Journal of Computing & Biomedical Informatics
ISSN: 2710 - 1606
Volume 08 Issue 02
2025

Research Article
https://doi.org/10.56979/802/2025

Textile Defect Detection in the Textile Industry using Deep Learning

Tanzeela Kiran[1], Mudasir Ali[2], Umair Ismail[1], Muhammad Altaf Ahmad[1], Urooj Akram[1], Wajahat Hussain[2], Muhammad Faheem Mushtaq[1,*]

[1] Department of Artificial Intelligence, The Islamia University of Bahawalpur, 63100, Bahawalpur, Pakistan
[2] Department of Computer Science, The Islamia University of Bahawalpur, 63100, Bahawalpur, Pakistan
[*]Corresponding Author: Muhammad Faheem Mushtaq. Email: faheem.mushtaq@iub.edu.pk

Received: February 04, 2025 **Accepted:** March 01, 2025

Abstract: In order to maintain high production standards, the quality control process in textile manufacturing mostly depends on the efficient detection of fabric flaws. Conventional defect monitoring techniques are labor-intensive, manual and prone to human mistake, which results in inconsistent quality. In this research, the hybrid deep learning model is proposed using convolutional neural networks and gated recurrent unit networks for textile defect detection in the textile industry. The goal is to increase this crucial process's precision, effectiveness, and dependability. Since fabric defect identification is a crucial step in quality control, it is one of the manual operations that has gradually been automated using the aforementioned techniques. The performance evaluations were conducted on proposed model and compare with other models including Convolutional Neural Networks (CNNs), Artificial Neural Networks (ANN), Recurrent Neural Networks (RNN), and Long Short-Term Memory (LSTM). CNN ability to extract features and GRU skill at sequential learning credited with this improved performance, which allow the model to successfully capture temporal and spatial relationships. The proposed hybrid model which combines CNN + Gated Recurrent Unit (GRU) are performed well as compare to other models and achieved the accuracy of 0.9841. The findings shows that CNN is a strong option for the given problem as it improves classification accuracy when combined with GRU.

Keywords: Deep Learning; Convolutional Neural Networks; Gated Recurrent Units; Textile Defect Detection; Fabric Quality Control

1. Introduction

In textile Industry, fabric tends to have a variety of flaws and distortion that have an enormous impact on the quality of the final outcome. Fabric is wasted due to a number of production flaws. May be a machine failure is one of the important causes of these flaws in fabric [1]. An inspection must be performed in order to determine whether the produced fabric meets market standard or not. Usually, this procedure involves human labor [2]. Actually, this is a manual inspection and highly skilled team of human labor is required to perform this manual inspection. On the other hand, this procedure takes a long time and human tiredness is one of the main causes that can lead to the defects [3]. These defects in fabric can also arise during manufacturing process due to variety of other factors as well like adding raw or unprocessed material, coloring, crafting or during the stage of sewing [4]. This leads to the creation of defective fabrics which could reduce customer satisfaction and quality of the fabric. Inferior items typically sent back to the textile mill for replacement which may lead to increase the overall production cost [5]. So, need to identify the defect and to remove them timely. Once we determine, how the flaws are manifesting, the production process could be enhanced to prevent them from happening in the initial stages rather than fixing them when they arise [6].

Although a great deal of research has been done on the detection of fabric defects and most of it has been based on datasets that involved carefully placed pictures of fabrics taken in controlled environment [7]. These kinds of datasets face a lot of difficulties in real-time production of fabric. The quality and placement of the images are poor because actual images vary from the real-time fabric that leads to the failure of fabric defect detection [8]. Moreover, the research on this issue can be divided into two main groups: the identification of faults in printed material and the identification of faults in plain fabric [9]. In case of printed fabric further segregation can be performed between regularly printed fabrics and irregularly printed fabric [10]. This is due to the fact that a wide range of prints make it challenging to differentiate between patterns and the flaws. As a result, various approaches must be performed to determine the stuff, quality and to identify the flaws in fabric [11].

In textile sector, upholding high standards of quality is essential and identifying fabric flaws is a key component of quality assurance [12]. Manual inspection is quite difficult because of wide variety of complexity of flaws that are identified by inherent uncertainty [13]. The efficacy of manual examination is further restricted by human exhaustion which leads to a poor detection rate. Hence, there must be an automated system that will be less prone to error and bugs, carefully examine every element and most importantly to turn out to be economical [14]. There must be a proper pattern related to these flaws, otherwise they may exhibit similar behaviors. Categorization can become considerably simpler if we detect some common kind of defects [15]. The ability to optimize the production process by concentrating on the individual processes that are creating certain types of faults is one of the many benefits of classification. By counting the number of defects in each class, we may determine which process step is the most problematic [16]. The textile industry is entering in a new era that is revolutionary and has a promising future. A comprehensive online system for detecting defects in textile fabrics relies on its detection algorithms [17]. According to researchers, neural networks with deep learning are highly accurate in handling classification challenges. As a result, automated defect detection systems driven by Machine Learning (ML) and Deep Learning (DL) methodologies have become increasingly popular [18]. Machine learning analyze fabric using data driven methods to identify patterns and abnormalities in fabric. Even in the cases where minor flaws in fabric occur, specific machine learning models are able to detect deviations from conventional fabric patterns [19]. Defects are classified by algorithms such as Support Vector Machine (SVM) and decision trees using predetermined features like color, texture of fabric and patterns. As ML systems learn from new data, they become better with the passage of time that make them more appropriate for adapting new manufacturing procedures [20].

