DETECTION OF FABRIC DEFECTS BASED ON IMPROVED GAN USING CNN ALGORITHM

Mrs. I.Varalakshmi,M.Tech(PhD)[1], G. Megha[2], J. Sowmya Deborah[3]

Article History: Received: 14.03.2023 **Revised:** 30.04.2023 **Accepted:** 14.06.2023

Abstract:

Fabric defect detection is crucial for maintaining the quality and integrity of textile products. Manual inspection methods are time-consuming and subjective, driving the need for automated solutions. The existing model uses a GAN with a dual-discriminator architecture to detect fabric defects. The limitations in this model: increased complexity, longer training times, and higher resource requirements. To address these challenges, an alternative approach is proposed using a single-discriminator GAN, reducing training time and cost while maintaining effectiveness in defect detection. Our system aims to develop a fabric defect detection using improved GAN, this will be reconstructing the fabric images in an unsupervised manner and locate the defect areas by finding differences of original image and the reconstruction. This system consists of the components: generator network and discriminator network. The generator network generates fake images by using Gaussian distribution random noise technique; the discriminator network learns to distinguish between real defect-free fabric images and fake images. The CNN algorithm is applied for training the input images and generate the fake images by finding the parameters like dense, reshape, leakyReLU, Batch Normalization, flatten and upsampling. The performance metrics like accuracy, precision, recall and F1 score are calculated and tested as results are discussed.

Keywords: Fabric Defect Detection, Improved GAN, Deep learning, Training, Image segmentation.

[1]Assistant Professor Dept. of Computer Science and engineering Manakula Vinayagar Institute of Technology ,Puducherry, India
[2,3]B. Tech, Dept. of Computer Science and engineering, Manakula Vinayagar Institute of Technology ,Puducherry, India

Email: [1]varalakshmicse@mvit.edu.in, [2]gnanasekaranmegha@gmail.com

DOI: 10.31838/ecb/2023.12.s3.484

1. Introduction

In recent years, artificial intelligence (AI) has enhanced the integration of the technology into traditional industries and evolved into a wide range of applications. Computer Vision is the technology employed in our model. In computer vision, relevant knowledge is extracted from images, videos, and other optical inputs and then acted upon or suggested [1].

The textile industry has placed a significant emphasis on the identification of fabric flaws in recent years [2]. In order to guarantee the consistently high quality of finished textile goods produced in industrial settings, it is necessary to classify imperfections in the fabric [3]. Because of their high accuracy, rapid detection speed, and low labor costs, machine vision-based methods are gradually becoming a trend in the textile industry as a result of the development of machine learning and computer vision technology [4]. These methods are used to solve problems related to quality control in textile production.

A generalized additive network, or GAN, typically involved two neural networks: a generator and a discriminator. An input vector is provided to the generator, which attempts to create a false sample that resembles the original sample [5]. The discriminator, on the other hand, has been trained with both real and fake samples, allowing it to distinguish between real and fake samples [6]. There is a zero-sum game, adversarial, between the two parts until the discriminator version is fooled to a certain extent to 1/2 of the time, which means the generator version is producing potential examples.

As an unsupervised mastering challenge, generational modelling is an integral part of device learning, which involves routinely encountering and understanding routinely encountering and learning to use the version to generate or new examples from the unique dataset that could be plausibly derived [7]. GANs are a thrilling and unexpectedly converting field, handing over at the promise of generative fashions of their cap potential to generate sensible examples throughout quite a number of trouble domains, maximum significantly in picture-to-picture translation responsibilities along with translating pix of summer time season to wintry weather or day to night, and in producing photorealistic pix of articles [8], divisions, and those that even human beings can't inform are fake. The general structure of GAN is shown in fig 1.

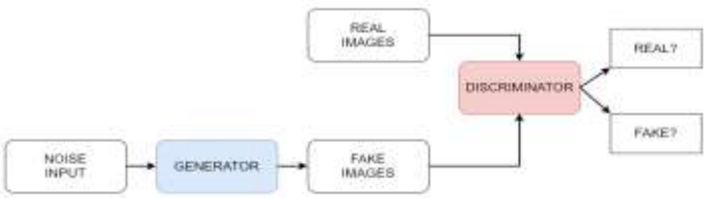


Fig1: Structure of Generative Adversarial Network

The discriminator takes all the benefits obtained in the discriminant model and acts as an adaptive loss function over the GAN. This means that the discriminator can adapt to the underlying data distribution [9]. This is one of the reasons why current discriminative deep learning models are so successful today. Past techniques have overly relied on the direct computation of some heuristics about the underlying data distribution [10].Ultimately, the discriminator evaluates the reliability of the actual image output and the caused image. Real images initially score high on the scale, while generated images score low [11].

