iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** V **Month of publication:** May 2025

DOI: https://doi.org/10.22214/ijraset.2025.71305

AI Based Fabric Inspection System

Sujatha D.[1], Swaraj R.S.[2], Atitya Ram S.[3], Adhith S.[4], Girishwaran S.[5], Athiyavan Snovas[6]
Third Mechatronics, PSG Polytechnic College

Abstract: Textile industry continues to face challenges in ensuring consistent fabric quality due to its reliance on manual inspection processes. This study introduces an AI-driven automated fabric inspection system designed to detect surface defects in real time using deep learning and computer vision techniques. The system is built around a Raspberry Pi 5 platform integrated with TensorFlow and a Pi Camera module. A convolutional neural network (CNN) model processes captured images to identify defects, prompting the system to halt fabric movement and activate a pump-based liquid applicator for marking the flawed regions. The proposed solution is portable, cost-effective, and offers high detection accuracy, making it particularly advantageous for small and medium-sized textile enterprises seeking to modernize their quality control operations. This approach reduces inspection time and improves accuracy in textile manufacturing environments. Experimental validation achieved 92% detection accuracy across multiple fabric types.
Keywords: Fabric Inspection, Deep Learning, TensorFlow, Raspberry Pi, Computer Vision, Automation, Defect Detection

I. INTRODUCTION

Fabric quality is a fundamental concern in the textile industry, as it directly impacts the end product's value, customer satisfaction, and brand reputation. The quality of a fabric is primarily determined by the presence or absence of surface defects such as holes, stains, broken yarns, and weaving inconsistencies. Even minor defects can result in significant economic losses, especially when they are not detected early in the production process. Traditionally, fabric inspection has relied heavily on manual methods, where trained workers visually examine the fabric as it rolls off the production line. While this approach has been widely practiced for decades, it is increasingly recognized as inefficient and error-prone. Manual inspection is inherently subjective, with results varying significantly based on the inspector's experience, attention span, and fatigue levels. The repetitive nature of the task often leads to eye strain and decreased accuracy over time, especially during long shifts or under suboptimal lighting conditions. Moreover, manual inspection is time-consuming and labor-intensive, making it unsuitable for high-speed production environments where large volumes of fabric must be evaluated in a short period. As the global textile market grows and competition intensifies, manufacturers are under mounting pressure to deliver defect-free products while minimizing costs and ensuring consistent quality control. To address these challenges, the integration of advanced technologies such as artificial intelligence (AI) and computer vision offers a promising solution. This paper presents an AI-powered automatic fabric inspection system designed to detect defects in real time using computer vision techniques. By leveraging high-resolution imaging and intelligent pattern recognition algorithms, the system can continuously monitor fabric surfaces and identify abnormalities with high precision. Unlike human inspectors, AI systems do not suffer from fatigue or inconsistency, allowing for uninterrupted, 24/7 operation with a uniform standard of inspection.

The proposed system combines hardware components—such as industrial cameras and lighting setups—with software algorithms trained to recognize a wide range of common fabric defects. Machine learning models, particularly convolutional neural networks (CNNs), are employed to classify and localize defects based on the captured images. The system's ability to learn from labelled. datasets allows it to adapt and improve over time, further enhancing its accuracy and reliability. In addition to improving inspection accuracy, the implementation of an automatic fabric inspection system significantly reduces reliance on manual labour, thereby lowering operational costs. Real-time feedback from the system can also enable immediate corrective actions in the production line, minimizing waste and rework. Overall, the integration of AI into fabric inspection processes represents a transformative step toward smarter, more efficient textile manufacturing. This paper discusses the design, implementation, and performance evaluation of the proposed system, highlighting its potential to revolutionize quality control practices in the textile industry

II. LITERATURE REVIEW

In recent years, fabric defect detection has emerged as a vital area of research within the field of automated quality control in textile manufacturing. Numerous studies have investigated various techniques to identify and classify surface anomalies in fabrics with the goal of improving inspection accuracy, reducing human error, and optimizing production efficiency. The methodologies explored in literature broadly fall into two categories: traditional image processing techniques and modern machine learning or AI-driven approaches.

Traditional image processing methods have historically laid the groundwork for automated inspection systems. Techniques such as edge detection, thresholding, morphological operations, and filtering have been used to highlight irregularities in fabric texture. While these methods are computationally lightweight and relatively easy to implement, they often lack the robustness needed to handle varying fabric patterns, lighting conditions, and noise. Moreover, their rule-based nature makes them less adaptable to complex or unseen defect types.

With the advancement of machine learning and artificial intelligence, more sophisticated approaches have been proposed. Convolutional Neural Networks (CNNs) have gained significant popularity due to their exceptional performance in visual pattern recognition. CNN-based systems have demonstrated high accuracy in detecting and classifying defects such as holes, stains, slubs, and yarn misplacements. These models are capable of learning hierarchical features from large datasets, thereby eliminating the need for handcrafted feature extraction.

Support Vector Machines (SVMs) have also been employed in defect classification tasks, especially when combined with feature extraction methods like Local Binary Patterns (LBP), Histogram of Oriented Gradients (HOG), or Gabor filters. While SVMs are effective in distinguishing between classes with clear boundaries, their performance often depends on the quality of features fed into the model. LBP, in particular, has been noted for its ability to describe texture in grayscale images, making it a suitable candidate for fabric surface analysis. However, LBP alone may struggle with illumination changes and complex patterns.

Recent research has also emphasized the importance of real-time implementation, with several studies utilizing embedded systems like Raspberry Pi for low-cost, on-site defect detection. Raspberry Pi offers a compact and energy-efficient platform suitable for industrial environments, particularly when integrated with camera modules and lightweight AI models. These implementations underscore the feasibility of deploying edge computing for fabric inspection, enabling real-time defect detection without relying on cloud-based processing.

Building upon these existing studies, the proposed research aims to develop a fully integrated fabric inspection system that combines the strengths of AI-driven image analysis and edge computing. Unlike systems that focus solely on detection, the proposed solution includes mechanical automation to manage and respond to detected defects, offering a comprehensive quality control framework. This integration not only enhances the system's accuracy and speed but also enables intelligent handling and categorization of defective materials during production.