In the same way intricate fabric benefits greatly from deep learning that elevates machine learning to a new level by automatically extracting properties from unprocessed data. Neural networks with Deep Learning (DL) plays a vital role in order to evaluate and finds defect much more quickly and accurately [21]. Due to the ability to learn and extract complex features from data, Convolutional Neural Networks (CNNs) have become one of the most effective methods for image-based defect identification [22]. CNNs, a subclass of deep learning models has completely changed computer vision and image processing [23]. CNNs are quite good at recognizing texture and pattern flaws such as holes, misalignments or stains. On the other hand, some more AI techniques in DL like Generative Adversarial Networks (GANs) and edge AI for real-time Monitoring and reinforcement learning are used for identification of flaws in fabric effectively [24]. Through these automated AI help to ensure the quality of fabric by detecting the flaws or defects at initial stages and by using ML and DL in fabric defect detection the accuracy will be achieved by identifies flaws with higher accuracy rate as compared to other manual methods. It also enables real time defect detection in production of fabric by minimizing the ratio of waste of fabric. It also reduces labor costs and easily adapt to different kind of fabric and production scales [25].

The important challenges that could be face by using deep learning in identifying defect in fabric is to attain high-quality labeled datasets for training because DL needs a significant computational resource. The following contributions are made in this context:

- This research demonstrated hybrid deep learning models for textile defect detection by utilizing Convolutional Neural Networks (CNN), Artificial Neural Networks (ANN), Recurrent Neural Networks (RNN), and Long Short-Term Memory (LSTM). The better results are obtained by hybrid the combined CNN + Gated Recurrent Units (GRU) model.

- To train deep learning models, the dataset is preprocessed for trials, which includes label encoding, image resizing, and data augmentation.
- The performance of the proposed method is thoroughly assessed using many commonly used assessment metrics, including as F1 score, recall, accuracy, and precision.

The remaining section of the document is structured as follows: The associated work that is pertinent to the proposed technique is explained in Section 2. The proposed CNN and GRU hybrid model were introduced in Section 3. The findings and analysis of this study are presented in Section 4 in order to assess the effectiveness of the proposed technique. The conclusion and future direction are finally concluded in Section 5.

2. Related Work

Due to the critical need of manufacturing high-quality textile, research on fabric defect detection has been ongoing for decades [26]. Many techniques which may be broadly categorized deep learning algorithm, spectral and statistical have been explored in past to automate the identification procedure of fabric's defect. The defective area is the one with a higher chance [27]. The CNNs receive the fabric defect probability map as a Pairwise Potential Activation Layer (PPAL) [28]. CNN was used to identify missing thread, oil stains, and hole flaws in the picture and to categorize the uniform textured materials with faults [29].

YOLO-based technique is proposed to identify hole faults, knot tying, and belt yarn in yarn-dyed cloth. YOLO9000, Tiny-YOLO, and YOLO-VOC were used to train the network; the corresponding precisions were 0%, 36%, and 86% [30]. They choose YOLO-VOC for their studies based on those first tests, and in order to optimize the network and increased the number of iterations to 20,000 and 30,000 with a learning rate of 0.01; this increased the precision to 94.5% and 90.6%, respectively [31]. Stacked Auto Encoders (SAEs), an efficient deep learning technique are used to identify flaws such as broken ends, holes, netted multiples, thick bars, and thin bars [32]. K-mean clustering was used to analyze the defect in dataset of images in order to examine the ideal anchor box size for YOLO identification. On a benchmark dataset of fabric images with 3000 samples and five classes, an accuracy of 97.2 was attained [33].

Their research was based on convolutional generative adversarial network. A residual map was created by subtracting the encoder's output which is the original image and the reconstructed image [34]. Possibly all faulty areas were highlighted in the map [35]. Using many data samples, the model attained a FNR of 12.09, an accuracy of 51.62%. However, the flaw for this model was noisy segmentation [36]. In this model the network mainly focusses on flaws by the attention mechanism. On the other hand, the proposed design uses feature concatenation to enhanced classification with the use of multi-task fusion [37]. By merging the classification and attention map branches, this fusion module enhances classification results even for minor flaws [38]. The proposed model obtained on precision 0.98, F1 score 0.987 and recall 0.994 but this was mainly implemented for plain fabric only [39].

Another research proposed a deep CNN architecture for classification for defects concentrated on developing a filter to eliminate images nonlinear mixed noise [40]. Image processing is done at first stage using Pseudo-convolutional neural network or P-CNN according to this study [41]. Like conventional convolutional network with three layers, adaptive window filters with weight initialization are used in the first feature extraction layer. A probabilistic distribution of noise is used to initialize these filter values [42]. A CNN is used in phase 2 to categorize and identify flaws. Mainly this model attained the results of 93.92% accuracy rate and 92.51% specificity rate. However, the algorithm struggled to accurately categorize the plain fabric samples and printed fabrics [43].

Multi-scale features were extracted from the input and template image using the Siamese feature pyramid network. An attention module was added to differentiate between input and template features [44]. A self-employed gadget was advised in order to correct the placement error between the input images and the reference images. This model yields the accuracy of 83.30% and a map of 47.1. The primary drawback of this study was that each design need template photos in order to identify the flaws or defects [45]. Furthermore, a fabric that include both plain and printed fabric for the training of a robust model. According to this study, nine courses and 19 distinct backgrounds with stain, holes, wrinkles, and thread end are implemented on cascade R-CNN [46]. The purpose of using these techniques was to fully detect the faults of small and medium sizes in high resolution snaps [47]. The final results obtained by combining these tiny fragments [48]. The results revealed a map of 75.30% but some flaws in this pattern still exist. Table 1 shows the summary of previous work.

