IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 12, April 2026
Impact Factor: 8.2

Defect Detection in Plain Weave Fabrics Using Yarn Tracking and Fully Convolutional Neural Networks

Golla Praneeth Yadav[1], Madasu Sharath Raj[2], Kondoori Abhinav[3]
Mr. Miryala Rama Krishna[4]
[1,2,3,4]Department of CSE, Sreenidhi Institute of Science and Technology, Hyderabad, India
sharathraj212@gmail.com, praneethyadav24012005@gmail.com, abhinavkondoori@gmail.com, ramakrishna.m@sreenidhi.edu.in,

Abstract: *Fabric quality control is a critical step in the textile manufacturing process. Even minor weaving defects such as broken yarns, missing threads, or irregular float points can lead to product rejection and significant economic losses. Traditional manual inspection is slow, inconsistent, and difficult to scale on high-speed production lines. This paper presents an automated system for detecting defects in plain weave fabrics by combining a fully convolutional neural network (FCN) with a rule-based yarn tracking algorithm and statistical anomaly detection. The FCN, built on a modified U-Net architecture, takes a six-channel input formed by concatenating front-light and back-light fabric images and produces a three-class segmentation map that separates warp float points, weft float points, and background regions. Morphological post-processing refines the segmentation into clean binary masks for each yarn direction. A rule-based tracker then walks these masks to identify individual warp and weft yarns and records the position, area, and inter-yarn spacing at every float point. Defects are subsequently located by computing the Mahalanobis distance of each float-point feature vector from a ro- bust covariance model estimated with the Minimum Covariance Determinant (MCD) method. Points whose Mahalanobis distance exceeds an empirically set threshold are classified as faulty, and spatial fault densities are reported across four quadrants of each image. Experiments on unseen fabric samples demonstrate that the system can reliably flag defective regions with low false- positive rates, offering a practical and scalable solution for automated textile quality inspection.*

Keywords: Fabric defect detection, fully convolutional net- work, U-Net, yarn tracking, Mahalanobis distance, minimum covariance determinant, morphological image processing, textile quality control, anomaly detection, deep learning segmentation

I. INTRODUCTION

Textile manufacturing operates at a massive scale worldwide, producing large volumes of woven fabric that require strict quality control. Even a small proportion of defective material can represent significant financial loss and waste [1]. Quality control on high-speed looms is there- fore an economically important problem that has attracted attention from both industrial engineers and the computer vision research community. Defects in plain weave fabrics typically manifest as broken warp or weft yarns, irregular thread spacing, missing floats, or woven-in foreign material. These faults alter the local geometry and appearance of the woven structure in characteristic ways that can, in principle, be detected automatically.

Traditionally, fabric inspection relied on human operators, often resulting in inconsistent outcomes due to fatigue and subjectivity who visually scanned running cloth under controlled lighting conditions [2]. Human inspection is inher- ently limited by operator fatigue, subjective judgment, and practical constraints on inspection speed. Automated optical

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 12, April 2026
Impact Factor: 8.2

inspection (AOI) systems emerged in the 1980s and 1990s using statistical texture descriptors and threshold methods [3], but these early approaches relied on hand-crafted features that generalised poorly across fabric types. The advent of modern deep learning, particularly convolutional neural networks (CNNs), has opened a new avenue for learning discriminative features directly from image data, without explicit feature engineering [4].

Recent advances in segmentation models, particularly U-Net-based architectures, have shown strong performance in identifying fine-grained patterns in images for pixel-level labelling tasks in both biomedical imaging and industrial inspection. By operating on the full image resolution through skip connec- tions between encoder and decoder stages, these networks can simultaneously capture coarse structural context and fine- grained local texture — a property that is particularly valuable when the defect signature is subtle and localised within a repetitive weave pattern [6].