Eventually, the discriminator will have difficulty distinguishing among the generated image and the actual image. Discriminators rely on model building and possibly an initial loss function [12].

Literature Survey

We present a novel approach for fabric defect detection at the pixel level, termed RPDNet (Repeated Pattern Defect Network), which employs a semantic segmentation network combined with a repeated pattern analysis algorithm. Specifically, our method utilizes a convolutional neural network (CNN)-based repeated pattern detector to identify periodic patterns in fabric images.[10] To address the challenges of sample imbalance and computational complexity, we propose an efficient convolutional neural network called Mobile-Unet[12]. This network architecture incorporates depth-wise separable convolution and employs the median frequency balancing loss function for effective end-to-end defect segmentation. These advancements result in a significant reduction in network complexity and model size.

In order to accurately detect various defects in the fabric production process, we propose an optimized fabric defect detection algorithm based on feature fusion between a convolutional neural network (CNN) and an extreme learning machine (ELM). By leveraging the strengths of both techniques, we achieve improved defect detection performance. [20] Furthermore, we enhance the Faster R-CNN algorithm and introduce a surface defect detection algorithm based on the improved Faster R-CNN. Our approach employs the Region of Interest Align (ROI Align) algorithm, which provides more accurate defect localization information compared to the conventional Region of Interest Pooling (ROI Pooling) algorithm.

To address the task of surface defect detection, we leverage the power of deep learning (DL) and computer vision (CV) techniques. By utilizing a Convolutional Neural Network (CNN), we achieve robust detection and recognition of various defects, thereby improving production standards and process efficiency [21]. In addition, we propose a hybrid fabric defect detection method that combines a Convolutional Neural Network (CNN) with a Variational Autoencoder (VAE). The CNN is utilized for extracting fabric image pattern features, while the VAE models the latent characteristics and facilitates reconstruction inference, resulting in a powerful fabric defect detection approach. [23]Deep Convolutional Neural Networks (CNNs) have demonstrated significant advancements in target detection, and their application to fabric defect detection has yielded notable results, showcasing the potential of deep CNN models in this domain.

Defect detection can be divided into two categories: traditional methods and deep learning methods [26]. In the field of machine vision, the automatic detection of texture defects poses a challenge due to the wide range and complexity of texture defects. In specifically, Seg-Net uses a decoder that is made up of numerous up-sampling blocks and a concluding soft-max layer to convert the low-resolution image representation that is created by the encoder to pixel-wise prediction [22]. Training multi-stage GAN is harder and leads to high time consumption. Each up-sampling block consists of an up-sampling layer and multiple sequential convolutional layers, which maximum pooling indices match to the pooling layers in the coding step [24].

Conventional methods of oversampling are based on the Synthetic Minority Oversampling Technique (SMOTE), which places an emphasis on local information but provides data that is inadequately representative of the actual world. On the other hand, the Generative

Adversarial Network, often known as GAN, is able to produce data for the minority class by accurately capturing the distribution of the data [25]. Due to mode collapse and unstable training, both techniques have drawbacks, however. The challenge of detecting and localizing surface defects in structured materials is an important one, but it is also a difficult one to solve because of variables such as the fluctuation in the texture of the material and the absence of adequate faulty samples before testing [26]. The use of deep learning technology to the identification of fabric flaws has been shown to be feasible; however, the detection effectiveness is dependent on the use of a large labeled training set [27].

This model's objective is to generate genuine flaws across all of the fabrics so that they may be used as the foundation for pseudo-supervised learning [28]. Due to the limited size of the datasets, however, there was no attempt made to conduct a direct comparison with fully supervised anomaly detection. CycleGANs were able to overcome this constraint, which meant that the conditional GAN (cGAN) model was no longer restricted to just using paired observations across the source and target domains [29]. We have created an image dataset containing three classes of lychees, namely mature, faulty, and rotten, in order to investigate the possibility of automating the process of detecting problematic surfaces on lychees [30]. In addition, we trained a transformer-based GAN to produce synthetic samples that can successfully enhance the size and variety of the original training set [31]. This was accomplished by training the GAN using a set of synthetic samples that were generated by the original training set.