Table Error! No text of specified style in document.**1.** Summary of the Related Work

References	Models	Results
[49]	ANN	Accuracy=95.46%
[50]	RNN	Accuracy=96%
[51]	CNN	Accuracy=98%
[52]	CNN	Accuracy=95%
[53]	Alex-net based CNN	Accuracy= 98.2%
[54]	CNN	Accuracy= 87.31%
[55]	VGG16	Accuracy= 98.1%
[56]	Dense Net	AUC= 18%

3. Proposed Methodology

The purpose of this work is to describe a non-locally centralized sparse representation-based defect detection model. To improve picture contrast, all photos are first preprocessed using grey-level transformation. Next, using adaptive compact sub-dictionaries derived from non-defective samples, the approximation pictures of the input images are created. Data preprocessing, which involves data cleansing, label encoding and feature extraction; data training and testing, where 80% of the dataset will be trained and 20% will be tested, using its corresponding model. Lastly, thresholding techniques are used to segregate any flaws in the residual picture of input image and its approximation. Fig. 1 shows the proposed work flow of system. The different primary components of the detection model are defect detection, which may be done in real-time, and dictionary learning, which is an offline procedure. The following will provide a detailed introduction to each of the proposed model's operations.

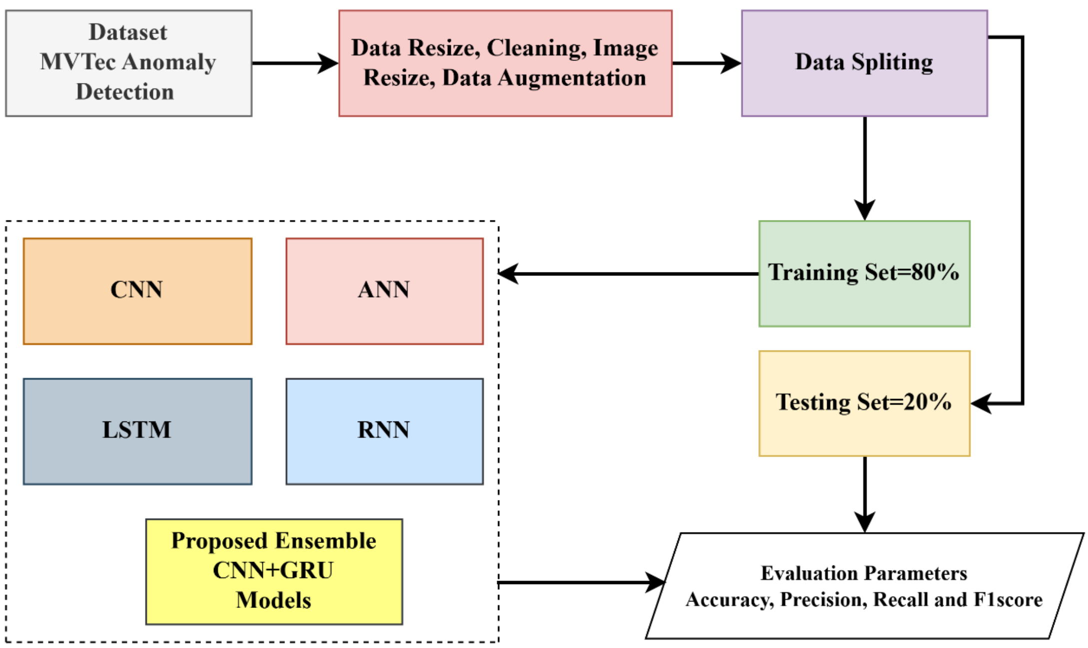

Figure 1. Proposed Methodology

3.1 Dataset Description

In the context of textile fabric, rare anomaly can occur, hence compromising the quality of the tissues. In order to avoid that in some scenario, it is crucial to detect the defect. This dataset is for educational purposes Image size: 32x32 or 64x64 classes: ['good', 'color', 'cut', 'hole', 'thread', 'metal contamination'] rotations: 8 different rotations in [0, 20, 40, 60, 80, 100, 120, 140]. Given an image size, a train and test dataset are available with randomly generated patches. Source images from the train and test are non-overlapping different tasks are possible: classification of the classes type classification of angles using only "good" images and testing of other classes texture representation learning / self-supervised. Fig. 2 shows the classes of the dataset.

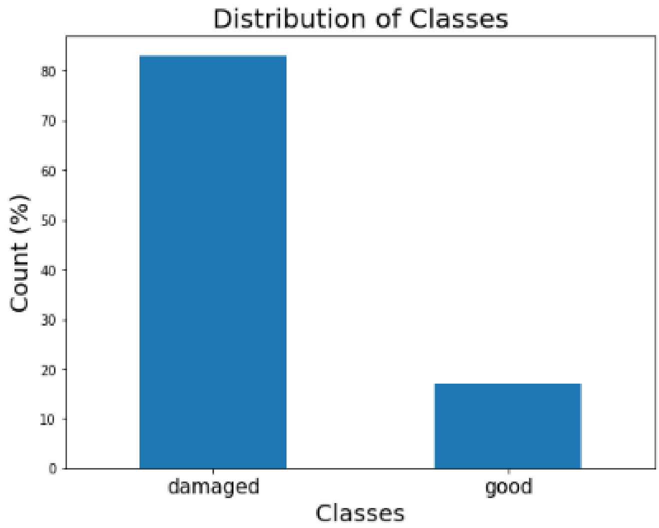

Figure 2. Classes of the Dataset.