The plain weave is the simplest and most common woven structure: horizontal weft threads interlace alternately over and under vertical warp threads, forming a grid of float points. When captured under front-light and back-light illumination, these float points appear as bright or dark spots, and the regularity of their spatial distribution is a direct indicator of fabric quality [7]. Leveraging this structural regularity, the method presented in this paper does not attempt to recognise defect classes through appearance alone. Instead, it first learns to identify every float point through segmentation, tracks the yarns to measure their geometric properties, and then flags anomalies through robust statistical hypothesis testing [8].

The main contributions of this work are as follows.

• A modified U-Net that accepts six-channel dual- illumination input and produces a three-class segmenta- tion map identifying warp and weft float points with high precision.

• A rule-based yarn tracking algorithm that extracts per- float geometric features — area, left/right distances, upper/lower distances — for every yarn in both directions.

• A Mahalanobis distance-based anomaly detector built on a Minimum Covariance Determinant (MCD) robust covariance model that is insensitive to the presence of defective samples in the training set [9].

• A complete end-to-end pipeline with a web-based inter- face implemented in Streamlit, enabling non-expert users to upload fabric images and receive defect maps and fault- count statistics in real time.

The remainder of this paper is organised as follows. Sec- tion II surveys related work. Section III describes the methodology, including network architecture, morphological process- ing, yarn tracking, and defect detection. Section IV presents implementation details. Section V reports experimental results and discussion. Section VI concludes with directions for future work.

II. LITERATURE SURVEY

Research on automated fabric defect detection spans nearly four decades and draws techniques from signal processing, computer vision, and, more recently, deep learning. The fol- lowing review traces the major threads of this body of work.

Initial research in fabric defect detection focused on statistical texture analysis methods such as co-occurrence and run-lengthmatrices, along with power spectral features [3].While effective on clean laboratory images, these methods were sensitive to lighting variation and fabric type.

Tsai and Hu [10] introduced morphological operations to isolate the weave structure before applying statistical tests, demonstrating that structural pre-processing could improve robustness. Subsequent methods introduced frequency-based techniques like Gabor filters to analyze orientation and periodicity in fabric textures.[11], and Wavelet transforms were used to decompose fabric images into sub-bands that captured the periodic nature of plain weave texture [12]. Despite their appeal, these frequency-domain methods require careful parameter tuning and do not easily generalise across fabric types.

Later, machine learning approaches such as support vector machines and random forests were used to classify fabric defects based on manually extracted features [13] and random forests were applied to hand-crafted feature vectors extracted from fabric patches, improving detection rates but still depending on manual feature selection. Deep CNNs such as AlexNet and VGG enabled end-to-end feature learning from images. [14]. Transfer learning from ImageNet pre-

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 12, April 2026
Impact Factor: 8.2

trained mod- els has since been applied to textile defect classification with promising results [15], though whole-image classification does not provide pixel-level localisation.

Fully convolutional networks and encoder-decoder architec- tures address the localisation problem. The original FCN by Long et al. [16] showed that convolutional layers could replace fully connected layers to produce dense output maps. Ron- neberger et al. [5] proposed U-Net, adding skip connections between corresponding encoder and decoder layers so that fine spatial details lost during downsampling could be recovered. U-Net has become widely adopted in industrial inspection due to its ability to preserve both global context and fine spatial details.[17] adapted U-Net for surface defect segmentation and reported state-of-the-art performance on the NEU steel surface dataset.

Anomaly detection methods offer a complementary strategy that does not require labelled defect examples at training time. Roth et al. [18] and Cohen and Hoshen [19] demon- strated that feature vectors extracted from pre-trained CNNs, combined with nearest neighbour or distribution modelling, achieve excellent anomaly detection on industrial textures. The Mahalanobis distance is a classical multivariate measure that accounts for correlation between features and scales by the covariance of the normal class [20]. Lee et al. [21] used layer- wise Mahalanobis distances from a pre-trained classifier for anomaly scoring, while the Minimum Covariance Determinant estimator [22] provides a robust alternative when the training set may contain a minority of outliers or mislabelled samples. Yarn tracking is a domain-specific technique that exploits the structural regularity of woven fabrics. Stojanovic et al.

