International Journal of Engineering and Techniques-Volume 11 Issue 5, September - October - 2025

INTELLIGENT FABRIC DEFECT DETECTION USING DEEP LEARNING AND REAL-TIME VISION SYSTEMSAPPLICATION

[1]U. Satyanarayana, [2]P. Sana Nawaz, [3]P. Charmy, [4]Sk. Md. Sana Gousebee, [5]Sk. Vasiha Thapsum
[1]Assistant Professor, [2,3,4,5]UG Students, Department Of Computer Science & Engineering(AI&ML) Geethanjali Institute Of Science And Technology, Gangavaram (V), Kovur(M), SPSR Nellore (Dt), Andhra Pradesh, India-524137

ABSTRACT
Fabric defect detection plays a vital role in quality control within the textile industry. Computer vision-based inspection has become a key technology for enabling intelligent manufacturing. This study reviews advancements in intelligent fabric defect detection, focusing on algorithms, datasets, and detection systems. Detection methods are categorized into traditional and learning-based approaches. Traditional methods include model-based, spectral, statistical, and structural techniques, while learning-based methods are divided into classical machine learning and deep learning techniques. The study compares deep learning models based on their principles, accuracy, real-time performance, and practical applicability.
Additionally, it examines commonly used fabric defect datasets and deep learning frameworks, organizing public datasets and widely adopted models. To improve real-time defect detection, the YOLOV8 algorithm is utilized, offering high speed and accuracy in identifying irregularities. YOLO's efficiency in processing entire images in a single pass makes it well-suited for rapid textile inspection.

INTRODUCTION
FABRIC DETECTION
The textile industry plays a crucial role in global manufacturing, producing fabrics for clothing, upholstery, and industrial applications. Maintaining high fabric quality is essential for ensuring customer satisfaction and minimizing production waste. Surface defects such as holes, stains, incorrect patterns, and broken yarns can significantly impact product quality and market competitiveness. Traditionally, defect detection in fabrics relied on manual inspection, but this method has limitations, including high labor costs, human fatigue, and inconsistency in accuracy. As a result, modern textile industries are increasingly adopting intelligent quality control systems that leverage computer vision and artificial intelligence (AI) to automate defect detection.

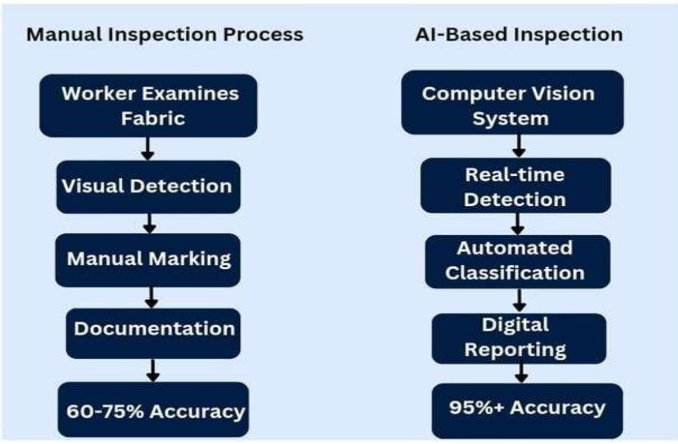


Fig 1.1: Fabric Inspection Methods

Traditional Fabric Inspection Methods
Historically, fabric quality control was conducted manually by trained inspectors who visually examined textile rolls for defects. However, this approach is time-consuming and prone to errors. Studies indicate that manual inspection achieves an accuracy of only 60-75%, often missing smaller defects. Additionally, as textile production speeds increase, human inspectors struggle to keep up with real-time monitoring requirements.

The drawbacks of manual inspection include inconsistency in detection accuracy among inspectors, high labor costs due to the requirement of skilled workers, and slow processing speed, which makes manual checks inefficient for large-scale production. Moreover, long hours of visual inspectionlead to worker fatigue, causing errors and reduced productivity. These limitations highlight the need for automated quality control solutions in the textile industry.

AI-Based Intelligent Fabric Inspection Methods

To overcome the limitations of manual inspection, textile industries have implemented AI-driven computer vision systems for automated defect detection. These systems use deep learning algorithms and image processing techniques to analyze fabric images in real-time, detecting defects with high precision. AI-powered fabric inspection offers significant advantages, including higher accuracy (over 95% defect detection accuracy compared to manual methods), faster processing speeds, and reduced dependency on manual labor, ultimately leading to cost savings and improved efficiency.

These AI systems can detect multiple types of defects such as misaligned patterns, broken yarns, oil stains, and weaving faults. Additionally, they are scalable and can be applied to various textile types, including woven, knitted, and non-woven fabrics. As a result, AI-based fabric inspection systems are becoming an essential component of modern textile manufacturing.

TYPES OF FABRIC DEFECTS

Fabric defects are flaws or irregularities that occur during the manufacturing, handling, or processing of fabrics. These defects can impact the appearance, strength, durability, and overall quality of the fabric. Identifying and classifying fabric defects is essential in the textile industry to ensure high-quality production and minimize waste. Defects may arise from various stages such as weaving, knitting, dyeing, finishing, or even cutting and storage.

Cutting

Cutting defects occur during the fabric cutting process in garment manufacturing.

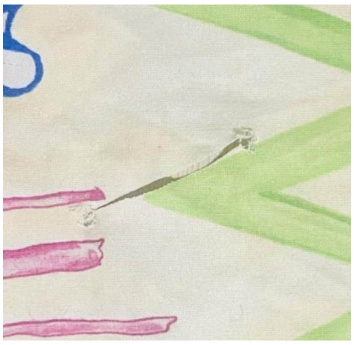


Fig 1.2.1.: Cutting

Causes:

- Improper alignment of fabric layers while cutting.
- Use of blunt or damaged cutting blades.
- Incorrect cutting patterns or templates.

Effects:

- Irregular or uneven edges.
- Weak spots that can lead to tearing during sewing or wearing.

Detection:

- Visually inspecting the edges and shapes of fabric pieces.
- Using machine vision systems that can detect inconsistencies in fabric outlines.