3.2 Data Preprocessing

A hybrid CNN+GRU model is being trained and evaluated using the MVTec Anomaly Detection dataset in order to detect textile defects. Various preprocessing approaches are used to improve generalization, boost speed, and make the input consistent before it is represented in the model. Following steps involved in training and evaluating the data preprocessing.

1. Data Cleaning and Formatting:
   - Convert all images to a common file format.
   - Remove damaged or unnecessary files.
2. Resizing:
   - Resize all images to fixed dimensions (64x64 pixels) to match CNN input requirements.
3. Data Augmentation:
   - Apply rotation to increase variety.
   - Use flipping (horizontal/vertical) to simulate different perspectives.
   - Apply zooming to simulate varied distances.
   - Adjust brightness to handle lighting variations.
   - These techniques help improve generalization and enhance defect detection performance, especially for rare defects.

4. Dataset Splitting:

- Split the dataset into training and testing sets.
- Use 80% of the data for training the model.
- Reserve 20% of the data for testing.

5. Purpose of Splitting:

- Training set allows the model to learn from the majority of the data.
- Testing set enables objective evaluation of the model's performance on unseen data.

3.3 Deep Learning Models

A device framework is made up of several layers of artificial neurons called a deep learning model. To generate predictions or judgment, each layer converts the input data into exponentially more complicated representation. In order to reduce errors, deep learning models are trained by adjusting the network's weights and biases. There is some key component of deep learning models such as input layer that accept unprocessed data and then it preprocessed in standardize format. There are certain hidden layers that perform calculations to extract features from the input then produce output layer that involve final analysis and predictions. Weight and biases are the parameter in deep learning model through which model learns to map input to output and then finally measures the difference between the model's predictions and true labels. Optimizer is also one of the key concepts in which optimizer uses the gradients of the loss function to update the weight and biases.

3.4.1 Convolutional Neural Networks

The Convolutional Neural Network model is, in fact, favored among deep learning architectures for the defect detection in textiles because it can automatically extract and learn spatial features from image data. The model generally starts from the input layer, where textile images are resized (e.g., 224x224 pixels) and normalized so that training can be stable and consistent. In the next step, convolutional layers were applied for filtering the images and detecting the important patterns such as edges, textures, or irregularities. They are activated using non-linear activation functions, ReLU, and are followed by pooling layers, for example, Max Pooling, which perform down-sampling to reduce the spatial dimensions of the data while retaining the most salient features. After a specified number of convolutional and pooling layers, the output is flattened to obtain a 1D vector and fed into fully connected (dense) layers for complex representations learning. Dropout layers can also be incorporated during training to avoid overfitting by randomly deactivating some neurons. The output layer usually applies SoftMax or sigmoid activation for classification to detect the images with defects or those that are defect-free. During training, the model takes a labeled dataset of textile images and aims to reduce the loss (e.g., cross-entropy) by backpropagation and optimization algorithms such as Adam. After learning the patterns of defects such as tears, stains, and weaving issues, a well-trained CNN would constitute an excellent alternative for the automatic quality assessment in textile manufacturers.

3.4.2 Artificial Neural Network

Artificial Neural Network (ANN) describes other machine learning techniques that may be applied in detecting defects in textiles since it is more suitable for tabular data or non-image features. In this case, an ANN can directly implement features extracted from the textile images (for example, texture metrics, color histograms, and statistical patterns) rather than raw image features. The architecture typically begins with an input layer that receives these pre-processed features, followed by one or more hidden layers, each continent with several neurons, allowing for the learning of patterns and relationships found in the input data through weighted connections and activation functions such as ReLU or sigmoid. The network utilizes regularization strategies to lesson overfitting and hidden layers to identify intricate patterns that point to flaws. SoftMax or sigmoid activation function functions, or defect classification of multi-class issues are used by the final output layer to forecast the presence or absence of defects.

Table 2. Pseudocode 1: Proposed CNN+GRU Hybrid Model.

```
BEGIN

# Step 1: Preprocessing
Load and preprocess video frames or fabric image sequences
Normalize pixel values (e.g., divide by 255)
Resize images to fixed dimensions
Segment into time sequences if video (optional)

# Step 2: Define CNN for spatial feature extraction
Define CNN_Model:
Input: Single image/frame [height, width, channels]
Apply Conv2D layer(s) with activation (e.g., ReLU)
Apply MaxPooling2D layer(s) to reduce spatial size
Repeat convolution and pooling to deepen network
Flatten or GlobalAveragePooling to reduce feature map to vector
Output: Feature vector for each frame

# Step 3: Extract features for each time step using CNN
For each time_step in input_sequence:
Apply CNN_Model to each image/frame
Collect sequence of feature vectors

# Step 4: Define GRU for temporal pattern recognition
Define GRU_Model:
Input: Sequence of feature vectors from CNN [batch_size, time_steps,
                                    feature_dim]
Apply one or more GRU layers (optionally bidirectional)
Output: Temporal representation (can be last output or sequence)

# Step 5: Fully connected layer and classification
Apply Dense (Fully Connected) layer(s)
Apply Softmax or Sigmoid for final classification based on task

# Step 6: Compile the model
Define loss function (e.g., categorical_crossentropy for multi-class)
Define optimizer (e.g., Adam)
Compile model

# Step 7: Train the model
Input: training_sequences, training_labels
Train model using epochs, batch size, validation set

# Step 8: Evaluate and use for prediction
Evaluate model on test data
Predict defect types or flag defective regions

END
```

