Grenze International Journal of Engineering and Technology, June Issue

Automatic Detection of Defects in Textile Fabrics

Dr. K Malarvizhi[1], J Clifford Lebo[2], G Kirubakaran[3] and V Selva[4]
[1-4]Coimbatore Institute of Technology, Coimbatore, India
Email: malarvizhi@cit.edu.in, {cliffordlebo02@gmail.com, kirubakarang041@gmail.com, neyveliselva@gmail.com

***Abstract*—Defects in fabrics affect the fabric quality at a larger scope. The goal of this project is to create sophisticated machine learning methods for computer-aided defect detection and classification. Deep convolutional neural networks are built to learn from different defect datasets during the training phase. The model was trained using customized YoloV5 model which archives a better real time performance. On detection of the defects they are documented separately and further analyzed to provide an ideology about the various defects using the deep learning algorithms. The defect classification accuracy is improved in the proposed model. The most important performance metrics of customized CNN models are classification accuracy, model complexity, and training time. Applying the defect detection methods improves product quality, meets customer requirements, and reduces cost depreciation.**

***Index Terms*— Defect Detection, Neural Network, YOLOv5 model, Decision Tree.**

I. INTRODUCTION

A. Fabric Defect Detection

Artificial intelligence has so far exceeded our expectations in industry and across all sectors. An important goal of artificial intelligence used in industrial applications is to use algorithms developed by artificial intelligence to solve industrial problems and improve production efficiency and safety. This is an interdisciplinary field made for engineering, data science, network, and communication. Therefore, in the field of textile production, it is necessary to self-regulate the production process. In the textile industry, textile defect detection is a quality control process that ensures that defects in textile fabrics are correctly identified and corrected in the subsequent stages of textile processing. These defects can reduce the price of fabrics by 40-60%.

Currently, most of the textile industry relies on manual human inspection. In many cases, human inspectors can easily find the defects by direct observation, but long-term observation can easily tire human eyes and cause an increasing number of defects to be missed, therefore, the traditional systems find it difficult to achieve the accuracy of 50 to 70%. Due to the very low defect detection rate compared to conventional production rates, typical automatic tissue inspection systems can detect defects with up to 90% accuracy and can be implemented to meet current needs and requirements.

This paper adopts a hybrid approach that utilizes statistical various open-source datasets and YOLOv5 - a CNN that is used here for fabric defect detection. Compared to human inspection, the main advantages of automatic defect detection systems are high efficiency, reliability, and consistency.

II. RELATED WORKS

The system presented in [1] is divided into several processes. First, the image acquisition system captures the

Grenze ID: 01.GIJET.9.2.311_1

tissue image and then preprocesses it with histogram smoothing to enhance the defect area. To identify the tissue defect, the image is then analyzed. A modified CNN architecture was used to develop a robust and computationally efficient fault detection system for the fault detection process. When the system detects a potential defect in the material roll, it stops and notifies the nearest operator. The operator then confirms or cancels the system's evaluation and allows the system to continue working and move on to the next image of the material. It also documents the type, location, and other characteristics of the defect so it can be inspected later. The accuracy obtained with this method is 80%.

A rule constraint generator [2] is used to generate an idle dataset and then build a lightweight model with an encoder and a decoder. The encoder section has a pyramidal transformation module that can adapt to tissue defects of various sizes. A gray image serves as the model's input, and fault segmentation produces the output. An idle data collection is utilized to train the model during the training phase. A test step uses a specific data set for testing. The accuracy obtained is 85%.

An improved SDCAE model [3] introduced as an image reconstruction model to reconstruct the images to be detected and determine the defect area in the image utilizing the similarity measurement. In addition, traditional defect detection methods use image blocks extracted from defect-free images. These methods randomly extract blocks of fixed-sized images from the defect-free images used to train the model.

The state-of-art detection results is achieved in [4] where the fabric defect probability map is introduced to the network as a dynamic activation layer, namely the pairwise potential activation layer (PPAL). A probabilistic map that incorporates prior knowledge of defects or statistical rules is essential for estimating potential defect areas. The accuracy obtained with this method is about 85.5%.