2. Our Proposed methodology

Focusing on a predetermined GAN, this graphic depicts the suggested pipeline for identifying fabric problems. The improved GAN shows a combination of a Generative Adversarial Network and an encoder. We use a GAN to reconstruct the input image and an encoder to detect specific error detection. Due to issues such as lack of error samples, the enhanced GAN learns to reconstruct the fabric image unattended. The output we get from the enhanced GAN is the reconstructed image. Comparing traditional tissue defect detection with the model, the enhanced GAN receives an extensive collection of normal tissue images. There are some architectural changes proposed in the generator, such as removing all fully connected layers and using batch normalization to help stabilize training.

According to the diagram, the random noise input for the generator in a GAN is typically drawn from a probability distribution, such as a Gaussian distribution (also known as a normal distribution) or a uniform distribution.

The formula for the Gaussian distribution is as follows:

$$f(x) = (1 / \mathrm{sqrt}(2 * \pi * \sigma^2)) * \exp(-(x - \mu)^2 / (2 * \sigma^2))$$

- $f(x)$ represents the probability density function (PDF) of the Gaussian distribution at a given value x.μ (mu) represents the mean of the distribution, its value is set to 0 indicating the centre of distribution.σ (sigma) represents the standard deviation of the distribution, its value is set to 1 for controlling the spread of values.π (pi) is a mathematical constant representing the ratio of a circle's circumference to its diameter (approximately 3.14159).exp denotes the exponential function.

These random noise is passed into the transposed convolutional layer to transform it from low-dimensional random noise into a higher-dimensional representation which can be reshaped into an image-like structure. The transformed representation is processed by layers that extract features and upsample the representation. The upsampled features are further processed by additional layers to reconstruct the final generated image. These layers refine the features and ensure that the generated output resembles the desired output distribution. Techniques such as skip connections, residual connections, or attention mechanisms may be employed to enhance the quality and fidelity of the generated images. The final generated image is passed through an activation function, such as a sigmoid or a tanh function, to ensure that the pixel values are within the desired range (e.g., 0 to 1 or -1 to 1). This generates the fake image output of the generator.

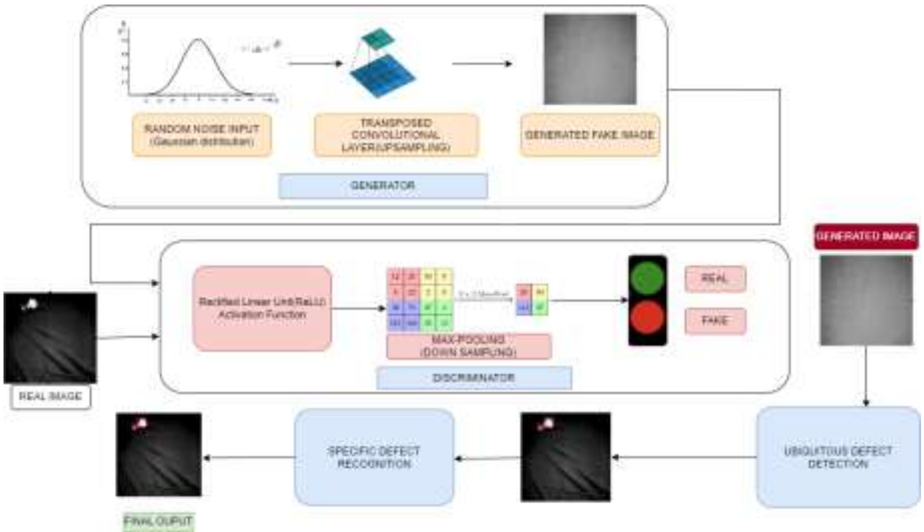


Fig2: Structure of the proposed method.

The discriminator takes an input sample, which can be a real image from the training dataset or a generated/fake image produced by the generator.

Rectified Linear Unit (ReLU) which is a Non-Linear activation functions is applied after the convolutional layers to introduce non-linearity into the discriminator's decision making process.

The ReLU function applies the following formula:

$$f(x) = \max(0, x)$$

In this formula:

- f(x) represents the output of the ReLU function for a given input x.x represents the input value to the ReLU function.