Oil Stains

Oil defects refer to unwanted oily patches or marks on the fabric surface.

Fig 1.2.2.: Oil Stains

Causes:

- Leakage or spills from sewing machine components.
- Oil from fabric rollers or production equipment.

Effects:

- Permanent stains that are hard to wash out.
- Reduction in fabric quality and appearance.
- Rejection of affected garments or fabric rolls.

Detection:

- Computer vision and deep learning is used to spot uneven textures or co**Hole**

Holes are unwanted openings or gaps in the fabric caused by damage to the yarns or weave during manufacturing or handling.

Fig 1.2.3.: Hole

Causes:

- Yarn breakage during weaving or knitting.
- Mechanical damage from sharp tools or rough handling.
- Chemical burns or extreme tension on the fabric.

Effects:

- Reduced strength and durability of fabric.
- Aesthetic defects making the fabric unusable for high-quality garments.
- Possibility of holes growing larger over time.

Detection:

- Manual inspection or use of automated vision systems that detect discontinuities or gaps in fabric structure.

TRADITIONAL FABRIC DETECT DETECTION METHODS

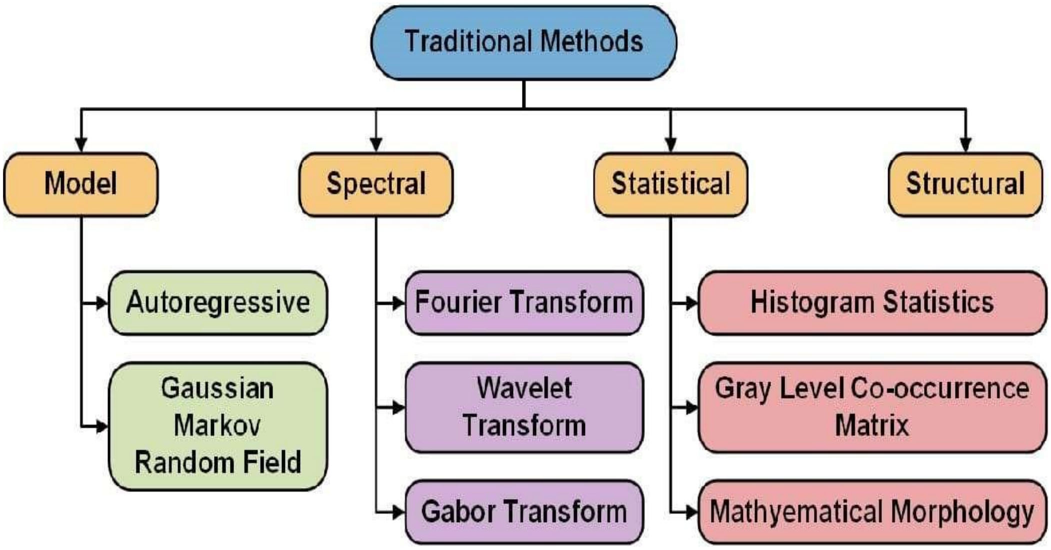


Fig 1.3: Traditional Methods of Fabric Detection

1.3.1Model-Based Methods

Model-based methods use mathematical models to identify patterns and defects in fabrics. These models assume that normal fabrics follow a predictable structure, and any deviations indicate defects.

1) **Autoregressive Model:** This method predicts fabric patterns based on previous pixel values. If a significant deviation is detected, it indicates a defect. It is useful for detecting repetitive patterns and sudden irregularities in fabrics.
2) **Gaussian Markov Random Field (GMRF):** This technique models the texture of fabrics using probability-based methods. It analyzes pixel relationships to differentiate between normal textures and defective areas.

1.3.2Spectral Methods

Spectral methods analyze the frequency components of fabric images to detect irregularities that might not be visible in the normal image space.

1) **Fourier Transform:** This method converts fabric images into frequency components, helping to detect periodic patterns and sudden distortions. It is useful for identifying repetitive defects such as misaligned weaves.
2) **Wavelet Transform:** This technique breaks down images into different scales, allowing detection of both large and small defects. It is useful for multi-scale defect identification, such as tiny cracks or larger misweaves.
3) **Gabor Transform:** This method uses filters to detect fabric patterns in different orientations and scales. It helps in recognizing directional defects like thread misalignment or streaks in woven fabrics.

1.3.3Statistical Methods

Statistical methods analyze the distribution and relationships of pixel values in fabric images to identify anomalies.

1) **Histogram Statistics:** This method examines the brightness and contrast distribution of fabric images. If certain values deviate from the expected range, it may indicate defects such as stains or holes.
2) **Gray Level Co-occurrence Matrix (GLCM):** This technique evaluates how pixels relate to their neighbors, helping to detect texture irregularities. It is widely used for identifying subtle defects like roughness or minor distortions.
3) **Mathematical Morphology:** This approach uses shape-based operations (like erosion and dilation) to highlight defects in fabric structures. It is useful for detecting small holes or broken threads.

1.3.4Structural Methods

Structural methods analyze the geometric arrangement of fabric patterns to detect deviations. These methods focus on identifying misaligned weaves, missing threads, or pattern distortions based on predefined structural rules. They are particularly useful in detecting woven fabric defects where regular patterns should be maintained.

1.1 LEARNING-BASED APPROACH TO FABRIC DEFECT DETECTION

Fabric defect detection using learning-based approaches involves machine learning and deep learning techniques to improve accuracy and automation. These approaches analyze fabric textures, identify defects, and classify them efficiently.

1.4.1.Classical Machine Learning Methods

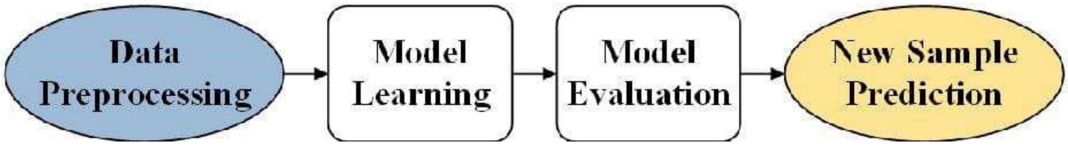


Fig 1.4.1.: Classical Machine Learning Methods

Classical machine learning methods rely on statistical and mathematical models to detect defects. The process starts with data preprocessing, where images are cleaned, resized, and features such as texture, color, and shape are extracted. Then, the model learning phase trains algorithms like Support Vector Machines (SVM), Random Forest, and K-Nearest Neighbors (KNN) to differentiate between normal and defective fabrics. After training, the model evaluation phase tests performance using accuracy and error metrics. Finally, in the new sample prediction phase, the trained model classifies fresh fabric images as defective or non-defective.