The main underlying assumption of method in [5] is that the defect is a texture property different from the background, that is, the texture is outside the background subspace. To take advantage of this control, a texture pre-map is made, indicating that input image pixels with higher values on this map are more likely to be defective pixels. The proposed method can handle simple or complex textured backgrounds and defect images with different defects. The accuracy obtained with this method is about 89%.

The FCSDA model proposed in [6] classifies the fabric patches into defectless and defective categories by using stacked denoising autoencoders. The method finds an advantage in classifying more complex jacquard warp-knitted fabric. It has been demonstrated that FCSDA-learned features are capable of adapting well to brightness variations and deformations, such as changes to scale and rotation angle. This method shows considerable performance in classifying fabrics with outstanding edges but not such optimized for defects with smooth grayscale differences. [7,10] provides an overview of computer vision-based approaches for detecting fabric defects in the textile industry is provided in this proposal. Among the methods studied are histogram-based approaches, color-based approaches, segmentation-based approaches, frequency domain operations, texture-based defect detection, sparse feature-based operations, and image morphology operations, as well as recent developments in deep learning. It also defines the various useful dataset that can be accommodated in this work for achieving better results.

The system proposed in [8] uses improved Faster RCNN and CBAM of classifying the fabric defect with a greater performance metrics .It introduces an attention module from a GNN(Graph Neural Network) which by inferring the attention map from the intermediate feature map, it can adaptively refine the feature by multiplying the attention map. It shows that applying computer vision based approaches in fabric defect detection can widely improves the performance accuracy.

The methodology proposed in [9] proposes an ideal method for detecting minor fabric defects using PETM-CNN algorithm. Here for the purpose of preprocessing the fabric image, the PE (Patches Extractor) algorithm extracts patches that may be defective patches. Then to predict the final label of the image, a TM- CNN (Triplet Metric CNN) method is used. Since this system concentrates only on minor fabric defects it doesn't able to achieve higher classification accuracy. The accuracy obtained with this method is about 79.32%.

Summarizing all relevant work, model over fitting and poor test accuracy are the dominant problems. To solve these problems, we use the YOLOv5 model, which is one of the powerful CNN algorithms. The data is also collected from various sources such as Kaggle and the TILDA dataset.

III. PROPOSED SYSTEM

A. System Architecture

The flow of the method is shown in Fig.1 portrays the overall flow of the system from feeding input to the final detection of the defects and distance i.e their position in the fabric material, thereby giving a clear view of the function of all the modules in the proposed system.

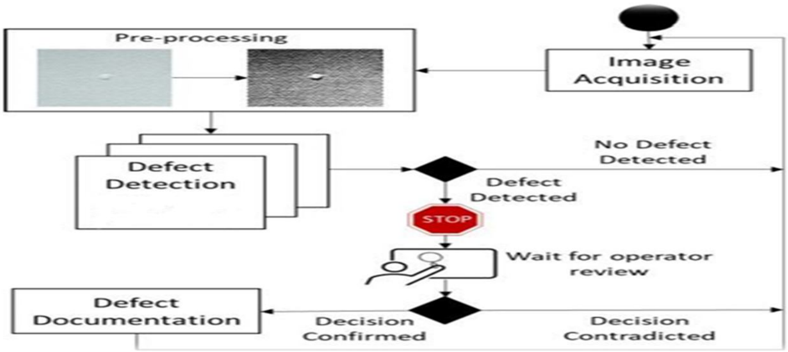

Fig. 1. System Architecture

As a first step, the entire system captures video using a camera illuminated by an external source of light. The video input is then passed on to the preprocessing module for processing. After the frames have been extracted from the video, they are subjected to a variety of preprocessing techniques, such as gray scale conversion, image augmentation, etc. As soon as the image frames have been processed, they will be loaded into the image detection module for defect detection. A classification process will be applied to each frame, determining whether there are any defects or not; if defects are detected, they will be further classified into one of the seven classes of fabric defects. Upon detection of a defect, the system checks for operator approval, an optional step. Finally, an Excel spreadsheet is used to document the defects, which includes both the location and type of defect within the fabric. By documenting, can later use this for defect rectification and further analysis for reducing the defects.