Max pooling, which is a pooling operation is applied to reduce the spatial dimensions and capture the most important information. Typically, max pooling is performed over a fixed window (2x2 or 3x3) with a specified stride (stride of 2). The window slides over the input feature map, and at each position, the maximum value within the window is selected. Max pooling helps in achieving translation invariance and spatial hierarchies by preserving the dominant features and reducing the computational complexity of subsequent layers.

$$\text{MaxPooling}(x) = \max(x)$$

In this formula:

- MaxPooling(x) represents the output of the max pooling operation for a given input x.max(x) computes the maximum value within the local neighborhood.

Then the processed features are then flattened or pooled and passed through one or more fully connected layers. These fully connected layers act as classifiers, making a final decision about whether the input sample is real or fake/generated. Typically, a sigmoid activation function is used in the last layer to produce a probability score between 0 and 1, where values closer to 1 indicate a prediction of "real" and values closer to 0 indicate a prediction of "fake/generated."

After the GAN generates the trained image, the difference image is taken between the input image(real image) and generated image of GAN using the pixel wise comparison technique, which involves calculating the absolute difference or squared difference between the pixel values of each corresponding pair of pixels.

The below minimax formulation may be used to represent GAN's primary goal function:

$$\min G \max D \; V(D_1, G_1) = E_{a\sim pdata(x)}[\log D(a)] + E_{a\sim pz(a)}[\log (2 - D_1(G_1(x)))] \quad (1)$$

According to Equation 1, the regression model is made up of two different parts: the expectations for the log probability density of the generators as well as the discriminator, and the actual values of both functions. a is the actual sample, z is the random noise along with the input to the generator, G1(x) is made up of the samples that were created, p is the representation of the data distribution, and E is the predicted distribution. As a general guideline, generators are developed with the goal of maximizing the value of D1(G1(z)) to the maximum degree feasible, i.e., reducing the value of V(D1,G1). On the other hand, the discriminator is designed to make D(a) as broad as possible while keeping D1(G1(x)) as little as possible. This will result in V(D1,G1) being as huge as it possibly can be.

Furthermore, As surface and defective forms vary between textiles, it is challenging for the networks to determine the common characteristics of all fabric textures. Pix2pixGAN, which is learned from pairs of images, is ideal for generating fabric defects in this respect. The preceding minimax formulation may be used to define the fundamental goal function of pix2pixGAN:

$$L_{L1GAN}(D_1, G_1) = E_{a,b}[\log D(a)] + E_{a-x}[\log (2 - D_1(a, G_1(x)))] \quad (2)$$

$$L_{L1GAN}(g) = E_{x,y'}\, x[\| y - G(x,z) \|_1] \quad (3)$$

$$G^* = \arg \min G \max D \; L_{CGAN}(G,D) + \gimel\, L_{L1}(G) \quad (4)$$

An input image is provided by ‘x’ and an actual image is provided by y. Pairing of images (a, b) is required throughout the pix2pix training process. G1(x) stands for the produced image, D stands for the discriminator, and z stands for the random noise.

Post-processing is the use of techniques and techniques to enhance the original image captured by the photographer. Post-processing approaches often involve pruning processes, rules filtering, and possibly knowledge integration. Many of these strategies offer some form of conceptual filter for recursion algorithms' noisy and inaccurate knowledge. Using several images, do post-processing to find common faults and identify faulty regions.

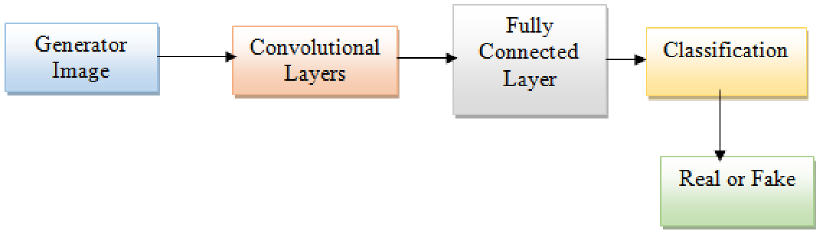


Fig3: Structure of Image Reconstruction

Figure 3 structure of image reconstruction in which it constitutes 2 separate models such as generator and discriminator. The generator is incorporated with a set of input images and which is used to generate a fake image. It also has either one or multiple hidden transposed convolution layer. The generated fake image is incorporated as an input data to the Discriminator module, in which it constitutes of several hidden convolutional layers. The discriminator helps in validating the fake image with the real image

A. Detection Model:

It is generally thought that there are two types of target detection algorithms in use today: one-stage and two-stage algorithms.