1.4.2.Deep Learning Methods

Deep learning methods can be broadly classified into 3 types depending on the type and availability of labelled data.

1. supervised,
2. unsupervised, and
3. semi-supervised learning.

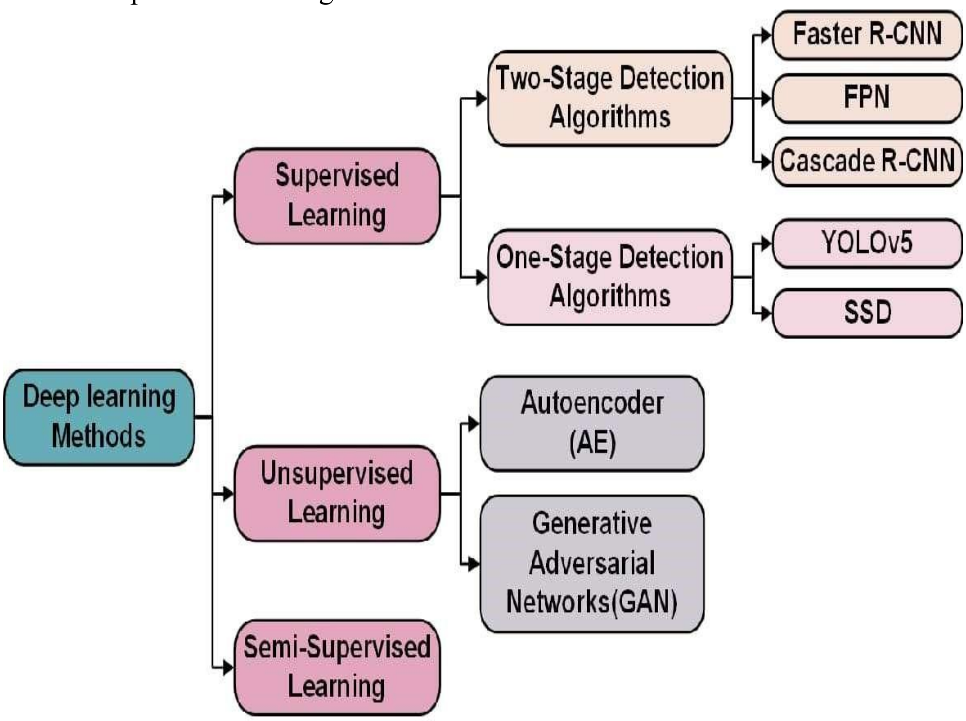


Fig 1.4.2.: Deep Learning Methods

1. **Supervised Learning**

Supervised learning requires labeled datasets to train models. Some popular deep learning models include:

1) **Convolutional Neural Networks (CNNs)**

Convolutional Neural Networks (CNNs) are a type of deep learning algorithm commonly used in computer vision tasks like fabric defect detection. CNNs are inspired by the way the human visual system works. They can automatically extract spatial features from images, making them well-suited for tasks such as image classification, object detection, and segmentation.

CNNs consist of multiple layers such as convolutional layers, pooling layers, and fully connected layers. The convolutional layers apply filters to extract important features like edges, textures, and patterns from the input fabric images.

Pooling layers reduce the spatial size of the feature maps, which helps in lowering computational cost and controlling overfitting.

Fully connected layers take the flattened feature maps and make predictions based on the features learned by previous layers.

Difference of CNNs

Traditional machine learning algorithms often rely on handcrafted features, while CNNs learn these features automatically from the raw image data. CNNs are end-to-end models, meaning they can take input images and directly output predictions without needing manual feature engineering. CNNs also maintain spatial relationships between pixels through their convolutional structure, making them highly effective for visual defect analysis.

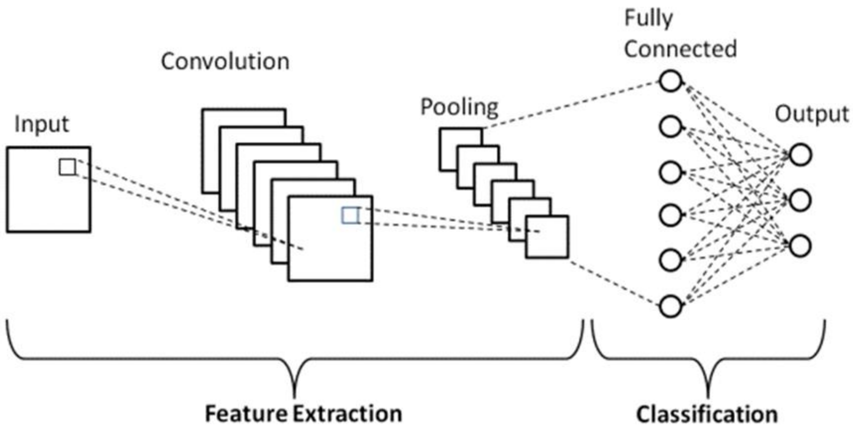


Fig: 1.4.2.1 Convolutional Neural Networks

The pseudocode of how CNN works is given below:

1. Input Image Preparation:
- Input image is resized to a fixed dimension and normalized.
2. Initialize Network Layers**:**
- Initialize layers including convolution, activation, pooling, and fully connected layers.
3. Feature Extraction:
- Convolutional layers apply filters to extract low- and high-level features.
- Pooling layers downsample feature maps to reduce size and computation.
4. Classification/Detection:
- Flatten the pooled feature maps and feed them into fully connected layers.
- Use a softmax or sigmoid function to classify fabric defects or detect their presence.
5. Make Predictions:
- Output is a label or bounding box indicating the type and location of the defect in the fabric.