B. Dataset

For this system, 3 different datasets are used namely TILDA, fabric defect dataset (kaggle), and strain dataset

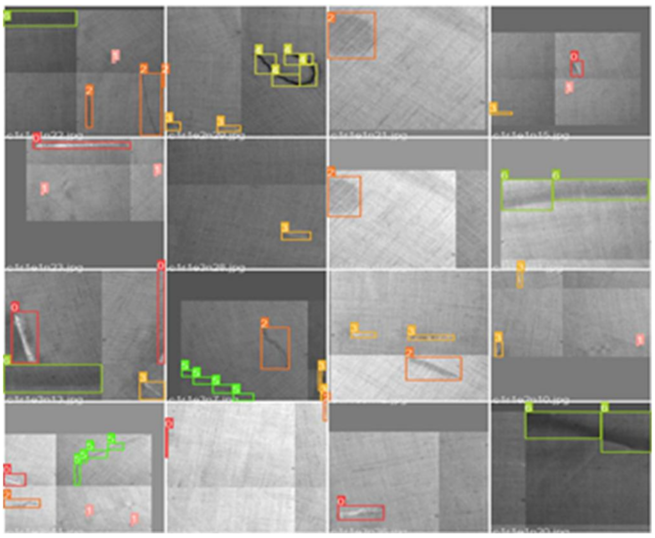

Fig. 2 Sample Dataset Images

The Tilda dataset is a dataset developed by the texture analysis team at DFG. This dataset is divided into six different types with seven defect classes representing 3200 images. Some images are not considered because those images are not related to the defects that are classified by the model.
Three kinds of defect images—horizontal, vertical, and hole—as well as three mask images are included in the Fabric Defect dataset from Kaggle. Dataset on Fabric Stains: The Intelligence Lab at the University of Moratuwa in Sri Lanka produced the dataset. This dataset includes two different fabric images of the spot defects.

C. Image Capturing and Preprocessing

It includes handling missing values and removing unwanted special characters.

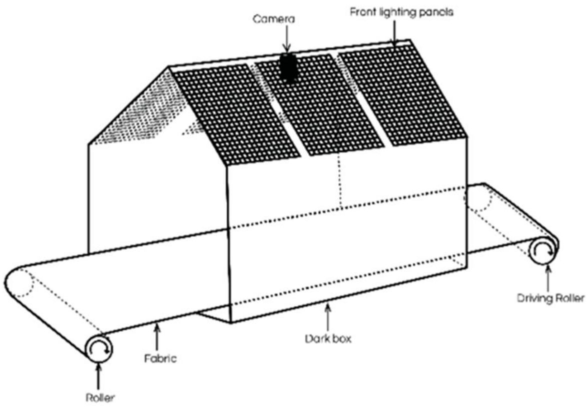

Fig 3 Image Capturing System

The images are collected from sources like video input and camera, these images need to be pre-processed as there could be a lot of noisy images in that dataset. This process includes grayscale conversion, normalization, data augmentation, and image standardization. All the images are reshaped to the dimension of 416 x 416 mm.
The grayscale conversion simplifies the algorithms used and eliminates the complexity associated with computation requirements. It also facilitates easy visualization of the data. It distinguishes between the shadow details and the highlights of an image since it is primarily two dimensional (2D) rather than three dimensional (3D).
To have a uniform pixel rate throughout the image, each frame must be normalized by adjusting the range of pixel intensity values. In order to increase the size of the dataset, image augmentation is performed, which involves rotating, shifting, flipping, and noise reduction of the video frames and obtained a uniform dataset after the preprocessing step, which can be used to train the model for the purpose of obtaining a more accurate resultant matrices

D. Defect Detection and Classification

Here, the YOLOv5 model is customized by altering the backbone (darknet) of the original code infrastructure to meet the requirement of getting higher accuracy and reducing false positive rate .Then the basic configurations of the detection module is altered to work with the respective classes and classify respectively .Then the hand crafted model is trained to perform the process of defect detection and classification over the data fetched from the sources and the results are mapped to the documentation module.
YOLOv5 model customization involves setting up the data.yaml file which is specific to the dataset and altering the custom_yolov5s.yaml file by changing the number of the backbone and head layers and reconstructing them as necessary.

E. Algorithm of the fabric defect detection model using YOLOv5

Input: Images captured by the system
Output: Identified defects and the corresponding location of those defects.
Begin
Step 1: The camera monitors the fabrics continuously.
Step 2: Frames will be extracted from the video and will be transferred to the YOLOv5 model
Step 3: When the defect is identified it will be noted and the boundary box will be displayed on the screen.
Step 4: The coordinates and other defects will be collected and stored as tuples of a CSV file.
Step 5: The tabulated details can be used to correct the defects and further research.
End.