3.4.3 Long Short Term Memory

For textile defect identification, LSTM networks operate effectively when examining sequential data, including video frames, scanned fabric lines, or sensor readings. Their proficiency in identifying relationships and temporal patterns in the data is exceptional. To learn long-term associations, the model analyzes sequential inputs through LSTM layers. The result is then mapped to dense layers for categorization. SoftMax is utilized for multi-class issues, whereas sigmoid activation is used for binary jobs. When it comes to time-based defect identification scenarios, such as real-time monitoring or seeing patterns in fabric scanning, LSTMs are perfect.

3.4.4 Recurrent Neural Networks

Recurrent Neural Networks (RNNs) may be used in certain situations to identify textile flaws in various sequences, including sensor output, fabric line scans, and video streams of whole fabric surfaces. To understand temporal patterns and relationships, they sequentially process data, remembering knowledge about previous actions at each stage. An RNN model comprises one or more RNN layers that learn temporal associations when an input layer receives sequential data. Dense layers are used to classify the output, and activation functions like SoftMax for multiclass tasks and sigmoid for binary jobs are used. RNNs are effective at identifying flaws that have developed sequentially, such as a slow texture change or reoccurring flaws.

3.4 Proposed Hybrid Models

For textile defect identification in situations requiring both spatial and sequential data, a CNN+ GRU hybrid model combines the advantages of Convolutional Neural Networks (CNNs) and Gated Recurrent Unit (GRU) networks, making it extremely successful. While the LSTM component records temporal dependencies or sequential patterns, such as slow changes in defects over scanned lines of fabric or video frames, the CNN component is utilized to extract spatial characteristics from pictures or frames, such as textures, patterns, and abnormalities. Usually, the architecture starts with a CNN, from which spatial characteristics are extracted from the input pictures using convolutional and pooling layers. The GRU layers process the temporal connections or sequential patterns after the CNN's output has been flattened or molded into a sequence format. For instantaneous textile monitoring, video-based flaw identification, or any application requiring the simultaneous analysis of spatial and temporal information, this hybrid technique is especially helpful. CNN+GRU models can offer very reliable and accurate defect detection by using the temporal strength of GRU and the spatial capabilities of CNNs.

4. Results and Discussion

For deep learning approaches, well-structured data collection is necessary. Creating a custom dataset according to the necessary fault types is an extra choice. The basic procedures such as rotation, flipping, translation, and random cropping may be applied to existing datasets to improve them and offer additional examples for better deep neural network training stages. Depending on the objective, there are several methods to evaluate the strategy's efficacy, including f1-score, recall, accuracy, and precision.

4.1 Performance of CNN Model

The convolutional neural network model performed exceptionally well at epoch 30, achieving the highest accuracy 0.9689, precision 0.9682, recall 0.9687, and F1-score 0.9681. These results demonstrate the model's ability to extract geographic information from the dataset and generate reliable and accurate predictions. Performance of CNN model are displayed in Table 3.

4.2 Performance of Artificial Neural Network

The Artificial Neural Network (ANN) model performed outstandingly, with the highest accuracy of 0.9322. The model also showed great accuracy (0.9418), recall (0.9322), and F1-score (0.9320) at epoch 30, indicating that it could correctly identify cases while maintaining a balanced trade-off between precision and recall. These results suggest that the ANN model accurately identified the underlying patterns in the dataset, leading to excellent generalization and classification performance. Table 4 show the performance of ANN Model.

Table 3. Performance of CNN Model.

Epoch	Accuracy	Precision	Recall	F1 score
5	0.8532	0.8645	0.8541	0.8513
10	0.8911	0.8915	0.8921	0.8909
15	0.9175	0.9268	0.9171	0.9183
20	0.9489	0.9585	0.9482	0.9491
25	0.9635	0.9629	0.9637	0.9626
30	0.9689	0.9682	0.9687	0.9681

Table 4. Performance of ANN Model.

Epoch	Accuracy	Precision	Recall	F1 score
5	0.7871	0.7664	0.8051	0.7881
10	0.8381	0.8172	0.8469	0.8251
15	0.8661	0.8575	0.8643	0.8556
20	0.8832	0.8734	0.8939	0.8821
25	0.9115	0.9004	0.9132	0.9119
30	0.9322	0.9418	0.9322	0.9320

4.3 Performance of Long Short Term Memory

The Long Short-Term Memory (LSTM) model performed better at 30 epochs, with a maximum accuracy of 0.9423. Furthermore, the model's F1-score of 0.9404, accuracy of 0.9438, and recall of 0.9402 demonstrated a well-balanced classification performance. These results show how effectively LSTM improves prediction accuracy by capturing long-range correlations in the dataset. Table 5 show the performance of LSTM Model.

Table 5. Performance of LSTM Model.