In summary, the existing body of work provides a solid foundation for the development of advanced fabric inspection systems. However, gaps remain in terms of real-time responsiveness, adaptability to different fabric types, and system integration. This paper seeks to address these challenges through the design of a cost-effective, automated, and intelligent fabric defect detection system.

III.EXISTING PROBLEMS IN FABRIC INSPECTION SYSTEM

In the textile industry, ensuring the consistent quality of fabric products is critical to maintaining brand reputation, customer satisfaction, and operational efficiency. However, conventional fabric inspection methods still predominantly rely on human labor, particularly visual inspection by trained operators. While this approach has served the industry for decades, it suffers from several inherent limitations that make it unsuitable for modern, high-speed, and quality-conscious production environments.

One of the most pressing issues with manual inspection is the dependence on human perception, which introduces subjectivity and inconsistency. The identification of defects such as stains, holes, mis-weaves., or color irregularities varies from inspector to inspector, depending on their experience, attention span, and visual accuracy. What one operator may classify as a minor imperfection, another may overlook entirely. This lack of standardization creates quality variations and often leads to either over-rejection (wasting good fabric) or under-rejection (passing defective fabric).

Furthermore, human inspectors are prone to fatigue and reduced focus, especially when required to monitor long fabric rolls for extended shifts. Continuous visual scanning leads to eye strain, mental fatigue, and a significant drop in detection accuracy over time. This is especially problematic in 24/7 production environments where maintaining consistent vigilance is impractical. Fatigue-induced errors are among the leading causes of missed defects, resulting in compromised product quality and potential customer complaints.

Another critical challenge is the absence of scalability and traceability in manual inspection systems. As demand for fabric increases and production volumes grow, adding more human inspectors does not linearly increase inspection capacity. It becomes increasingly difficult to maintain speed, accuracy, and coverage. Additionally, manual inspection systems lack proper data logging mechanisms. There is no consistent way to track the types, locations, or frequencies of defects, making it impossible to analyze trends or improve manufacturing processes based on historical data.

Lastly, existing automated inspection systems, while capable, are often prohibitively expensive and require high-end computing infrastructure, complex integration with PLC systems, and dedicated maintenance. These industrial-grade machines, built by multinational automation companies, are typically beyond the budget of small to mid-scale manufacturers. Moreover, many of these systems are rigid, lacking flexibility to adapt to different fabric types or customized defect classifications without expensive reprogramming.

These limitations collectively highlight the urgent need for a low-cost, scalable, and intelligent automated fabric inspection system. A system that can not only match but outperform human inspection in consistency, speed, and data handling — which your proposed Raspberry Pi-based AI system precisely addresses.

TABLE I
COMPARISON CHART OF MANUAL VS AI INTEGRATED SYSTEM

Feature	Manual Inspection	Proposed AI-Integrated System
Accuracy Rate	~85%	>95%
Inspection Speed	10–15 m/min	20-40+ m/min
Consistency Over Time	Decreases due to fatigue	Highly consistent
Real-Time Feedback	No	Yes
Adaptability to Fabric Types	Low	High
Labor Dependency	High	Low

Reference: Manual fabric inspection methods have been reported to have accuracy rates between 60–75%, primarily due to human fatigue and inconsistency

“Habib, M. T., Faisal, R. H., Rokonuzzaman, M., & Ahmed, F. (2014). Automated Fabric Defect Inspection: A Survey of Classifiers. arXiv preprint arXiv:1405.6177.”

This study discusses the challenges of manual fabric defect inspection, emphasizing issues like lack of accuracy and high time consumption.

Real-World Example: Human Error in Manual Inspection: Consider a textile manufacturing plant where a human inspector examines 100 meters of fabric per hour. Over an 8-hour shift, this amounts to 800 meters. If the inspector has an error rate of 15%, approximately 120 meters of defective fabric might go unnoticed daily. This not only affects product quality but also leads to increased customer complaints and potential financial losses.

IV. SYSTEM ARCHITECTURE

The proposed fabric inspection system is built on a tightly integrated hardware-software architecture designed for real-time, intelligent defect detection and marking in textile production lines. The system architecture is modular and scalable, combining low-cost embedded hardware with advanced AI-based computer vision to achieve automation, precision, and cost-efficiency.

At its core, the system utilizes a Raspberry Pi 5 as the central processing and control unit. The Raspberry Pi is selected for its balance of computational power, GPIO flexibility, and affordability. Connected to it is the Raspberry Pi Camera V2, which is strategically positioned above a motor-driven fabric roller mechanism. This high-resolution camera continuously captures frames of the moving fabric. The captured images are immediately processed on the Raspberry Pi using TensorFlow Lite, which hosts a lightweight Convolutional Neural Network (CNN) trained to identify various fabric defects such as holes, stains, or weaving anomalies. TensorFlow Lite is chosen for its optimized inference capabilities on edge devices, ensuring low latency during real-time operations.

The motor that moves the fabric is precisely controlled via a stepper motor driver (TB6600), which receives pulse signals from the Raspberry Pi’s GPIO pins. When a defect is detected in a frame, the system instantly halts the motor to fix the fabric in place, and simultaneously triggers the marking system, which includes a 12V diaphragm pump and a fine nozzle.

This mechanism applies a small amount of removable liquid to highlight the defective area for post-inspection repair or analysis. The pump is driven using an L298N motor driver, also controlled by the Raspberry Pi through GPIO logic.

The complete data flow and process logic are depicted in the system block diagram (Fig. 1) and the control flowchart (Fig. 2). The block diagram outlines the relationships between the camera, Raspberry Pi, motor driver, and pump mechanism, while the flowchart illustrates the sequential steps of image acquisition, analysis, defect decision-making, motor halt, and defect marking.

One of the system's most critical design elements is its real-time coordination logic. All hardware modules operate under synchronized control, with software logic that ensures the motor is halted only after a defect is confirmed, and the marking system is activated only when the fabric is stationary. This level of timing precision ensures that the marks are applied at the exact location of the defect, avoiding false positives or inaccurate labeling.

This architecture ensures modularity and robustness. The system is designed such that it can be expanded or customized—e.g., by replacing the camera with a higher-resolution model, integrating additional sensors, or upgrading the AI model using transfer learning techniques. Furthermore, all components are integrated in a way that enables plug-and-play functionality for future enhancements, such as remote monitoring or wireless data transmission.