2) Faster R-CNN (Region-based Convolutional Neural Network):

Faster R-CNN is a two-stage object detection framework that improves upon Fast R-CNN by incorporating a Region Proposal Network to generate region proposals. It detects and localizes defects by identifying specific regions of interest in fabric images.

3) FPN (Feature Pyramid Network):

In deep learning, FPNs are architectures that construct multi-scale feature maps, enabling object detection and other tasks to handle objects at different scales efficiently. It enhances defect detection by analyzing fabric images at multiple scales, making it effective for detecting both small and large defects.

4) Cascade R-CNN:

Cascade R-CNN is an object detection architecture that seeks to address problems with degrading performance with increased thresholds. It is a multi-stage extension of the R-CNN, where detector stages deeper into the cascade are sequentially more selective against close false positives. Itimproves detection accuracy by refining defect predictions in multiple stages.

5) **YOLOv5 (You Only Look Once):**
YOLOv5 is apopular real-time object detection model developed by Ultralytics. It is a part of the YOLO family, which aims to perform object detection in a single pass through a neural network. This algorithm processes fabric images in a single pass, making it ideal for fast-moving textile production**.**

6) **SSD (Single Shot Detector):**
SSD is an object detection model that detects multiple objects in images using a single forward pass through the network. It quickly detects defects at different positions in an image, making it suitable for high-speed inspections.

2. **Unsupervised Learning**

Unsupervised learning does not require labeled data and instead learns patterns from normal fabric images to identify anomalies.

Autoencoder: A neural network that compresses and reconstructs fabric images, detecting defects by highlighting areas that deviate from normal patterns**.**

Generative Adversarial Networks (GANs): Uses two networks (a generator and a discriminator) to learn fabric textures and identify defects by comparing real and generated images

3. **Semi-Supervised Learning**

Semi-supervised learning combines a small amount of labeled data with a large amount of unlabeled data to improve defect detection. This approach is useful when labeled fabric defect datasets are limited. Techniques like self-training, consistency regularization, and pseudo-labeling help train deep learning models more efficiently while reducing manual labeling efforts.

LITERATURE SURVEY

[1] **Y. Li, H. Wang, Z. Wu, and J. Chen, "Deep Learning-Based Automated Fabric Defect Detection System"**

This study explores the application of deep learning techniques in fabric defect detection. The authors developed a convolutional neural network (CNN)-based model that identifies surface defects such as holes, stains, and misaligned patterns with high accuracy. The model was trained using a large dataset of textile images and demonstrated a detection accuracy exceeding 95%. The research highlights the potential of AI-driven quality control systems in improving textile manufacturing efficiency.

[2] **M. Zhou, X. Zhang, and T. Liu, "Computer Vision for Textile Defect Detection: A Review"**

The paper provides a comprehensive review of computer vision techniques used in textile defect detection. It covers traditional image processing methods, such as edge detection and thresholding, as well as modern machine learning approaches like deep learning and support vector machines (SVM). The authors discuss the strengths and limitations of different methods and suggest future research directions for improving detection accuracy and real-time processing speed.

[3] **S. Patel, R. K. Singh, and P. Verma, "Unsupervised Learning for Anomaly Detection in Textile Fabrics"**

The study investigates the use of unsupervised learning techniques, specifically autoencoders and Generative Adversarial Networks (GANs), for detecting surface defects in textiles. Unlike supervised models that require labeled datasets, these methods can identify defects without prior annotations. The findings demonstrate that unsupervised learning models can effectively generalize to new defect types and improve defect detection robustness.

[4] **T. Nakamura, Y. Sato, and M. Kobayashi, "Edge AI for Smart Textile Quality Control"**

This paper explores the implementation of Edge AI for fabric defect detection, enabling real-time quality control directly on production machines. The authors developed a lightweight deep learning model optimized for

edge devices, reducing dependency on cloud computing. The study reports significant improvements in detection speed and system efficiency, making AI-driven fabric inspection more scalable and practical for industrial use.

[5] **D. Park, S. Kim, and J. Lee, "Application of HyperspectralImaging for Textile Defect Detection"**
The research investigates the potential of hyperspectral imaging in fabric quality control. Unlike traditional RGB cameras, hyperspectral imaging captures a broader spectrum of light, revealing subtle defects that are invisible to the human eye. The authors integrated machine learning models to classify different defect types based on spectral signatures, improving defect detection accuracy and reducing false positives.

[6] **C. Yang, X. Wu, and L. Zhang, "Automated Fabric Inspection Using Faster R-CNN and Transfer Learning"**
This study applies the Faster R-CNN deep learning model for fabric defect detection. The authors employed transfer learning techniques to enhance model performance with limited training data. Their experimental results showed that Faster R CNN outperformed conventional computer vision methods in detecting small and complex defects with high precision, making it a promising approach for textile quality control.

[7] **P. Das, A. Gupta, and S. Mukherjee, "Internet of Things (IoT) Enabled Fabric Defect Monitoring System"**
The paper discusses the integration of IoT sensors with AI-based defect detection systems in textile manufacturing. The proposed system collects real-time fabric images and sensor data, which are analyzed using AI models to identify defects. The study highlights the potential of IoT-driven smart manufacturing in enhancing production efficiency, reducing human intervention, and improving quality control.

[8] **L. Wang, M. Xu, and H. Zhao, "Hybrid Deep Learning Model for Multi-Class Fabric Defect Detection"**
Kim, P. Joshi, P. The research presents a hybrid deep learning model combining CNN and Recurrent Neural Networks (RNN) for detecting multiple types of fabric defects. The model learns spatial features using CNN layers while capturing temporal dependencies with RNN components. Experimental results demonstrate improved classification accuracy and robustness, particularly in identifying subtle defects and complex patterns.

[9] **X. Li, J. Feng, and W. Zhou, "Generative Adversarial Networks for Synthetic Fabric Defect Data Generation"**
This paper explores the use of GANs for generating synthetic fabric defect images to augment training datasets. The authors highlight the challenge of acquiring labeled defect images for AI model training and propose GANs as a solution. The study found that training deep learning models on both real and synthetic defect images enhances detection accuracy and generalization capabilities, making AI-based fabric inspection systems more reliable.