F. Learning Approach- YOLOv5

1. Network architecture of the YOLOv5 model
2. Convolutional Layer

The convolution layer is used as the first layer of a CNN consisting of learnable channels spatially placed along the width and height of the information layer. The convolution is prepared by sliding each channel across the height and width of the information volume and calculates the spot results of the channel coefficients and

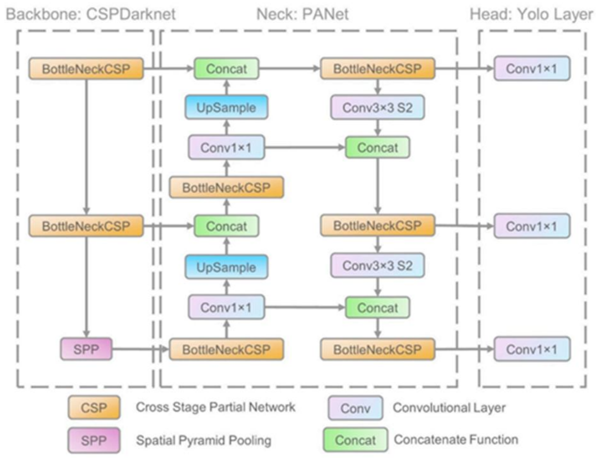

Fig 4 YOLOv5 CNN Architecture

contributions at random locations. At the same time, two initiatives for all information are created, which respond to each location of the channel. Instinctively, another 2D control guide for each channel is issued. Finally, all the starting cards are assembled and considered as the output of the convolutional layer.

3. Activation Layer

A network's fitting ability is increased by using the activation function. The neural network is usually connected by this type of connection between its nodes. An activation function based on Relu was proposed in order to solve the gradient disappearance problem and reduce the complexity of the computation. The Relu function defined in (1).A Relu activation function can be used to reduce the amount of computation, but it can also result in an increase in the sparsity of the net work, thereby solving the overfitting problem.

$$f_{(x)} = max(0, x) \quad (1)$$

when x is negative input, function returns 0 and when x is positive, function returns x itself.

4. Pooling Layer

It is sent through a pooling layer after the data is activated. Pooling takes the information which represents the image and compresses it, making it smaller. The pooling process makes the system more flexible and more apt at recognizing objects/images based on the relevant features.

5. Flattening

The final layers of the YOLOv5, the densely connected layers, require that the data be in the form of a vector to be processed. The data must be "flattened" for this reason. The long vector or a column of sequentially ordered numbers is compressed from the values.

6. Fully Connected Layer

A fully connected layer is the obvious type of closed layer that CNN should use. This is used to reinforce high performers across multiple loans that more accurately predict returns. In a fully connected layer, each neuron connects to the previous layer. This appears to be normal nerve tissue. The network gain is used to resolve the activation of the neurons.

G. Performance Evaluation

The method proposed in this work was evaluated and compared against state-of-the-art models using standard metrics based on the results. These metrics can be defined as:

$$Accuracy = \frac{TP+TN}{TP+TN+FN+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$Precision = \frac{TP}{TP+FP} \quad (4)$$

$$F1-score = (1+\beta^2) \quad \frac{recall \times precision}{(\beta^2 \times precision)+recall}, with\ \beta = 1 \quad (5)$$

$$MCC = \frac{TP+TN-FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \quad (6)$$

TP stands for True Positives, TN for True Negatives, and FP for False Positives.
(1) TP: defective images that have been detected or identified as defective
(2) TN: defect-free images that have been detected or classified as defect free
(3) FP: defect-free images that have been detected or classified as defective
(4) FN: defective images that have been detected or identified as defect-free

IV. IMPLEMENTATION

The selection of algorithms is the most important aspect of efficient implementation. Various object detection algorithms, such as Fast R-CNN, and a few versions of YOLO i.e. YOLOv3, YOLOv4, and YOLOv5 are thus compared.