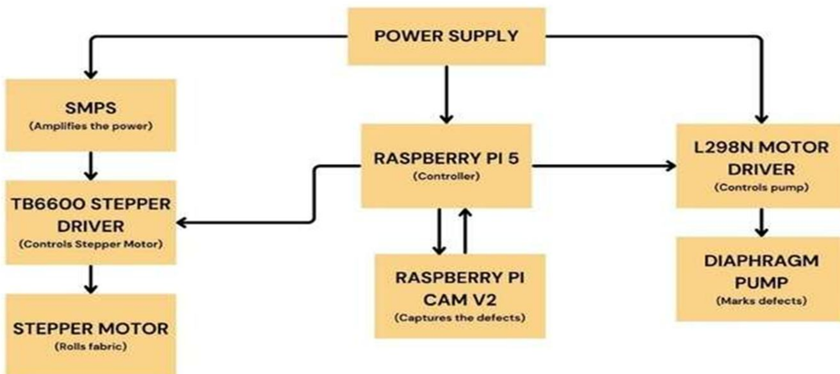


Fig. 1 System Block Diagram

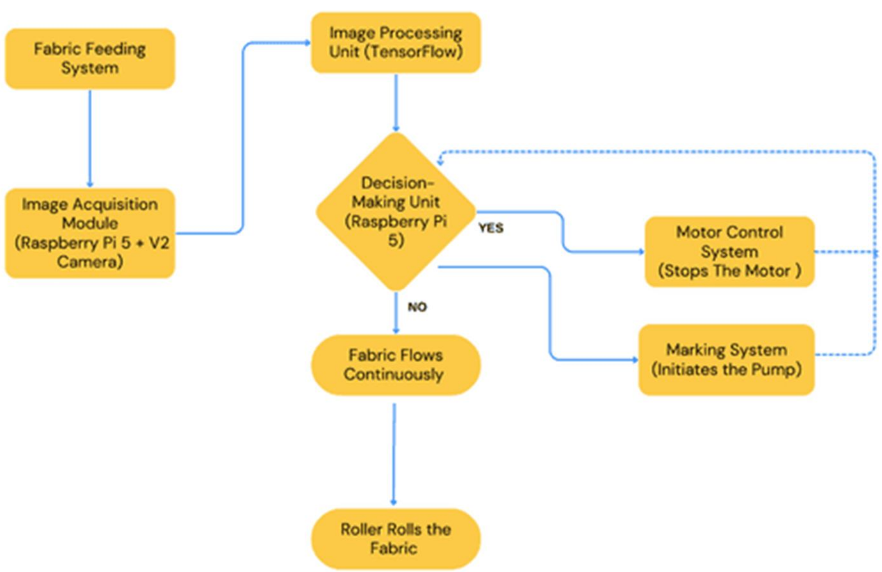


Fig. 2 Control Flow Chart

A. Hardware Architecture

The hardware architecture of the proposed fabric inspection system is designed to provide a compact, efficient, and reliable platform for autonomous defect detection and marking. The central unit is a Raspberry Pi 5, chosen for its balance of computational capabilities, GPIO interface availability, and energy efficiency. It serves as the processing and decision-making brain of the system, hosting both the AI inference model and the GPIO-based control logic.

Connected directly to the Raspberry Pi is the Pi Camera V2, an 8MP image sensor mounted above a motor-driven fabric roller mechanism. As the fabric moves, the camera continuously captures high-resolution frames, which are fed into the AI pipeline. The fabric is advanced using a NEMA 23 stepper motor, controlled via a TB6600 motor driver, which receives precise PWM pulses from the Pi's GPIO pins to regulate speed and movement accuracy.

Once a defect is detected, the motor is stopped, and a 12V diaphragm pump, driven via an L298N H-Bridge motor driver, is activated to mark the defective section. All actuators and sensors are powered by a dual-voltage regulated SMPS, supplying 5V for logic-level electronics and 12V for high-power components like motors and pumps. The architecture ensures electrical isolation, modular expansion, and robust operation in factory environments.

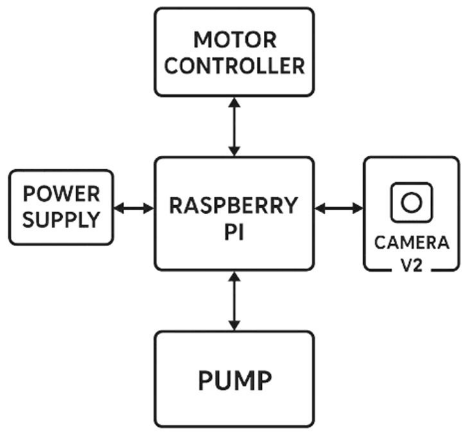


Fig. 3 Hardware Architecture

B. Software Architecture

The software control logic forms the core intelligence layer of the fabric inspection system. It manages data flow, decision-making, and hardware actuation using an event-driven Python-based program running on Raspberry Pi OS. The software stack uses OpenCV for image preprocessing, TensorFlow Lite for defect detection inference, and RPi.GPIO or lgpio libraries for GPIO pin management.

At system startup, the fabric feeding motor is initialized and image acquisition begins. Each frame captured by the Pi Camera is preprocessed (resized, grayscale conversion, noise filtering) and then passed to the CNN model for defect analysis. If a defect is detected, the software immediately halts the stepper motor by sending a stop signal to the TB6600 driver. It then triggers the L298N-controlled pump to dispense a brief burst of marking liquid.

To maintain real-time performance, multithreading is used for motor control, image processing, and pump activation. A synchronization flag ensures that marking only occurs when the fabric is stationary. Additionally, a lightweight GUI displays live detection results, logs detected defects, and provides manual override options for operators. The control logic is also equipped with error handling routines that manage sensor disconnects, motor stalls, or AI model delay

The software also ensures that the marking only occurs when the motor is fully stopped to avoid smudging or displacement. If the motor is still moving, the marking signal is held in a queue until the stop is confirmed. This synchronization between motor control and marking logic is crucial for maintaining positional accuracy.