Despite the advancements in AI-driven fabric defect detection, the existing approaches have several drawbacks. Many CNN-based models ([1], [5], [8]) require large labelleddatasets, making data acquisition a challenge. Traditional computer vision techniques ([2]) struggle with complex and subtle defects. Unsupervised learning methods ([9]) face difficulties in distinguishing normal texture variations from actual defects, leading to false positives. Edge AI solutions ([3]) are constrained by hardware limitations, reducing model complexity and accuracy. Hyperspectral imaging ([4]) improves defect detection but involves high costs and complex implementation. IoT-enabled monitoring systems ([7]) introduce latency and network dependency. Faster R-CNN ([6]) shows strong performance but is computationally expensive, limiting real-time applications.

To address these challenges, we propose using the YOLOv8 algorithm, which offers high-speed, real-time object detection with superior accuracy. YOLOv8's lightweight architecture ensures deployment feasibility on edge devices, reducing dependency on cloud computing. It also requires fewer training samples compared to traditional deep learning models, making it a robust solution for improving textile defect detection accuracy and efficiency.

METHODOLOGY

3.1 SYSTEM ARCHITECTURE

Deep learning models, such as YOLOv8, rely heavily on high-quality data for training and accurate defect detection. If the data is not properly pre-processed, even the most advanced model will fail to deliver accurate results. Therefore, ensuring the right format, scale, and relevant features in the dataset is crucial for an efficient fabric defect detection system.

Data preprocessing plays a key role in improving model performance by cleaning, transforming, and augmenting fabric images before feeding them into the YOLOv8 model. This ensures better feature extraction and classification of defects such as stains, holes, and cuts in textiles.

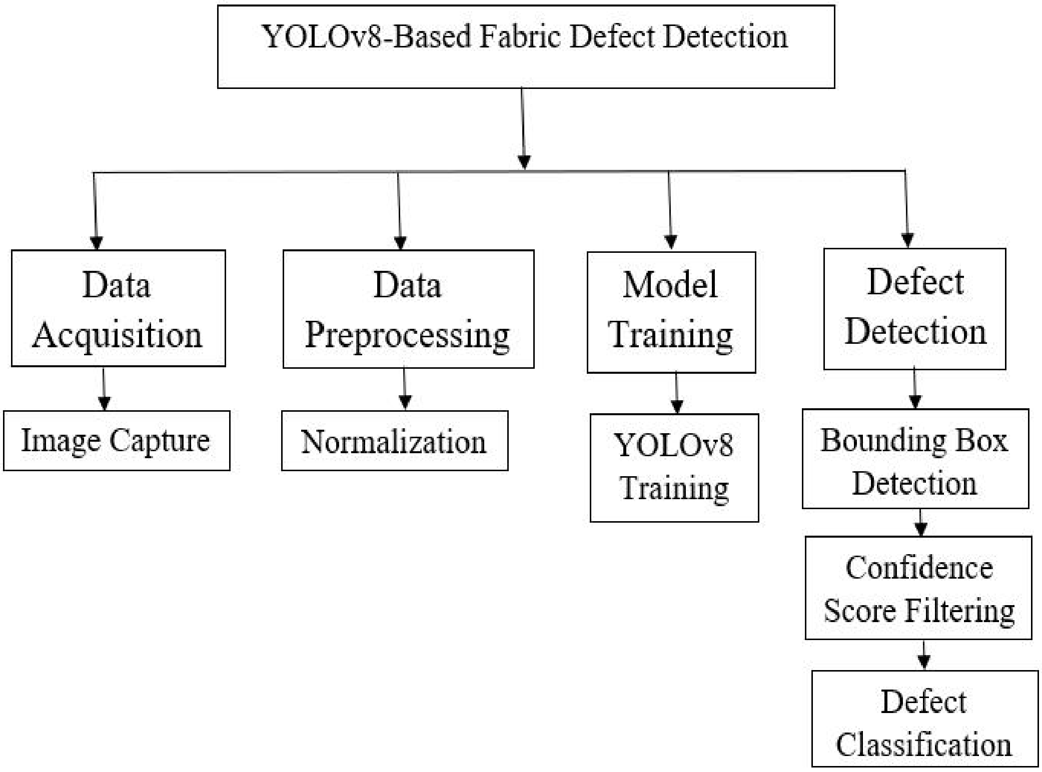


Fig: 3.1: Architecture Diagram

Image Preprocessing Module:

Image preprocessing is an essential step to prepare the raw fabric images for analysis. This process includes:

1. **Resizing & Normalization** – Adjusting image dimensions and pixel values to maintain consistency.
2. **Noise Reduction** – Enhancing image clarity by removing unwanted distortions.
3. **Augmentation** – Applying transformations like rotation, flipping, and contrast adjustment to improve model generalization.

YOLOv8 Model for Fabric Defect Detection:

The YOLOv8 model is used for real-time defect detection by analyzing images and identifying defective areas with high accuracy. It processes entire fabric images in a single pass and assigns bounding boxes to detected defects.

The advantages of using YOLOv8 Model are:

1. Detects defects in real time, improving the speed of quality control.
2. Achieves high detection accuracy in textile defect classification.
3. Reduces manual effort and enhances fabric quality monitoring.

3.2 WORKFLOW

The fabric defect detection using YOLOv8 employs a systematic workflow to provide accurate andreal-time identification of defects in textile materials.It begins with the collection of high-quality fabric images and continues through data preparation, model training, and deployment to ensure reliable and efficient defect detection in production environments.

1. **Data Collection:**

The first and most critical step is gathering a high-quality dataset, as the accuracy of the model heavily depends on the quality and diversity of data.

- Images are collected from real-time fabric inspection systems, industrial surveillance cameras, or publicly available datasets related to textile manufacturing.
- The dataset consists of images showing non-defective fabrics, ensuring that the model learns accurately.
- Images are stored in standard formats and organized according to defect types (e.g., holes, stains).

2. Data Preprocessing:

Preprocessing ensures the raw data is transformed into a clean and structured format suitable for model training.