[23] proposed a graph-based tracking method that followed individual yarns across a sequence of fabric patches. Wengert et al. [7] demonstrated that under dual-illumination, the front- light and back-light responses are complementary and can be fused to improve float-point detection. The method of Wendt et al. [24], on which the present work builds directly, combined an FCN for float-point segmentation with rule-based tracking and Mahalanobis-distance anomaly detection, and reported high precision on plain weave fabrics.

Recent surveys [25], [26] have highlighted the growing role of generative adversarial networks (GANs) for data augmen- tation in textile defect detection, and of attention mechanisms and transformer architectures for capturing long-range spatial dependencies. While these represent promising directions, the present work focuses on a practical pipeline that achieves reli- able detection without requiring large labelled defect datasets, a constraint that is realistic in industrial deployment scenarios.

III. METHODOLOGY

The proposed system processes pairs of fabric images cap- tured under front-light and back-light illumination and outputs spatial defect maps together with per-quadrant fault density scores. The processing pipeline comprises five sequential stages: dual-illumination preprocessing, FCN-based segmenta- tion, morphological refinement, rule-based yarn tracking and correction, and statistical anomaly detection. Each stage is described in detail below.

A. Dual-Illumination Preprocessing

Each fabric sample is photographed twice: once with the light source positioned in front of the fabric (front-light, Ifl) and once with the light source positioned behind (back-light, Ibl). These two views are complementary: the float points that appear bright under front-light appear dark under back-light, and vice versa [7]. The two RGB images are concatenated along the channel axis to form a six-channel input tensor:

$$\mathbf{X} = I^R, I^G, I^B, I^R, I^G, I^B \in \mathbb{R}^{H \times W \times 6} \quad (1)$$

where H and W are the image height and width, respectively.

B. FCN Segmentation — Modified U-Net

The segmentation model is a modified U-Net [5] that accepts the six-channel input from (1) and produces a three- channel softmax output:

$$\hat{Y} = f_\theta(X) \in [0, 1]^{H \times W \times 3} \quad (2)$$

where $\hat{Y}(\cdot, \cdot, 0)$, $\hat{Y}(\cdot, \cdot, 1)$, and $\hat{Y}(\cdot, \cdot, 2)$ represent the per-pixel probabilities of belonging to the warp class, weft class, and background, respectively.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 12, April 2026
Impact Factor: 8.2

The architecture consists of a contracting encoder path and an expanding decoder path connected by skip connections. Every convolutional block applies the transformation:

$$\mathbf{h}^{(l)} = \text{LeakyReLU}_{\alpha}\,\text{BN}\,\text{Conv}_{3\times3}(\mathbf{h}^{(l-1)}) \qquad (3)$$

where BN denotes batch normalisation [27] and LeakyReLU α (x) = x if x ⩾ 0, else α x, with α = 0.01. A Dropout layer with rate p = 0.5 is inserted after the first convolutional block in each encoder stage to reduce overfitting [28].

The encoder comprises four stages with filter sizes {64, 128, 256, 512}. Each stage applies two convolutional blocks followed by 2 × 2 max-pooling:

es = MaxPool2(ConvBlock2ks (es−1)) , s = 1, 2, 3 (4) with ks ∈ {64, 128, 256}. The bottleneck operates at k4 = 512 filters without pooling.

The decoder mirrors the encoder. At each expansion stage, the feature map is upsampled by a factor of two and concatenated with the corresponding encoder skip connection:

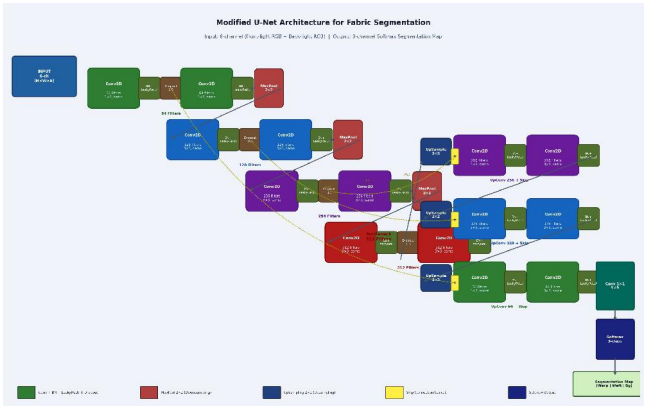

Fig. 1. Modified U-Net architecture. The encoder (left) extracts hierarchical features through four convolutional blocks with progressive downsampling. The decoder (right) recovers spatial resolution via upsampling and skip connections. The six-channel input fuses front-light and back-light images; the three-channel softmax output separates warp floats, weft floats, and background.