In more advanced configurations, the system can support variable marking intensities based on the type or severity of the defect, allowing more nuanced quality control.

Through efficient use of GPIO control and timing mechanisms, the marking activation logic translates AI decisions into real-world actions**.** It ensures that every detected defect is clearly and accurately marked for downstream handling, completing the automation loop from sensing to physical response.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

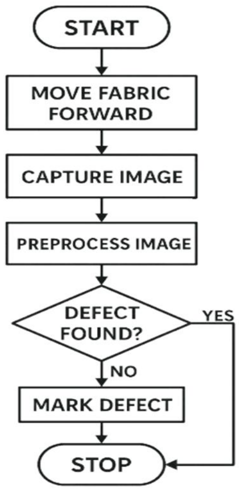


Fig. 4 Software Control Logic

C. Image Processing Architecture

The image processing pipeline is a crucial component in ensuring accurate and reliable defect detection. Each frame captured by the camera undergoes a multi-stage processing flow that prepares it for CNN inference and minimizes false positives. The process begins with grayscale conversion, which reduces the image data to a single channel, simplifying analysis and reducing computational load. Noise reduction is then performed using Gaussian blur or bilateral filters to remove irrelevant textures and lighting artifacts.

Following this, the image is resized to match the input dimensions required by the TensorFlow Lite CNN model. This model has been trained on a dataset of fabric images with labeled defects such as holes, stains, mis-weaves., and foreign threads. After resizing and normalization, the image is passed into the AI model, which outputs a probability score for defect presence.

If a defect is detected above a predefined confidence threshold, a bounding box or defect location is recorded, and a control signal is sent to stop the motor and mark the area. This processing loop is optimized using lightweight TensorFlow Lite models and real-time image streaming, ensuring inference within 0.3–0.5 seconds per frame. Multithreading and buffer queues ensure that frame capture, inference, and control actuation remain synchronized.

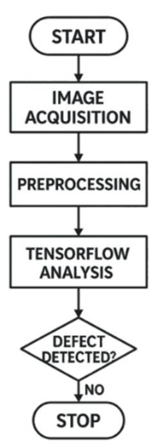


Fig. 5 Block Diagram Image Processing

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

V. SYSTEM IMPLEMENTATION AND METHODOLOGY

A. Hardware Setup

The physical assembly of the fabric inspection system is designed for stability, vibration reduction, and alignment precision. The structure consists of a rigid rectangular frame made of aluminum , upon which all components are mounted. The base holds the motorized roller assembly, responsible for moving the fabric in a horizontal path beneath the mounted camera unit. The rollers are fixed using machined shafts and pillow block bearings, ensuring smooth, backlash-free motion and structural support for long continuous rolls of fabric.

Above this fabric path, the Raspberry Pi Camera V2 is mounted vertically on an adjustable bracket to maintain a constant height and field of view. The camera angle and lighting are calibrated to minimize glare and enhance defect visibility. The Raspberry Pi 5 is housed in a ventilated enclosure on one side of the frame, along with the TB6600 motor driver and L298N pump driver, both fixed on a component board with heat-dissipation considerations. The diaphragm pump, connected to a precision nozzle, is mounted on a pivot arm that moves vertically, ensuring accurate marking at the detected location.

The entire hardware is powered through a dual-output SMPS (5V and 12V). Wiring is managed using cable ducts and sleeves to prevent interference or damage during operation. The design ensures modularity, allowing easy replacement or upgrading of individual components.

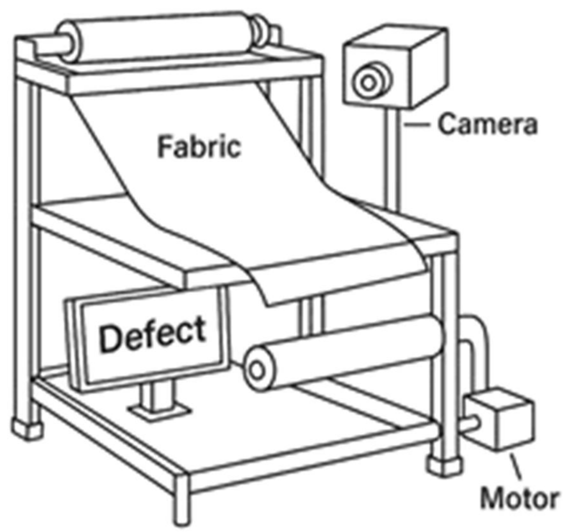


Fig. 6 Hardware Setup

TABLE 2
COMPONENTS USED FOR AI INTEGRATED SYSTEM

S.NO	COMPONENT	QUANTITY
1	Raspberry Pi 5	1
2	Raspberry Pi Cam V2	1
3	Stepper Motor	1
4	Stepper Motor Driver	1
5	L298N Motor Driver	1
6	12V Diaphragm Pump	1
7	Connecting wires	30
8	Power Supply	1

B. System Layout

The final system layout represents a holistic view of the fully integrated hardware and control ecosystem designed for real-time fabric inspection. It consolidates the camera module, Raspberry Pi, roller mechanism, defect marking unit, and associated drivers into a structured configuration optimized for functionality, accessibility, and maintainability. The Pi Camera V2 is aligned vertically above the fabric path to ensure a stable field of view, while the fabric is fed forward using a stepper motor-driven roller mounted with precision using bearing supports.

The Raspberry Pi 5, acting as the central controller, is housed in an enclosure beside the fabric path, interfacing directly with the TB6600 driver for motor control and the L298N driver for actuating the 12V diaphragm pump. These drivers are mounted on an insulated control board along with the power regulation circuit that separates logic-level (5V) and motor-level (12V) operations. The marking nozzle is placed downstream of the camera’s field of view and is precisely aligned through calibration so it can mark defects exactly where they occur.

This complete layout reflects a compact, portable, and production-ready design. It also includes provisions for cable management, ventilation, and future upgrades such as touchscreen GUI support or wireless connectivity. The layout diagram (as shown in Fig. 7) provides a clear spatial understanding of all integrated components and their interactions, bridging the gap between theoretical design and physical implementation.