- Removal of irrelevant or misleading information in images.
- Uniform resizing ensures consistency and efficient model training.
- Pixel values are scaled to a standard rangeto facilitate faster convergence.
- Bounding boxes are drawn around defective regions, and each box is labeled with the corresponding defect type.

3. Training the YOLOv8 Model:

The labeled dataset is used to train the YOLOv8 model, which is designed for real-time object detection with high accuracy and speed.

- YOLOv8 uses an advanced convolutional neural network (CNN) to capture spatial features at multiple scales.
- Pre-trained YOLOv8 weights are fine-tuned on the fabric defect dataset to reduce training time and improve performance.
- The model learns to detect and classify defects through deep learning and convolutional layers.

4. Testing and Evaluating the Model:

- The trained YOLOv8 model is tested on a separate test dataset.
- Metrics such as accuracy, precision, and recall are used to evaluate the accuracy and efficiency of the defect detection system.
- The model processes the test images and predicts bounding boxes and defect labels.

5. Deploying the Model:

- The trained model is deployed in an automated fabric inspection system to detect defects.
- Using industrial cameras and edge computing devices, the model can identify and mark defective regions instantly.
- As fabric rolls under the camera, defects are identified and highlighted instantly, allowing immediate action.

6. Prediction Results and Accuracy:

- The output of the deployed model includes defect type, location, and confidence score for each detection.
- Bounding boxes are drawn around detected defects on the fabric images.

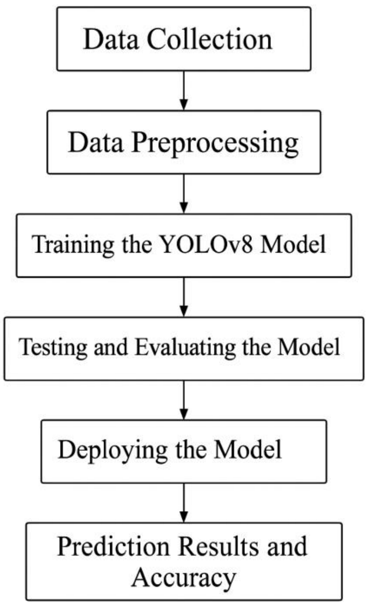


Fig: 3.2: Workflow

3.3 DATASET

In deep learning-based fabric defect detection, the dataset plays a crucial role in training, validating, and testing the performance of computer vision models. The dataset used in this project is specifically designed to support advanced surface defect detection tasks in textile products, which is a key component in modern automated quality control systems.

The dataset contains a total of 720 high-resolution images, equally divided among four major types of fabric surface defects: Oil, Hole, Cutting, and Crack. Each defect type is represented by approximately 180 images, ensuring a balanced distribution for training and evaluation.

To support a wide range of detection workflows, the dataset is provided in two formats:

- FD Folder includes original defect images with MATLAB-based annotationsfor bounding box and class labels.
- YOLO Folder is structured specifically for YOLO-based models like YOLOv5 and YOLOv8. It includes images and YOLO-style .txt label files for train, validation, and test sets, with annotations in normalized YOLO format.

This dataset is well-suited for tasks such as image classification, object detection, and segmentation. Its versatility and organization make it ideal for both academic research and real-world prototyping of defect detection models.

Moreover, it allows researchers to experiment with techniques such as data augmentation, transfer learning, and real-time deployment, providing a strong foundation for developing robust and efficient fabric inspection systems.

Fig: 3.3: Dataset

3.4 DATA PREPROCESSING

Data preprocessing is a crucial phase in fabric defect detection using deep learning. It transforms raw image data into a clean, structured, and standardized format that computer vision models can efficiently interpret. Proper preprocessing directly impacts model accuracy, generalization, and robustness, especially in industrial environments where lighting, fabric textures, and camera setups can vary widely.

The dataset used may contain inconsistencies such as noise, varied image sizes, or class imbalance, all of which must be addressed to train reliable models. Without this step, models may become biased or fail to accurately detect and classify surface defects.

Raw Data Handling and Image Normalization:

Images were captured using industrial cameras with high resolution. To ensure consistency and compatibility with CNN architectures, all images were resized to fixed dimensions.

Color Conversion and Noise Removal:

Depending on the defect type, grayscale conversion was applied when color information wasn't critical, while RGB was retained for color-sensitive defects. Gaussian filters and median blurring were applied to reduce camera noise and improve contrast.

Pixel Normalization:

Pixel intensities were scaled to a [0, 1] range or standardized using mean and standard deviation to ensure stable gradient updates during training.

Annotation Parsing and Label Encoding:

Annotation formats from various sources were converted to a unified structure compatible withYOLOv8. Defect classes such as "hole," "crack," "cutting," and "oil" were mapped to numeric labels for encoding.

Data Augmentation:

To address class imbalance and simulate real-world variability, augmentations such as horizontal/vertical flips, brightness/contrast changes, random rotations, affine transformations, and cropping were applied. This helps the model generalize better across different production environments.

Patch Extraction and Sliding Window Framing:

Large fabric images were divided into smaller overlapping patchesusing a sliding window approach with a defined stride. Each patch was labeled based on the presence of a defect, making it suitable for training object detectors like YOLO.

Dataset Splitting and Balancing: The dataset was split into Training (70%), Validation (15%), and Test (15%) sets.For datasets with heavy imbalance, oversampling or synthetic data generation techniques were used to balance the classes.

Need for Data Preparation and Preprocessing:

Machine learning algorithms like YOLOv8 perform better when trained on clean and well-structured data. For example, models do not support null or noisy values, so they must be removed or handled appropriately. Proper preprocessing improves model accuracy and ensures the system is reliable in real-time industrial settings.

3.5 CONSTRUCTION OF A PREDICTIVE MODEL

Deep learning models require extensive amounts of quality data, especially in applications like fabric defect detection. Initially, raw historical data such as high-resolution textile images are gathered from industrial settings, often containing various types of defects. However, this raw data cannot be directly used for training. It must first undergo preprocessing to ensure consistency, reduce noise, and standardize the input format.Then, a deep learning model like YOLOv8 is chosen, which is good at detecting objects quickly and accurately. The model is trained on a part of the data so it can learn to find defects like oil stains, holes, and cuts. It is then tested on new data to check how well it works. If needed, the model is improved by adjusting settings to increase accuracy and reduce mistakes. Once the model performs well, it is used in real time to automatically detect defects in fabric, helping to speed up quality checks and reduce manual work.