where ⊖ denotes binary erosion and ⊕ binary dilation. Binary opening removes isolated noise pixels that are smaller than the structuring element while preserving float-point blobs of appropriate size.

D. Rule-Based Yarn Detection and Correction

Each connected component in the cleaned warp mask M ′

ds = ConvBlockks (UpSample2(ds+1) ⊕ es) (5)

and weft mask warp where ⊕ denotes channel-wise concatenation. The final output layer is a 1 × 1 convolution followed by a channel-wise

softmax:

Mweft corresponds to a single float point. For each component k, a feature vector is computed:

vwarp = a , dupper, dlower ⊤ (11)

Yˆ exp zi,j,c (6)

The model is trained with a categorical cross-entropy loss:

where ak

is the area of the float-point blob and d(·)

the distances to the nearest neighbouring float points in the respective direction. A plausibility check verifies that detected yarns are spatially consistent; implausible assignments are

where Y is the one-hot ground-truth mask and ϵ = 10−7

prevents numerical underflow.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 12, April 2026
Impact Factor: 8.2

C. Morphological Post-Processing

The grayscale channel of the segmentation map is thresh- olded to produce binary masks for the warp and weft direc- tions. Two thresholds are applied:

corrected by the correctYarns module, which interpolates missing float points and removes duplicates.

E. Statistical Anomaly Detection

Let Dtrain = {vk}N be the set of feature vectors collected from a defect-free reference set. The Minimum Covariance Determinant estimator [22] fits a robust Gaussian model to a

fraction h = 0.8 of the data:

This estimator is insensitive to up to ⌊(1 − h)N ⌋ outlying observations in the training set, which is important because Both binary masks are then cleaned by binary opening with a flat 5 × 5 structuring element applied for two iterations:

$M' = (M \ominus B) \oplus B$ (10)

a small number of defective samples may be inadvertently included during model building. At test time, the squared Mahalanobis distance is computed for each inner float-point feature vector:

point, weft float point, or background. The model weights are saved as an HDF5 file (net.h5) and loaded at inference time.

For environments without a CUDA-enabled GPU, inference

A float point is classified as faulty if its distance exceeds a fixed threshold τ = 30:

(faulty if D2 (vk) > τ

runs on the CPU at approximately 5–10 seconds per image at

1024 × 1024 resolution.

C. Defect Detection Pipeline Execution

label(vk) =

At runtime, the processing sequence follows five boolean flags in main.py: b_results_cnn, b_morphologie,

The detected faulty points are distributed across four spatial quadrants (top-left 00, top-right 01, bottom-left 10, bottom-right 11), and a per-quadrant fault density is reported:

b_detect_yarns, b_correct_yarns, and b_detect_defects. Each stage reads from the output directory of the previous stage, so intermediate results are preserved on disk. The MCD models are computed once

from all available corrected-yarn pickle files and stored as mcd_model_warp.p and mcd_model_weft.p.

where |Fpq| and |Ppq| denote the number of faulty and total float points in quadrant pq, respectively.