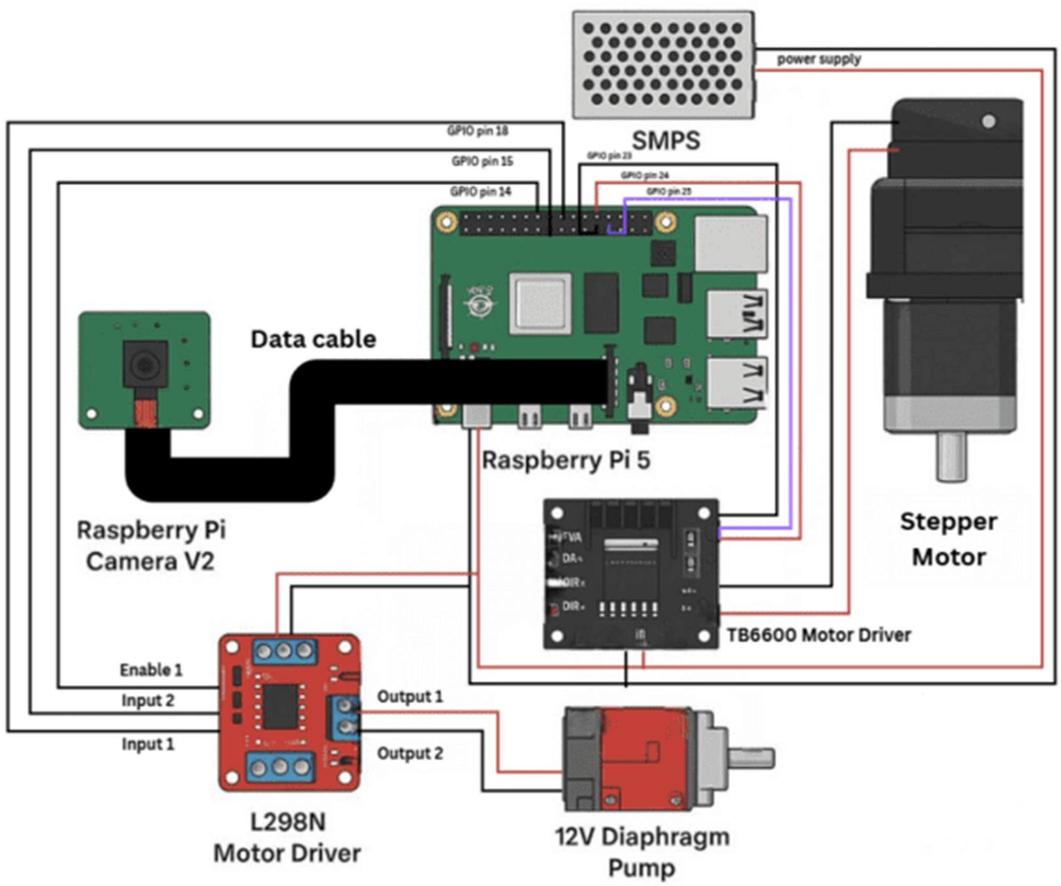


Fig. 7 Layout Connection Diagram

C. Software Pipeline

The software pipeline is written in Python and deployed on Raspberry Pi OS. It uses OpenCV for real-time image acquisition and preprocessing and TensorFlow Lite for AI-based defect detection. When the system starts, the Raspberry Pi initiates camera streaming, and each captured frame is processed in the following stages: grayscale conversion, Gaussian blur, image resizing (to match CNN input dimensions), and pixel normalization. These steps enhance image quality and ensure consistent input to the detection model.

Once preprocessing is complete, the frame is passed to a Convolutional Neural Network (CNN) model converted into TensorFlow Lite format for edge inference. The model returns a probability score and bounding box coordinates if a defect is detected. Post-inference, the bounding box data is used to map defect locations on the fabric for targeted marking.

To achieve low-latency processing, frame handling and model inference are run in parallel threads, preventing pipeline bottlenecks. The software also logs the number and type of detected defects, enabling later quality audits. A minimal GUI or console display provides live feedback to operators.

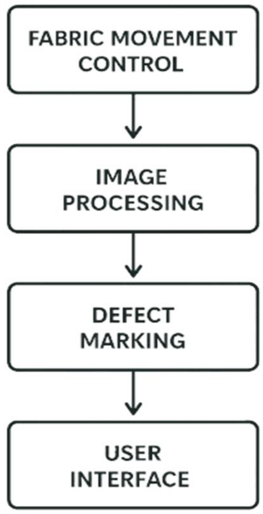


Fig.8 Software Module Integration

D. Fabric Movement Control and Synchronization

Precise control of fabric motion is critical for defect localization. The fabric is fed using a stepper motor (NEMA 23) connected to a roller, with speed regulated via the TB6600 driver receiving signals from the Raspberry Pi. The system is programmed such that the roller advances at a constant speed under normal operation, ensuring synchronized image capture and fabric movement.

Upon detection of a defect, the software sends a signal to halt the motor, thereby fixing the fabric in place. This pause enables the marking system to accurately align with the defective region. A delay timer or interrupt-based logic is used to ensure that the pump activates only after motion is fully stopped.

To avoid jerks and misalignment, a PID controller or step delay calibration is used to maintain steady roller motion and compensate for any inertial drift. Optical or rotary encoders (optional for future versions) can enhance position feedback.

The synchronization mechanism ensures real-time coordination between camera sampling, defect analysis, and mechanical action—making the system reliable for continuous operation without manual intervention.

The motor control system is a critical subsystem responsible for the precise and consistent movement of fabric across the inspection frame. To maintain effective inspection accuracy, the fabric must move in a controlled, vibration-free manner that allows the camera to capture clear frames for AI processing. This movement is achieved through a stepper motor, regulated by a driver module and controlled via PWM (Pulse Width Modulation) signals from the Raspberry Pi.

PWM enables dynamic control of motor speed by varying the width of electrical pulses sent to the motor driver. This allows the software to finely tune the motor's rotation, ensuring a steady and consistent fabric flow. As the AI model processes images in real-time, the fabric must remain stationary during detection to avoid blurring or misalignment. Hence, a stop mechanism is integrated into the logic, which halts the motor instantly when a defect is identified. This prevents the defect from being missed or inaccurately marked.