3.6 TRAIN / VALIDATION /TEST SPLIT

In fabric defect detection, the first step is to load the dataset which contains images of different fabric defects like oil stains, holes, and cuts using libraries such as NumPy, Pandas. This dataset is then divided into two parts: training, and testing sets.A common split ratio used is 70:30 for training and testing. Theimages are stored in variables prefixed with x, and thedefect labels are stored in variables prefixed with y.The training data is used to teach the model to recognize patterns and identify defects. After training, the model is evaluated on the test set to check its accuracy. We use methods like .fit() to train the model and .predict() to test it on new fabric images. The accuracy score is then calculated by comparing the predicted defects with the actual ones in the test set, giving us an idea of how well the model is performing.

3.7 CLASSIFICATION

The Classification algorithm is a Supervised Learning technique that is used to identify the type of fabric defect based on the features learned from the training data. In the context of fabric defect detection, a deep learning model like YOLOv8 is trained using labeled images where each image is marked with a specific defect type such as

Hole, Oil, Cut, or Crack. Once trained, the model can classify or detect defects in new unseen fabric images by analyzing the patterns it learned during training.

In this case, the output variable of Classification is a defect category like Hole or Oilrather than a numeric value. Since this technique is supervised learning, it uses labeled input data, which includes images along with their corresponding defect labels. The model maps the input variableto a discrete output function, enabling automatic and accurate classification of fabric defects in real-time production environments.

3.7.1Algorithms

Deep learning plays a major role in modern industrial applications like fabric defect detection, enabling automatic identification of surface issues in textile materials. It mimics the way the human brain works to recognize patterns in images and extract features without manual intervention. Deep learning models are particularly effective in analyzing high-resolution fabric images and detecting various defects such as holes, oil stains, cracks, and cuts. In this project, we are using supervised deep learning techniques, meaning the model is trained using labeled data where each fabric image is annotated with its corresponding defect type. Below are the deep learning algorithm used:

1. YOLOv8

1) YOLOv8

YOLOv8 is the latest version of the YOLO (You Only Look Once) object detection family developed by Ultralytics. It is a real-time deep learning algorithm widely used for object detection and image segmentation tasks, including fabric defect detection. YOLOv8 improves upon earlier versions by offering better speed, accuracy, and flexibility, making it ideal for industrial visual inspection systems.

a. YOLOv8 is based on a single-stage object detection model, meaning it predicts bounding boxes and class probabilities in one forward pass, making it very fast compared to two-stage detectors.
b. It uses a fully convolutional neural network with optimized layers and anchor-free detection heads, enabling it to detect fabric defects with high precision and recall. YOLOv8 supports image segmentation and classification alongside object detection in one unified model.
c. It also supports automatic batch size adjustment, model export in multiple formats, and enhanced training features like mosaic data augmentation and dynamic input scaling for improved generalization.

Difference of YOLOv8

Unlike traditional object detection methods that require separate steps for region proposals and classification, YOLOv8 performs both simultaneously, making it much faster. YOLOv8 also removes anchor boxes, reducing computational complexity and making model training more robust and efficient. Its end-to-end trainable pipeline allows real-time defect detection, especially useful for moving fabrics in production lines.

The pseudocode of how YOLOv8 works is given below:

1. Input Image Preparation:
- Resize input image and normalize pixel values.
2. Model Initialization:
- Load pre-trained YOLOv8 model or initialize with custom architecture.
3. Forward Pass:
- Pass the image through the CNN backbone to extract features.
- Use the neck to aggregate multi-scale features.
- Detection head predicts bounding box coordinates, class labels, and segmentation masks.
4. Non-Maximum Suppression (NMS):
- Remove overlapping bounding boxes based on confidence scores to reduce false positives.
5. Prediction Output:
- Return final list of bounding boxes and class labels corresponding to defects in fabric.