Epoch	Accuracy	Precision	Recall	F1 score
5	0.7035	0.7156	0.7035	0.7094
10	0.7842	0.7983	0.7842	0.7912
15	0.8235	0.8357	0.8235	0.8295
20	0.8528	0.8634	0.8528	0.8581
25	0.8812	0.8923	0.8812	0.8867
30	0.9423	0.9438	0.9402	0.9403

4.4 Performance of Recurrent Neural Networks

The Recurrent Neural Network (RNN) produced the best outcome, with the highest accuracy of 0.9402. The model also demonstrated its strong capacity to recognize patterns and sequential correlations in the dataset by achieving remarkable accuracy (0.9418), recall (0.9402), and F1-score (0.9399) with the use of an epoch of 30. By maintaining a high level of overall performance while finding a balance between recall and precision, these results show how well the RNN does in classification tasks. Table 6 show the performance of RNN Model.

Table 6. Performance of RNN Model.

Epoch	Accuracy	Precision	Recall	F1 score
5	0.6980	0.7351	0.762	0.7482
10	0.7994	0.7891	0.7988	0.7899
15	0.8211	0.8113	0.8117	0.8119
20	0.8634	0.8656	0.8644	0.8639
25	0.9161	0.9188	0.9185	0.9278
30	0.9402	0.9418	0.9402	9.9399

4.5 Performance of Proposed Hybrid Models

The proposed Hybrid CNN+GRU model excelled in the classification task, with a highest accuracy of 0.9841. By attaining an accuracy of 0.9849, a recall of 0.9840, and an F1-score of 0.9841 with 30 epochs, the model proved its robustness and effectiveness in capturing both spatial and temporal interactions within the dataset. The addition of CNN layers enabled efficient feature extraction, while the GRU component's capacity to recognize sequential dependencies led to improved classification performance. These results show that the model is a very reliable approach to the problem at hand because of its excellent generalization across samples. Table 7 show the performance of proposed Hybrid CNN+GRU Models.

Table 7. Performance of Proposed Hybrid Models

Epoch	Accuracy	Precision	Recall	F1 score
5	0.8792	0.8765	0.8823	0.8794
10	0.9152	0.9182	0.9206	0.9194
15	0.9405	0.9386	0.9447	0.9416
20	0.9503	0.9472	0.9534	0.9503
25	0.9631	0.9609	0.9654	0.9631
30	0.9841	0.9849	0.9840	0.9841

4.6 Performance Analysis of All Models

The findings and discussion section examined the performance of several models, including CNN, ANN, RNN, and LSTM, in the given classification task. Each model's accuracy was evaluated, and while conventional architectures like as CNN and LSTM performed well, they did not yield the greatest results. Because it was able to capture the spatial and sequential linkages in the data, the proposed hybrid model which combines CNN + GRU performed better than the others. By combining CNN's feature extraction capabilities with GRU's ability to maintain long-term dependencies, this hybrid approach was able to attain the highest accuracy of 0.9841. This significant improvement shows that convolutional and recurrent architectures may be used to enhance classification performance. Table 8 show the performance of analysis of all employed models.

Table 8. Comparison Analysis of All Models

Epoch	Accuracy	Precision	Recall	F1 score
CNN	0.9689	0.9682	0.9687	0.9681
ANN	0.9322	0.9418	0.9322	0.932
RNN	0.9402	0.9418	0.9402	0.9399
LSTM	0.9423	0.9438	0.9402	0.9403
Proposed Hybrid Models	0.9841	0.9849	0.9840	0.9841

5. Conclusions

The MVTec Anomaly Detection dataset is a benchmark for unsupervised anomaly identification, designed to replicate real-world industrial inspection scenarios. It enables the evaluation of anomaly detection methods across diverse object and texture classes with various types of defects. It is possible to evaluate anomaly detection methods for both image-level classification and pixel-level segmentation when pixel-precise ground truth labels for anomalous regions in the images are provided. Several state-of-the-art methods and two conventional methods were thoroughly evaluated on this dataset, showing that there is still much room for improvement and providing a baseline for this dataset. A few of deep learning models, including CNN, ANN, RNN, and LSTM, were implemented and evaluated. Despite the fact that these models shown significant proficiency in classification tasks, the proposed hybrid model, which combines CNN and GRU, had the highest accuracy of 0.9841. This enhanced performance shows how the model's ability to recognize both temporal and spatial patterns in the input may be enhanced by using convolutional layers for feature extraction and GRU for sequential dependencies. According to the findings, the CNN+GRU hybrid model is a very effective strategy for raising classification accuracy in challenging datasets. In order to improve generalization and robustness, future research might investigate additional optimization hybrid model by adjusting hyperparameters, adding attention mechanisms, and testing with bigger datasets.