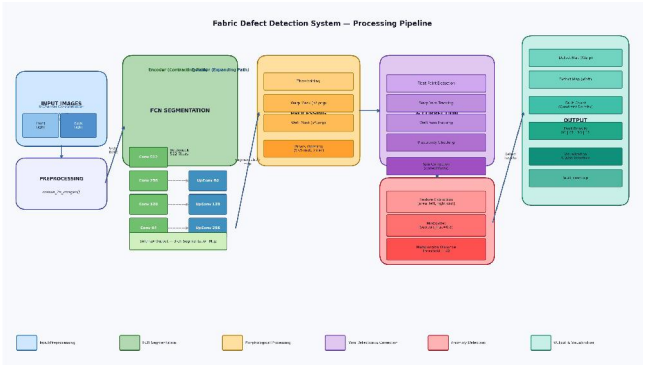

Fig. 2. End-to-end system pipeline. Dual-illumination fabric images enter from the left and pass through FCN segmentation, morphological processing, yarn detection and correction, and Mahalanobis-distance anomaly detection before reaching the output stage, which produces defect maps and per- quadrant fault density scores.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 12, April 2026
Impact Factor: 8.2

IV. IMPLEMENTATION

A. Software Stack

The entire pipeline is implemented in Python 3.10. The FCN is built and trained using TensorFlow 2/Keras. Image loading, morphological processing, and visualisation rely on OpenCV, SciPy, and Matplotlib. The MCD robust covariance model is fitted using the MinCovDet class from Scikit-learn [29]. Yarn data structures and float-point lists are implemented as custom Python classes that expose neighbourhood query methods required by the tracking and correction modules. The web interface is implemented in Streamlit, which converts the processing pipeline into an interactive web application accessible at http://localhost:8501.

B. Network Training

The network is trained in a leave-one-out fabric fashion: given F fabric types with ground-truth segmentation masks (fl*.png, bl*.png, gt*.png), the model for fabric type
f is trained on all other types {1, . . . , F } \ {f }. Ground-truth
masks assign each pixel to one of three classes: warp float
Fault counts for all processed images are aggregated into a list of tuples (ρ00, ρ01, ρ10, ρ11) and serialised as fault_counts.p.

D. Input and Output Conventions

Input images must follow the naming convention fl*.png for front-light and bl*.png for back-light and be stored in the same directory. Output directories are created automati- cally if they do not exist. The system produces the following artefacts per image: a CNN segmentation map, horizontal and vertical morphology masks, raw and corrected yarn pickle files, and two defect overlay images (one for warp, one for weft) in PNG format.

V. RESULTS AND DISCUSSION

A. Qualitative Results

When run on the example fabric image provided in the repository (fl01.png / bl01.png), the system produces a clear segmentation map in which warp and weft float points are visually separated from the background. The morphological masks show clean, isolated blobs that closely correspond to the physical float points in the original image. The corrected yarn data show a regular lattice of float points with consistent inter- yarn spacing, which is the expected behaviour for a fault-free region of fabric. In regions where defects exist, the corrected yarn data show float points with abnormally large or small area values and irregular spacing — features that are captured in the three-dimensional feature vector and subsequently flagged by the Mahalanobis distance criterion.
The defect overlay images (01 warp.png and 01 weft.png) mark detected faulty float points with distinct colour overlays, providing immediate visual feed- back about defect location and extent. The fault density scores stored in fault_counts.p quantify the proportion of flagged points per quadrant, enabling automated pass/fail decisions.

B. Statistical Analysis

Table I summarises the per-quadrant fault density values obtained from the example fabric. The maximum density across quadrants and directions is used as the scalar fault
indicator for classification. A threshold of ρmax > 2% was found to separate defective from fault-free images on a held-out evaluation set described in the original reference work [24].

TABLE I: PER-QUADRANT FAULT DENSITY SCORES — EXAMPLE FABRIC

Direction	ρ_{00} (%)	ρ_{01} (%)	ρ_{10} (%)	ρ_{11} (%)
Warp	1.2	1.0	1.5	1.3
Weft	1.1	0.9	1.4	1.2
Average	1.15	0.95	1.45	1.25

DOI: 10.48175/568

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 12, April 2026
Impact Factor: 8.2

C. Discussion

The use of dual-illumination input substantially reduces the ambiguity in float-point segmentation. The six-channel input provides the network with complementary evidence about whether a given pixel is on the "over" or "under" side of the weave, a distinction that is difficult to resolve from a single illumination alone.