A. R-CNN vs YOLO

R-CNN uses a region-based proposed method. RCNN does not take the full picture. RCNN takes a particular part of the image which has a higher chance of holding the object. The training time of the network is very large. It is determined that YOLOv3 has a higher F1 score and FPS than Faster R-CNN.

B. YOLO

The abbreviation for the term YOLO is 'You Only Look Once. This algorithm achieves real-time recognition of various objects in a frame. YOLO's object detection is a regression problem that provides the class probabilities of the detected images.YOLOv3 is accurate and faster compared to other deep learning algorithms such as Faster R-CNN and SSD. A comparison between the variation of the YOLO algorithm i.e. YOLOv3, YOLOv4, and YOLOv5 shows that YOLOv4 is more accurate than YOLOv3 in terms of both accuracy and FPS. However, the reported processing time of YOLOv4 versus YOLOv5 shows that YOLOv4 is significantly faster but far less accurate.

TABLE I: COMPARISON BETWEEN THE STRUCTURE OF YOLOV3, YOLOV4, AND YOLOV5

	YOLOv3	YOLOv4	YOLOv5
Neural Network Type	Fully convolution	Fully convolution	Fully convolution
Backbone Feature Extractor	Darknet-53	CSPDarknet53	CSPDarknet53
Loss Function	Binary cross entropy	Binary cross entropy	Binary cross entropy and Logits loss function
Neck	FPN	SSP and PANet	PANet
Head	YOLO layer	YOLO layer	YOLO layer

Since the model needs better accuracy and processing, the time for all the algorithms is almost similar. YOLOv5 is selected for the implementation of the model.

V. EXPERIMENTAL RESULTS

Tabel II. Shows the Experimental results obtained from evaluating the trained model against test split (25% of the complete Dataset). From the score obtained for various metrics it can be concluded the trained model could be used in a real time application for obtaining better working results.

TABLE II. RESULTS OF STANDARD METRICS

METRICS	SCORE
Precision	0.9233
Recall	0.8840
mAP_0.5	0.8955
Box_loss	0.0361
cls_loss	0.0036
obj_loss	0.0103

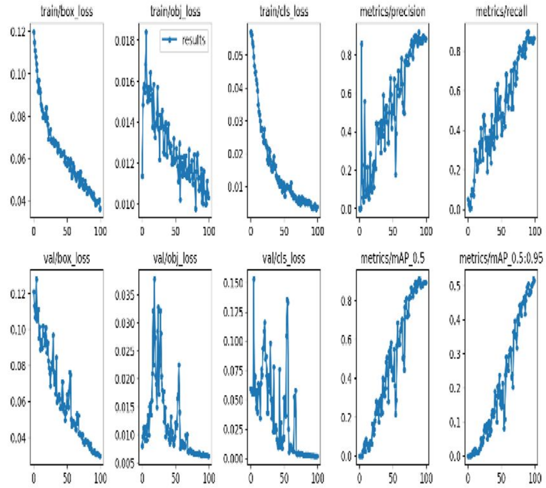

Fig 5. Result metrics of training model

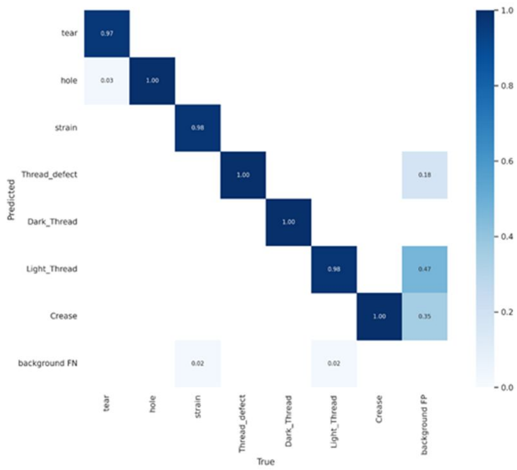

Fig 6. Correlation between the classes in the model

Fig 5 shows the evaluation of various metrics under time to epoch graphs and fig 4 shows the correlation relationship between the defect classes based on there occurrence on over training dataset.