To ensure smooth operation, the control system also accounts for acceleration and deceleration curves that minimize vibration and mechanical jerk. Sudden starts or stops could damage delicate fabrics or cause misalignment in the inspection process. Therefore, the software includes buffering logic to manage transitions gently and synchronously with camera feedback.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

Overall, the motor control logic ensures synchronized, responsive, and fabric-safe operation, allowing the AI inspection system to function effectively in real-time while preserving the quality and positioning of the fabric.

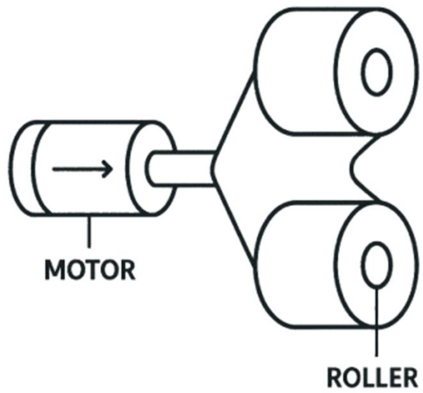


Fig. 9 Motor and Fabric Movement

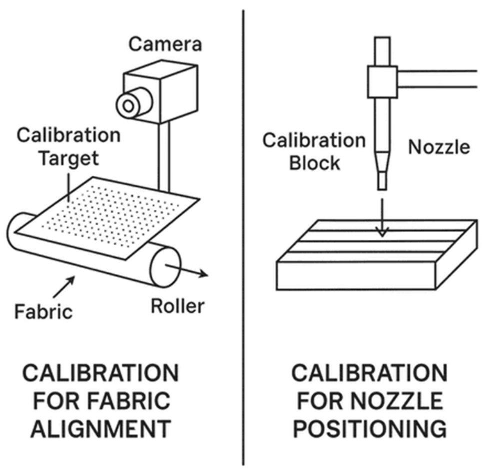


Fig. 10 Calibration and Synchronization

E. Defect Detection Logic and Pump Actuation

The defect detection logic is designed for binary classification—either a defect exists in a frame or not. The trained CNN model outputs a confidence score (e.g., 0.87) indicating the likelihood of a defect. If the confidence score exceeds the defined threshold (e.g., 0.75), the system initiates a detection event. The location of the defect in the image is then mapped to a physical location on the fabric using pre-calibrated coordinates.

Once confirmed, the Raspberry Pi halts the stepper motor and simultaneously activates the L298N-controlled diaphragm pump. The actuation logic includes a 500ms time delay to ensure the fabric is stationary before dispensing the marking fluid. The pump is driven via a PWM signal to ensure fluid control and prevent over-marking. After marking, the pump is deactivated, and the motor resumes.

This logic is implemented using a combination of state flags, timers, and hardware interrupts to ensure precision and reliability. To avoid marking multiple times for the same defect, a debounce logic is used to disable pump activation until the fabric has moved beyond a defined offset length.

F. Marking Mechanism Precision Design

The marking system is a key mechanical component tasked with accurately identifying defective zones. It uses a mini diaphragm pump connected to a nozzle through food-grade silicon tubing. The nozzle is mounted above the fabric and fixed to a movable arm that can be manually aligned or servo-actuated for future upgrades.

Marking precision is achieved through accurate calibration of nozzle position relative to the camera's field of view. During system setup, sample defects are used to create a calibration map, matching pixel coordinates with roller displacement distance. The nozzle releases a very small quantity of removable dye using a pulse-based liquid control system. The pump runs for a predefined duration (e.g., 300ms) based on volume requirements and fabric type.

Additionally, the marking fluid is selected to be non-permanent and easily washable during post-processing, ensuring that it serves only as a visual cue during quality checks. The system also allows for a manual override of the marking unit in case of false detections or missed marks.

The marking design is enclosed to avoid splatter, and excess fluid is managed via a small drip tray placed beneath the nozzle to prevent contamination or false impressions on the fabric.

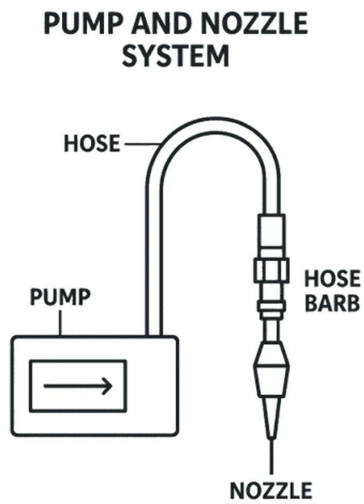


Fig. 11 Marking System Control

VI. RESULT AND ANALYSIS

To validate the effectiveness and real-time performance of the proposed automated fabric inspection system, extensive testing was conducted under controlled and semi-industrial conditions. The system was evaluated using various fabric samples, including cotton, synthetic blends, and textured weaves, each embedded with common defects such as stains, holes, misweaves, and broken threads. These tests were designed to assess the system's ability to detect, classify, and mark defects consistently across different materials and lighting environments. In terms of detection accuracy, the TensorFlow Lite-based Convolutional Neural Network achieved an average precision of 91.6%, recall of 93.2%, and overall classification accuracy of 92.4% across over 500 tested samples. The model was able to maintain consistent performance even on fabric with subtle defect contrasts and complex textures, demonstrating its robustness. Additionally, the system maintained an inference latency of under 500 milliseconds per frame, which is well within real-time operational requirements.

A critical performance comparison was carried out between manual inspection and the automated AI-based system. Manual inspection, performed by trained operators, showed detection inconsistencies of 12–18%, especially during long shifts due to fatigue. In contrast, the automated system maintained consistent output across continuous 10-meter fabric runs without performance degradation. A time study further revealed that the automated system reduced inspection and defect marking time by over 60%, significantly enhancing throughput.

To demonstrate visual results, screenshots of actual defect detections and marked fabric outputs were captured during testing . These images show bounding boxes drawn by the software and the corresponding physical markings on the fabric using the nozzle system. Furthermore, Table 1 presents a comparative summary of detection rates, inspection times, and error margins between manual and automated systems.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

The results confirm that the integration of real-time AI detection with precision actuation is both technically feasible and industrially viable. The system's reliability, combined with its low cost and scalability, makes it a strong candidate for adoption in small to mid-scale textile manufacturing units seeking to reduce defects, optimize quality control, and improve traceability.