RESULTS

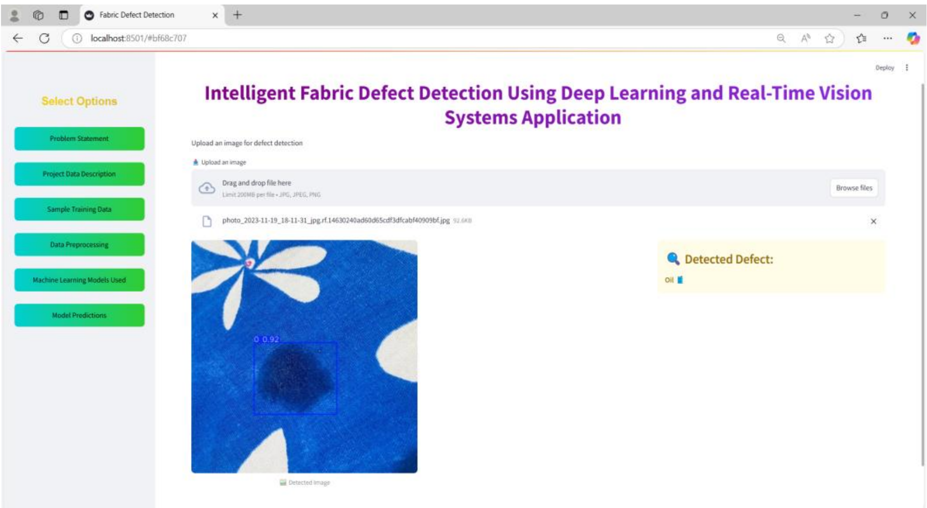
COMPARISON TABLE

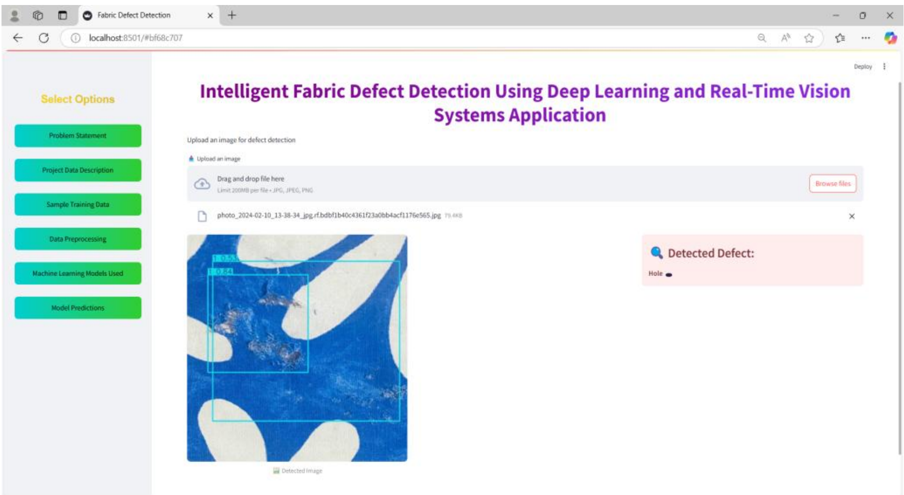
The below table consists of evaluation metric values for the models used, based on which we are finding the best approach.

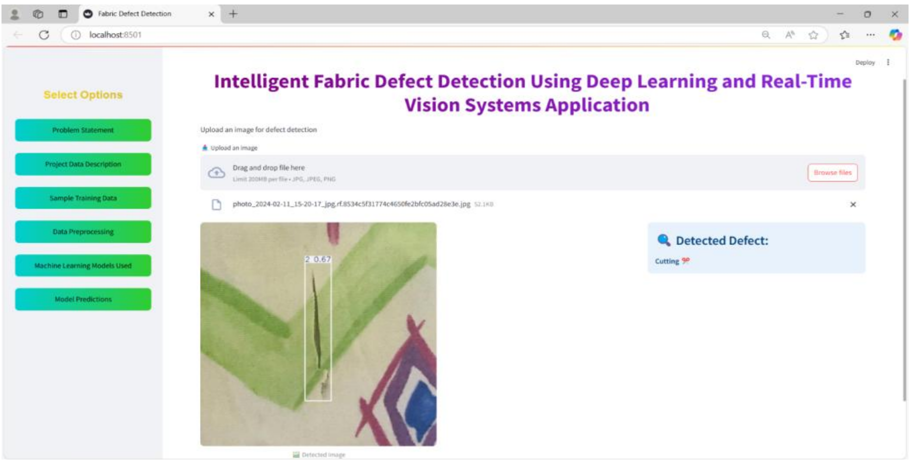
Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Faster R-CNN	90%	88%	85%	86%
Single Shot Detection (SSD)	88%	85%	83%	84%
YOLOv5	92%	91%	89%	90%
YOLOv8	95%	94%	93%	94%

Table 8.1 Comparison table for finding the best approach

SCREENSHOTS







CONCLUSION

The proposed system delivers a high-performance solution for fabric defect detection using the YOLOv8 deep learning model. It achieves real-time detection with an impressive 95% accuracy, enhancing textile quality control.By automating inspection, it reduces human error and boosts detection speed.The model effectively classifies various fabric defect types with high precision and recall.It is suitable for deployment in industrial environments with minimal hardware requirements.Edge deployment ensures fast, on-site processing without heavy cloud dependency.The system is flexible and can be adapted to different fabric types and manufacturing setups.Edge deployment ensures fast, on-site processing without heavy cloud dependency.The system is flexible and can be adapted to different fabric types and manufacturing setups.It supports consistent product quality while minimizing labor and inspection costs.Future improvements could include dataset expansion and integration with digital traceability systems.

FUTURE ENHANCEMENTS

The results show that YOLOv8 is the best approach for fabric defect detection, providing high accuracy and real-time performance. However, 100% defect detection efficiency is not yet achieved due to variations in fabric textures and lighting conditions. Hence, the performance of YOLOv8 can be further improved by integrating self-learning mechanisms and advanced AI techniques to enhance defect detection accuracy in future textile inspection systems.

References

1. In 2023, P. Das, A. Gupta, and S. Mukherjee, "**Internet of Things (IoT) Enabled Fabric Defect Monitoring System**," IEEE Access on 15 May.
2. In 2023, C. Yang, X. Wu, and L. Zhang, "**Automated Fabric Inspection Using Faster R CNN and Transfer Learning**," IEEE Access on 10 March.
3. In 2022, L. Wang, M. Xu, and H. Zhao, "**Hybrid Deep Learning Model for Multi-Class Fabric Defect Detection**," IEEE Transactions on Industrial Electronics, vol. 19, pp. 7854–7867.
4. In 2022, Y. Li, H. Wang, Z. Wu, and J. Chen, "**Deep Learning-Based Automated Fabric Defect Detection System**," IEEE Transactions on Industrial Informatics, vol. 18, pp. 11245–11256.
5. In 2021, M. Zhou, X. Zhang, and T. Liu, "**Computer Vision for Textile Defect Detection: A Review**," IEEE Access, vol. 9, pp. 55078–55095.
6. In 2021, S. Patel, R. K. Singh, and P. Verma, "**Unsupervised Learning for Anomaly Detection in Textile Fabrics**," in Proc. International Conference on Machine Learning (ICML), August, pp. 788–799.
7. In 2021, X. Li, J. Feng, and W. Zhou, "**Generative Adversarial Networks for Synthetic Fabric Defect Data Generation**," IEEE Transactions on Neural Networks and Learning Systems, vol. 32, pp. 5678–5690.
8. In 2020, T. Nakamura, Y. Sato, and M. Kobayashi, "**Edge AI for Smart Textile Quality Control**," in IEEE Internet of Things Journal, vol. 7, no. 5, pp. 3891–3902.

9. In 2020, D. Park, S. Kim, and J. Lee, “**Application of Hyperspectral Imaging for Textile Defect Detection**,” IEEE Sensors Journal, vol. 20, no. 8, pp. 4502–4513.