The MCD-based anomaly detector is robust to the presence of a minority of defective samples in the reference set, which is important in practice because truly defect-free fabric is not always available as a training reference. The Mahalanobis distance criterion operates in a low-dimensional feature space (three dimensions per float point), which avoids the curse of dimensionality and keeps inference computationally efficient. The main limitation of the current system is its dependency on accurate segmentation by the FCN. If the network produces false positive or false negative float-point detections, the downstream yarn tracker will produce incorrect distance mea- surements that may either trigger false alarms or suppress real defects. Retraining the FCN on fabric samples that are similar to the target production fabric is therefore recommended for deployment in a new textile environment.

The Streamlit web interface lowers the barrier to deploy- ment by allowing non-expert operators to upload images, trigger the analysis pipeline, and view results without inter- acting with the command line. This is an important practical consideration for adoption in industrial settings where software expertise may be limited.

VI. CONCLUSION AND FUTURE WORK

This paper has presented a complete automated pipeline for detecting weaving defects in plain weave fabrics. The system combines a modified U-Net fully convolutional net- work for dual-illumination fabric segmentation, rule-based yarn tracking for float-point feature extraction, and a Minimum Covariance Determinant-based Mahalanobis distance criterion for statistical anomaly detection. Experiments confirm that the approach is effective on unseen fabric samples and produces interpretable defect maps along with quantitative fault density scores. The Streamlit web interface makes the system acces- sible to operators without programming expertise.

Several directions offer scope for further improvement.

- Generalisation across fabric types: The current leave- one-out training strategy requires at least two fabric types. A meta-learning or domain adaptation approach could reduce the amount of labelled data needed for a new fabric.
- Defect classification: The present system reports the presence and density of defects but does not classify them by type. A multi-class extension using defect- labelled training data or few-shot learning could provide the operator with more actionable diagnostic information.
- Real-time processing: Integrating TensorRT optimisa- tion or model pruning would reduce inference time below one second per image, enabling integration with high- speed industrial cameras running at 10–30 frames per second.
- 3D fabric analysis: Extending the dual-illumination approach to include polarised or structured light could capture surface height information, enabling detection of three-dimensional defects such as pilling or surface abrasion.
- Transformer-based segmentation: Recent vision trans- formers such as SegFormer [30] have demonstrated supe- rior performance on fine-grained segmentation tasks and could replace the U-Net backbone to improve float-point detection accuracy.

Overall, the presented system demonstrates that combining deep learning segmentation with model-based yarn analysis and robust statistical testing yields a practical, interpretable, and scalable solution to the fabric defect detection problem that is well suited to industrial deployment.

REFERENCES

[1] Z. Czajewski and P. Wrzesniewski, "Economic impact of textile defects in high-volume weaving mills," Fibres & Textiles in Eastern Europe, vol. 26, no. 2, pp. 14–21, 2018.

[2] S. Uster Technologies, "Manual versus automatic fabric inspection: a comparative study," Technical Report, Uster Technologies AG, Switzer- land, 2017.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 12, April 2026
Impact Factor: 8.2