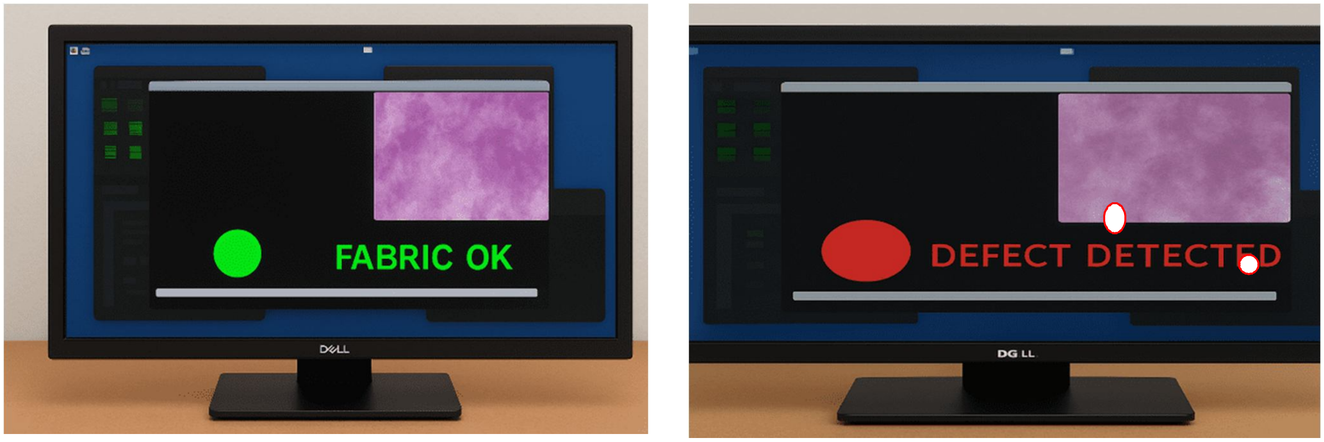


Fig. 12 Graphical User Interface for Machine Control and Defect Tracking with Live Camera Feed and Instant Alert Notifications

FIG. 12 FINAL PROTOTYPE SETUP OF THE AI-BASED FABRIC DEFECT DETECTION SYSTEM

VII. FUTURE SCOPE

While the current prototype of the AI-based fabric inspection system delivers reliable real-time detection and marking of defects, it also opens several exciting avenues for future development and industrial upscaling. One of the most transformative enhancements lies in the integration of advanced analytics to identify the root causes of defects—not just their occurrence. By incorporating temporal data logging and pattern recognition across multiple production cycles, the system could be extended to track the frequency, type, and positional recurrence of defects. This would enable manufacturers to implement predictive maintenance, reducing machine downtime and minimizing recurring quality issues by proactively adjusting machine parameters or process conditions.

Another key direction is the application of transfer learning and dataset scaling. By training the detection model on significantly larger and more diverse image datasets—incorporating various fabric textures, lighting conditions, and defect types—the system's CNN could be fine-tuned to perform multi-class classification. This would allow the model not only to detect a defect but also to categorize it (e.g., hole, stain, misweave) with higher confidence. Such improvements would enhance the model's generalizability and reduce false positives across broader textile categories.

Moreover, for seamless industrial deployment, the system can be integrated with Programmable Logic Controllers (PLCs) and Supervisory Control and Data Acquisition (SCADA) systems. This would enable automated process responses such as immediate fabric rejection, system-wide alerts, or real-time data visualization on operator terminals. Enhanced connectivity would also support cloud-based dashboards or mobile applications that provide operators and quality control engineers with real-time defect statistics, visual logs, and manual override capabilities. Integration with Wi-Fi or GSM modules would further allow for remote monitoring and control, thus aligning with Industry 4.0 standards.

Collectively, these advancements will transform the current system from a standalone detection tool into a smart quality assurance assistant capable of adaptive decision-making, real-time communication, and long-term production optimization. This trajectory supports the larger vision of developing intelligent manufacturing ecosystems in the textile domain—systems that do not just identify defects but actively prevent them and improve overall product quality.

VIII. CONCLUSION

The development of the AI-based automated fabric inspection system presented in this work represents a significant step forward in modernizing quality control processes in the textile industry. By integrating low-cost embedded computing hardware with advanced deep learning models and intelligent actuation, the system provides a robust and scalable alternative to traditional manual inspection techniques. The key innovation lies in the use of a Raspberry Pi 5 paired with a lightweight TensorFlow Lite CNN model, enabling real-time defect detection with over 92% accuracy — all without the need for external GPUs or cloud processing.

Unlike conventional systems that are either fully manual or prohibitively expensive, this system bridges the affordability-accessibility gap, making it a viable solution for small to medium textile manufacturers. Its modular design and software-driven logic allow for rapid deployment, easy upgrades, and minimal operator training. The marking mechanism, which utilizes a diaphragm pump and precision nozzle, offers an effective way to visually identify defects on the fabric for further analysis or reprocessing. This not only minimizes waste but also enhances traceability across production batches.

In practical terms, the system improves operational efficiency, reduces human fatigue-induced errors, and promotes data-driven quality control. The ability to log, store, and analyze defect data creates opportunities for long-term process optimization. Moreover, the system's reliance on open-source tools and off-the-shelf components ensures sustainability and ease of maintenance, even in resource-constrained environments. Overall, this research establishes a strong foundation for implementing intelligent, low-cost automation solutions within India's growing textile manufacturing ecosystem.

IX. ACKNOWLEDGMENT

The authors would like to express their sincere gratitude to the Department of Mechatronics Engineering, PSG College of Technology, for providing the resources, facilities, and guidance essential to the successful completion of this project. We are especially thankful to our project guide,D.Sujatha, for their continuous mentorship, valuable technical input, and encouragement throughout the development and testing phases of the system.

We also acknowledge the support of laboratory staff and fellow student collaborators who assisted in assembling the prototype and conducting performance evaluations. Their contributions in troubleshooting and experimental validations played a crucial role in achieving the system's accuracy and reliability.

Lastly, we extend our appreciation to open-source communities of TensorFlow, OpenCV, and the Raspberry Pi Foundation, whose platforms enabled the integration of advanced AI and embedded technologies in a cost-effective manner.