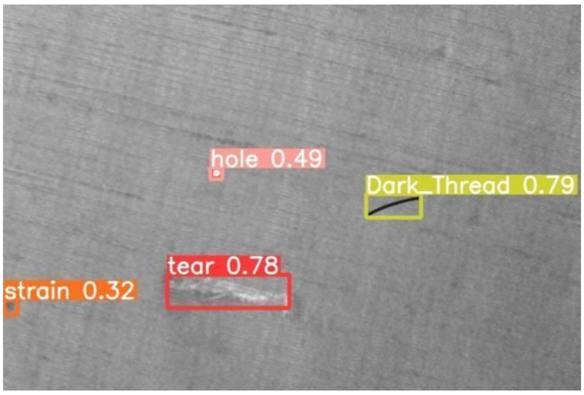

Fig .7 Testing result in real-time image

VI. Conclusion

Using the YOLOv5 algorithm to identify the defects in the fabric materials can improve the efficiency and the throughput as they can detect the defects with easy computation and less time. A state-of-art system is proposed to be successful in detecting defects in textile fabrics in real-time. A series of experimental results on a variety of textured fabric detection data sets show that this method can achieve the most advanced detection accuracy and high detection efficiency. Besides, the method can be run in real time, even on low-cost hardware.

Scope for Future

In future work, it will be interesting to develop larger datasets with real-world examples to enable the training and implementation of more sophisticated fault detection systems. Larger data sets allow the use of more complex models such as support vector machines (SVMs) and transformer networks. To improve the FN reduction function, test a more robust method performed during training, such as a custom loss function. The ultimate goal is to implement the developed system in a real environment to test and compare it with existing inspection systems.

References

[1] Almeida, Tomás, Filipe Moutinho, and João Pedro Matos-Carvalho. "Fabric Defect Detection with Deep Learning and False Negative Reduction." IEEE Access 9 (2021): 81936-81945.

[2] Wang, Zhen, and Junfeng Jing. "Pixel-wise fabric defect detection by CNNs without labeled training data." IEEE Access 8 (2020): 161317-161325.
[3] Xie, Huosheng, Yafeng Zhang, and Zesen Wu. "Fabric defect detection method combing image pyramid and direction template." IEEE Access 7 (2019): 182320-182334.
[4] Ouyang, Wenbin, et al."Fabric defect detection using activation layer embedded convolutional neural network." IEEE Access 7 (2019): 70130-70140.
[5] Huangpeng, Qizi, et al."Automatic visual defect detection using texture prior and low-rank representation." IEEE Access 6 (2018): 37965-37976.
[6] Li, Yundong, Weigang Zhao, and Jiahao Pan. "Deformable patterned fabric defect detection with fisher criterion-based deep learning." IEEE Transactions on Automation Science and Engineering 14.2 (2016): 1256-1264.
[7] Rasheed, Aqsa, et al. "Fabric Defect Detection Using Computer Vision Techniques: A Comprehensive Review." Mathematical Problems in Engineering 2020 (2020)
[8] He, Yuan, et al. "Fabric Defect Detection Based on Faster RCNN with CBAM." CS & IT Conference Proceedings. Vol. 11. No. 6. CS & IT Conference Proceedings, 2021.
[9] Wen, Zhijie, Qikun Zhao, and Lining Tong. "CNN-based minor fabric defects detection." International Journal of Clothing Science and Technology (2020).
[10] Silvestre-Blanes, Javier, et al. "A public fabric database for defect detection methods and results." Autex Research Journal 19.4 (2019): 363-374.
[11] Qiu, Junhao, et al. "Textile Defect Classification Based on Convolutional Neural Network and SVM." AATCC Journal of Research 8.1 (2021): 75-81.
[12] Zhang, Hong-Wei, et al. "Yarn-dyed fabric defect detection with YOLOV2 based on deep convolution neural networks." 2018 IEEE 7th data-driven control and learning systems conference (DDCLS). IEEE, 2018.
[13] Loonkar, Ms Shweta, and Dhirendra Mishra. "A Survey-Defect Detection and Classification for Fabric Texture Defects in Textile Industry." Statistics 50 (2015): 100.
[14] Banumathi, P., and G. M. Nasira. "Fabric inspection system using artificial neural networks." Int. J. Computer Engin. Sci 2.5 (2012): 20-27.
[15] Jeyaraj, Pandia Rajan, and Edward Rajan Samuel Nadar. "Computer vision for automatic detection and classification of fabric defect employing deep learning algorithm." International Journal of Clothing Science and Technology (2019).