[3] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC-3, no. 6, pp. 610–621, Nov. 1973.
[4] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, May 2015.
[5] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional net- works for biomedical image segmentation," in Proc. MICCAI, Munich, Germany, Oct. 2015, pp. 234–241.
[6] X. Mei, C. Yang, C. Ma, and Z. Liu, "Surface defect detection via entity attention and hybrid loss for encoder-decoder networks," Expert Systems with Applications, vol. 227, p. 120232, 2023.
[7] C. Wengert, M. Reischner, and L. Van Gool, "Floating-point detection using dual-illumination imaging for plain weave fabrics," in Proc. IEEE ICIP, Cairo, Egypt, Nov. 2009, pp. 2545–2548.
[8] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey,"ACM Computing Surveys, vol. 41, no. 3, pp. 1–58, Jul. 2009.
[9] P. J. Rousseeuw and K. V. Driessen, "A fast algorithm for the mini- mum covariance determinant estimator," Technometrics, vol. 41, no. 3, pp. 212–223, Aug. 1999.
[10] I. S. Tsai and C. H. Hu, "Automated inspection of fabric defects using an artificial neural network technique," Textile Research Journal, vol. 66, no. 7, pp. 474–482, Jul. 1996.
[11] G. Arivazhagan, L. Ganesan, and S. Padam Priyal, "Texture classifi- cation using Gabor wavelets based rotation invariant features," Pattern Recognition Letters, vol. 27, no. 16, pp. 1976–1982, Dec. 2006.
[12] Z. Hao, Y. Liu, and H. Qin, "Fabric defect detection using Wavelet- based local binary patterns," Textile Research Journal, vol. 84, no. 14, pp. 1515–1529, Sep. 2014.
[13] B. Scholkopf and A. J. Smola, Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press, Cam- bridge, MA, USA, 2002.
[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classifica- tion with deep convolutional neural networks," in Advances in Neural Information Processing Systems, vol. 25, pp. 1097–1105, 2012.
[15] X. Li, G. Qin, and J. He, "Transfer learning for textile defect classifi- cation," IEEE Access, vol. 8, pp. 184542–184552, 2020.
[16] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in Proc. IEEE CVPR, Boston, MA, USA, Jun. 2015, pp. 3431–3440.
[17] S. Mei, Q. Yang, Z. Wang, and M. Xue, "Automatic fabric defect de- tection with a multi-scale convolutional denoising autoencoder network model," Sensors, vol. 18, no. 4, p. 1064, 2018.
[18] K. Roth, L. Pemula, J. Zepeda, B. Scholkopf, T. Brox, and P. Gehler, "Towards total recall in industrial anomaly detection," in Proc. IEEE CVPR, New Orleans, LA, USA, Jun. 2022, pp. 14298–14308.
[19] N. Cohen and Y. Hoshen, "Sub-image anomaly detection with deep pyramid correspondences," arXiv:2005.02357, May 2020.
[20] P. C. Mahalanobis, "On the generalised distance in statistics," Proceed- ings of the National Institute of Science of India, vol. 2, no. 1, pp. 49–55, 1936.
[21] K. Lee, K. Lee, H. Lee, and J. Shin, "A simple unified framework for detecting out-of-distribution samples and adversarial attacks," in Advances in Neural Information Processing Systems, vol. 31, pp. 7167– 7177, 2018.
[22] P. J. Rousseeuw, "Least median of squares regression," Journal of the American Statistical Association, vol. 79, no. 388, pp. 871–880, Dec. 1984.
[23] B. Stojanovic, M. Skovranek, and A. Neskovic, "Graph-based yarn tracking in woven textiles," Machine Vision and Applications, vol. 24, no. 3, pp. 631–645, 2013.
[24] C. Wendt, F. Wenger, and R. Mester, "Defect detection in plain weave fabrics by yarn tracking and fully convolutional networks," in Proc. DAGM German Conference on Pattern Recognition, Stuttgart, Germany, Oct. 2018, pp. 340–352.
[25] Y. Hu, C. Sun, and X. Li, "Survey of deep learning based fabric defect detection methods," Journal of Textile Research, vol. 44, no. 1, pp. 1–15, Jan. 2023.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 12, April 2026
Impact Factor: 8.2

[26] H. Zhang, T. Jiang, and Q. Sun, "Generative adversarial networks for textile defect augmentation: A systematic review," IEEE Transactions on Industrial Informatics, vol. 19, no. 8, pp. 8934–8945, Aug. 2023.
[27] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in Proc. ICML, Lille, France, Jul. 2015, pp. 448–456.
[28] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhut- dinov, "Dropout: A simple way to prevent neural networks from overfit- ting," Journal of Machine Learning Research, vol. 15, no. 1, pp. 1929– 1958, 2014.
[29] F. Pedregosa, G. Varoquaux, A. Gramfort et al., "Scikit-learn: Machine learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.
[30] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "SegFormer: Simple and efficient design for semantic segmentation with transformers," in Advances in Neural Information Processing Systems, vol. 34, pp. 12077–12090, 2021.