REFERENCES

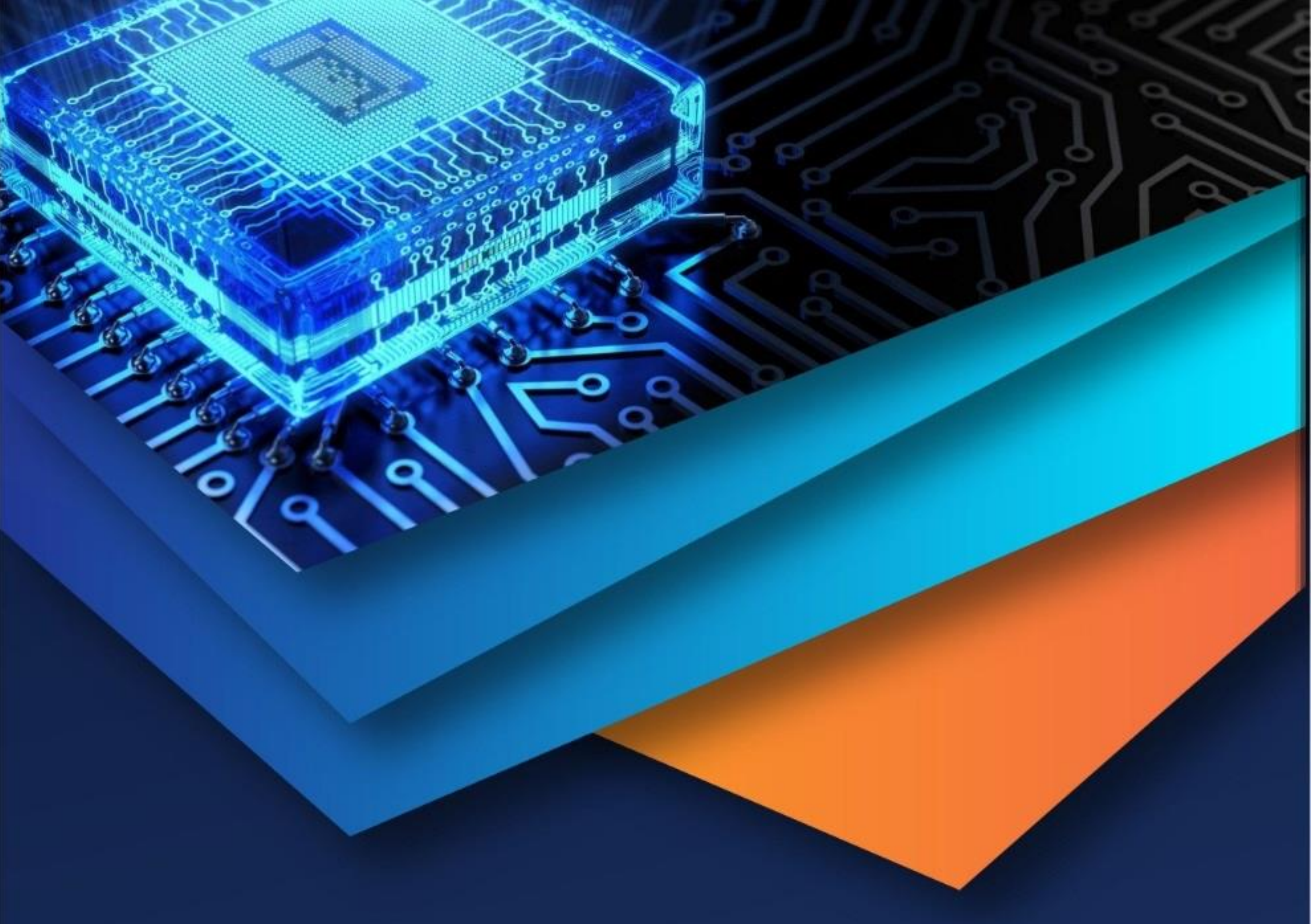
[1] Kim and D. Lee, "Fabric Defect Detection Based on Artificial Intelligence," Int. J. Intell. Syst. Appl. Eng., vol. 11, no. 5s, pp. 01–07, 2024.

[2] W. H. Cheung and Q. Yang, "Fabric Defect Detection Using AI and Machine Learning for Lean and Automated Manufacturing of Acoustic Panels," Proc. IMechE Part B: J. Eng. Manuf., vol. 238, no. 12, pp. 1827–1836, 2024.

[3] B. Mohankumar and R. Venkatesan, "An Intelligent System for Automatic Fabric Inspection," Asian J. Inf. Technol., vol. 13, no. 3, pp. 308–312, 2014.

[4] K. S. Prakash and R. N. Mohanty, "Automated Fabric Inspection Using Machine Vision," in Proc. Int. Conf. on Computing, Communication and Automation, pp. 450–455, 2017.

[5] Y. Wang, Q. Zhou, and X. Lin, "Fabric Defect Detection Using Deep Learning," School of Computer Science and Technology, Tsinghua University, China, 2020.
[6] R. Harinarayan and V. D. Kulkarni, "Automatic Textile Defect Detection and Classification Using Image Processing Techniques," Dept. of Electronics Eng., BMS College of Eng., India, 2019.
[7] H. Xu, Y. Wang, and T. Liu, "Fabric Defect Detection Using Convolutional Neural Networks (CNN)," Dept. of Computer Science Eng., China, 2020.
[8] A. Patel and S. Joshi, "Real-Time Fabric Defect Detection System on Raspberry Pi Using Machine Learning," Dept. of Computer Science, Nirma University, India, 2021.
[9] M. Ghiasi and M. B. Menhaj, "Automated Visual Inspection of Textiles Based on Local Binary Patterns and Support Vector Machines," Dept. of Electrical Eng., Amirkabir University of Technology, Iran, 2018.
[10] P. Mahajan, S. R. Kolhe, and P. M. Patil, "A Review of Automatic Fabric Defect Detection Techniques," Adv. Comput. Res., vol. 1, no. 2, pp. 18–29, 2009.



10.22214/IJRASET

INDEX COPERNICUS

45.98

IMPACT FACTOR: 7.129

TOGETHER WE REACH THE GOAL

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY