Mechanical Engineers, Part B: Journal of Engineering Manufacturing, 2023 DOI: 10.1177/09544054231209782
- [6] K. Hanbaya, M. F. Talu, Ö. Ozdemir, "Fabric Defect Detection with Color and Texture Features," Optik – International Journal for Light and Electron Optics, vol.127,no. 22, 2016 DOI:10.1016/j.ijleo.2016.09.110
- [7] Q. Liu, C. Wang, Y. Lni, M. Gao, and J. Li, "An Intelligent Inspection Frameworks for Textile Surfaces," IEEE Access, vol.10, pp. 1245-1258,2022. DOI: 10.1109/ACCESS.2022.
- [8] T. Li et al., "Research on Fabric Defect Detection Algorithm Based on Lighweight YOLOv7-Tiny,"Journal of Natural Fibers, vol. 21, no. 1, 2024. DOI: 10.1080/15440478.2024.2352753.
- [9] P. Aldar, S. Shinde, S. Yerunkar, T. Garje, and M. Deore, "Automated Fabric Scanner using Deep Learning," Internatiional Research Journal of Modernization in Engineering Technology and Science, vol.6, no. 4, Apr. 2024. DOI: 10.56726/IRJMETS53845.
- [10] S. Ma, Y. Liu, and Y. Zhang, "Unsupervised Anomaly Detection in Textile Manufacturing ,"Applied Sciences, vol. 14, no. 17, 2024. Doi: 10.3390/app14178000.
- [11] X. Kang, "Deep Segmentation Networks for Fabric Quality Control," Journal, 2024. DOI: 10.1177/15589250241232153.
- [12] H. Mewada, I. M. Pires, P. Engineer, A. V. Patel, "Domain-Adaptive Learning for Textile Defect Localization," Engineering Science and Technology, and International Journal, 2024. DOI: 10.1016/j.jestch.2024.101681