References

1. Shahid, Muhammad, et al. "Automatic patents classification using supervised machine learning." *Recent Advances on Soft Computing and Data Mining: Proceedings of the Fourth International Conference on Soft Computing and Data Mining (SCDM 2020), Melaka, Malaysia, January 22– 23, 2020*. Springer International Publishing, 2020.
2. T. Czimmermann et al., "Visual-based defect detection and classification approaches for industrial applications—a survey," Sensors, vol. 20, no. 5, p. 1459, 2020.
3. C. Li, J. Li, Y. Li, L. He, X. Fu, and J. Chen, "Fabric defect detection in textile manufacturing: a survey of the state of the art," *Security and Communication Networks*, vol. 2021, no. 1, p. 9948808, 2021.
4. Sarwar, Ayesha, et al. "IoT networks attacks detection using multi-novel features and extra tree random-voting ensemble classifier (ER-VEC)." *Journal of Ambient Intelligence and Humanized Computing* 14.12 (2023): 16637-16651.
5. Akram, Urooj, et al. "IoTTPS: Ensemble RKSVM model-based Internet of Things threat protection system." *Sensors* 23.14 (2023): 6379.
6. Farooq, Umer, et al. "Time Series Analysis of Solar Power Generation Based on Machine Learning for Efficient Monitoring." *Engineering Reports* 7.2 (2025): e70023.
7. R. Sabeenian, M. Paramasivam, and P. Dinesh, "Detection and location of defects in handloom cottage silk fabrics using MRMRFM & MRCSF," *International Journal of Technology and Engineering System*, vol. 2, no. 2, pp. 172-176, 2011.
8. Ali, Mudasir, et al. "Pneumonia detection using chest radiographs with novel efficientnetv2l model." IEEE Access (2024).
9. R. Divyadevi and B. V. Kumar, "Survey of automated fabric inspection in textile industries," in 2019 International Conference on Computer Communication and Informatics (ICCCI), 2019: IEEE, pp. 1-4.
10. Ali, Mudasir, et al. "An Ensemble Approach for Firewall Log Classification using Stacked Machine Learning Models." *Journal of Computing & Biomedical Informatics* 8.02 (2025).
11. Akram, Urooj, et al. "A comprehensive survey on Pi-Sigma neural network for time series prediction." *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)* 9.3-3 (2017): 57-62
12. Ali, Muhammad, et al. "Elderly fall activity detection using supervised machine learning models." *International Conference on Soft Computing and Data Mining*. Cham: Springer International Publishing, 2022.
13. Shahroz, Mobeen, et al. "Feature discrimination of news based on canopy and KMGC-search clustering." IEEE Access 10 (2022): 26307-26319.
14. Abid, Muhammad Adeel, et al. "Comparative analysis of TF-IDF and loglikelihood method for keywords extraction of twitter data." *Mehran University Research Journal of Engineering and Technology* 42.1 (2023): 88-94.
15. Mushtaq, Muhammad Faheem, et al. "A cognitive architecture for self learning in Humanoid Robots." *International Journal of Computer Science and Network Security* 17.5 (2017): 26â.
16. Asghar, Muhammad, et al. "Finding correlation between content based features and the popularity of a celebrity on Twitter." *International Journal of Computer Science Issues* 11.4 (2014): 177-181.
17. Ali, Mudasir, et al. "Hybrid machine learning model for efficient botnet attack detection in iot environment" *IEEE Access* (2024).
18. P. Anandan and R. Sabeenian, "Fabric defect detection using discrete curvelet transform," Procedia computer science, vol. 133, pp. 1056-1065, 2018.
19. Abid, Muhammad Adeel, et al. "Recommending domain specific keywords for twitter." *Recent Advances on Soft Computing and Data Mining: Proceedings of the Fourth International Conference on Soft Computing and Data Mining (SCDM 2020), Melaka, Malaysia, January 22– 23, 2020*. Springer International Publishing, 2020.
20. Nadeem, Muhammad Waqas, et al. "Fusion-Based Machine Learning Architecture for Heart Disease Prediction." *Computers, Materials & Continua* 67.2 (2021).
21. Karim, Abdul, et al. "Anticipating impression using textual sentiment based on ensemble LRD model." *Expert Systems with Applications* 263 (2025): 125717.
22. Z. Liu, C. Zhang, C. Li, S. Ding, Y. Dong, and Y. Huang, "Fabric defect recognition using optimized neural networks," Journal of Engineered Fibers and Fabrics, vol. 14, p. 1558925019897396, 2019.
23. Shahzad, Faisal, et al. "Improving Queuing System Throughput Using Distributed Mean Value Analysis to Control Network Congestion." *Communications and Network* 7.01 (2015): 21.
24. A. Kayumov, M. Sobirov, and K. Musayev, "Methods of fabric defect detection using expert systems-a systematic literature review," in E3S Web of Conferences, 2024, vol. 538: EDP Sciences, p. 04015.
25. Ishaq, Abid, et al. "Extensive hotel reviews classification using long short term memory." *Journal of Ambient Intelligence and Humanized Computing* 12 (2021): 9375-9385.
26. Iqbal, Umer, et al. "Functional expansions based multilayer perceptron neural network for classification task." *Computación y Sistemas* 22.4 (2018): 1625-1635.
27. M. Li, S. Wan, Z. Deng, and Y. Wang, "Fabric defect detection based on saliency histogram features," Computational Intelligence, vol. 35, no. 3, pp. 517-534, 2019.
28. Mughal, Bushra, Faheem Mushtaq, and Attaullah Buriro. "Classification of Breast Lesions in Combination with Metamorphic Segmentation and Saliency Feature Block." *International Conference on Intelligent Technologies and Applications*. Singapore: Springer Singapore, 2019.

29. S. Shahrabadi, Y. Castilla, M. Guevara, L. G. Magalhães, D. Gonzalez, and T. Adão, "Defect detection in the textile industry using image-based machine learning methods: a brief review," in *Journal of Physics: Conference Series*, 2022, vol. 2224, no. 1: IOP Publishing, p. 012010.
30. Khan, Tareef Ali, et al. "SN2SMS: Location-Aware Social Network Based SMS Health-Care System." *International Journal of Computer Science Issues* 11.4 (2014).
31. H.-w. Zhang, L.-j. Zhang, P.-f. Li, and D. Gu, "Yarn-dyed fabric defect detection with YOLOV2 based on deep convolution neural networks," in 2018 IEEE 7th data driven control and learning systems conference (DDCLS), 2018: IEEE, pp. 170-174.
32. Y. Li, J. Zhang, and Y. Lin, "Combining Fisher criterion and deep learning for patterned fabric defect inspection," IEICE Transactions on Information and Systems, vol. 99, no. 11, pp. 2840-2842, 2016.
33. Zulqarnain, Muhammad, et al. "Efficient processing of GRU based on word embedding for text classification." *JOIV: International Journal on Informatics Visualization* 3.4 (2019): 377-383.
34. Z. Zhu, G. Han, G. Jia, and L. Shu, "Modified densenet for automatic fabric defect detection with edge computing for minimizing latency," IEEE Internet of Things Journal, vol. 7, no. 10, pp. 9623-9636, 2020.
35. I. Koulali and M. T. Eskil, "Unsupervised textile defect detection using convolutional neural networks," Applied Soft Computing, vol. 113, p. 107913, 2021.
36. D. Yapi, M. Mejri, M. S. Allili, and N. Baaziz, "A learning-based approach for automatic defect detection in textile images," IFAC-PapersOnLine, vol. 48, no. 3, pp. 2423-2428, 2015.
37. Amna, et al. "The Comparative Performance Analysis of Clustering Algorithms." *International Conference on Soft Computing and Data Mining*. Cham: Springer International Publishing, 2022.
38. K. Hanbay, M. F. Talu, and Ö. F. Özgüven, "Fabric defect detection systems and methods—A systematic literature review," Optik, vol. 127, no. 24, pp. 11960-11973, 2016.
39. A. Rasheed et al., "Fabric defect detection using computer vision techniques: a comprehensive review," Mathematical Problems in Engineering, vol. 2020, no. 1, p. 8189403, 2020.
40. P. Li, J. Liang, X. Shen, M. Zhao, and L. Sui, "Textile fabric defect detection based on low-rank representation," Multimedia Tools and Applications, vol. 78, pp. 99-124, 2019.
41. L. Tong, W. K. Wong, and C. K. Kwong, "Fabric defect detection for apparel industry: a nonlocal sparse representation approach," IEEE access, vol. 5, pp. 5947-5964, 2017.
42. S. Dlamini, C.-Y. Kao, S.-L. Su, and C.-F. Jeffrey Kuo, "Development of a real-time machine vision system for functional textile fabric defect detection using a deep YOLOv4 model," Textile Research Journal, vol. 92, no. 5-6, pp. 675-690, 2022.
43. A. Kumar, "Computer-vision-based fabric defect detection: A survey," IEEE transactions on industrial electronics, vol. 55, no. 1, pp. 348-363, 2008.
44. J. Silvestre-Blanes, T. Albero-Albero, I. Miralles, R. Pérez-Llorens, and J. Moreno, "A public fabric database for defect detection methods and results," Autex Research Journal, vol. 19, no. 4, pp. 363-374, 2019.
45. J. Zhou, D. Semenovich, A. Sowmya, and J. Wang, "Sparse dictionary reconstruction for textile defect detection," in 2012 11th International Conference on Machine Learning and Applications, 2012, vol. 1: IEEE, pp. 21-26.
46. M. Szarski and S. Chauhan, "An unsupervised defect detection model for a dry carbon fiber textile," Journal of Intelligent Manufacturing, vol. 33, no. 7, pp. 2075-2092, 2022.
47. R. Jin and Q. Niu, "Automatic fabric defect detection based on an improved YOLOv5," Mathematical Problems in Engineering, vol. 2021, no. 1, p. 7321394, 2021.
48. B. Wei, B. Xu, K. Hao, and L. Gao, "Textile defect detection using multilevel and attentional deep learning network (MLMA-Net)," Textile Research Journal, vol. 92, no. 19-20, pp. 3462-3477, 2022.
49. K. Hanbay, M. F. Talu, Ö. F. Özgüven, and D. Öztürk, "Real-time detection of knitting fabric defects using shearlet transform," Textile and Apparel, vol. 29, no. 1, pp. 1-10, 2019.
50. S. Dash, "Defect detection of fabrics by grey-level co-occurrence matrix and artificial neural network," in Handbook of research on modeling, analysis, and application of nature-inspired metaheuristic algorithms: IGI Global, 2018, pp. 285-297.
51. W. Ouyang, B. Xu, J. Hou, and X. Yuan, "Fabric defect detection using activation layer embedded convolutional neural network," IEEE Access, vol. 7, pp. 70130-70140, 2019.
52. P. Bandara, T. Bandara, T. Ranatunga, V. Vimarshana, S. Sooriyaarachchi, and C. De Silva, "Automated fabric defect detection," in 2018 18th International Conference on Advances in ICT for Emerging Regions (ICTer), 2018: IEEE, pp. 119-125.
53. Y. Chen, L. Wu, X. Li, and Y. Zhao, "A High-precision fusion algorithm for fabric quality inspection," DEStech Trans. Comput. Sci. Eng., vol. 102, 2017.
54. Y. Huang, J. Jing, and Z. Wang, "Fabric defect segmentation method based on deep learning," IEEE Transactions on Instrumentation and Measurement, vol. 70, pp. 1-15, 2021.
55. Z. Liu, B. Wang, C. Li, B. Li, and X. Liu, "Fabric defect detection algorithm based on convolution neural network and low-rank representation," in 2017 4th IAPR Asian Conference on Pattern Recognition (ACPR), 2017: IEEE, pp. 465-470.