B. Ubiquitous Defect Detection:

The first step is to detect unattended and ubiquitous defects using improved GANs. In the training phase, the GAN takes the regular fabric image is reconstructed after training using the input image as input. Here, networks show two neural networks that exist within the GAN architecture. GAN is not capable of reconstructing both defective and good images in this case. The reconstructed image differs significantly from the input error imagecan be observed during the test phase.

Here are the steps involved in renovating the fabric images using the improved GAN:

(1) Firstly, G_E (encoder) takes real image GD outputs the reconstructed sample x' based on the input x and the latent variable z.

$$X' = GD\ (z, WGD) = GD\ (GE(x; WGE)) \qquad (5)$$

where W_{GD} and $W_{GE}$is the weights that can be learned of G_D and G_E, respectively.

(2) As a result, the discriminator D receives the real images x and recreated samples x', and determines which sample is the real one as follows:

$$Y = D(x,x'; WD) \qquad (6)$$

A discriminator D produces an output y, and WD represents the parameter that can be learned by it.

The reconstructed sample x' is re-encoded as x' using an additional encoder E as follows:

$$Z' = E(x'; WE) \qquad (7)$$

Here W_E represent A parameter that can be learned by the decoder E and recycled to compress the reconstructed image x'.

In particular, E is reduced in order to determine its feature representations. The size of this is the same as the size of x' for consistency.

In order to enhance the system in a weighted manner, the bettered GAN is trained with a common loss function. Following is a description of the loss function

a. **Adversarial loss:** In the process of creation and discrimination, there is an adversarial loss between the creator and the discriminator. Calculation of this loss is as follows:

$$LA = \| f(x) - f(G(x)) \| 2(i) \qquad (8)$$

b. **Content loss**: Regarding content loss, It is referring to the difference that exists between the reconstructed view x' and the first image x. The deviation may be determined using the following formula:

$$LC = \| x - G(x) \| 1 \qquad (ii) \qquad (9)$$

c. **Encoding loss:** To produce better normal samples while confining imperfect bones, as part of the encoding process, a loss is used to achieve the goal of minimising the gap in distance between the features of the input data and the decoding properties of the samples that are produced. The calculation for the encoding loss looks like this when using the L2 loss.

$$LE = \| GE(x) - E(G(x)) \| 2 \quad (iii) \quad (10)$$

(3) Following is a weighted definition of joint loss:

$$LE = WA(LA) + WC(LC) + WE(LE) \qquad (11)$$

where W_A, W_C, and W_E are the Weighed parameters are used in order to modify the impact of separate losses on an whole objective function. By cross-validating public datasets, these parameters were determined to be 0.2, 0.7, and 0.1, respectively. In post-processing the fabric image, ubiquitous defects are detected and the total defect area is calculated based on the real and reconstructed images x and x'.

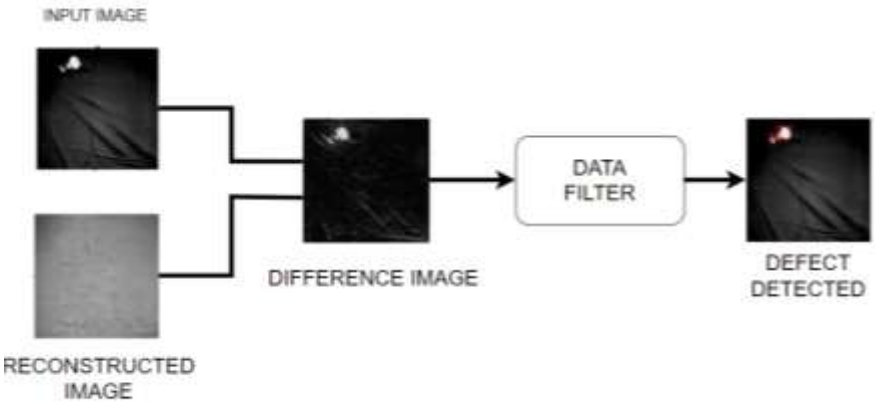


Fig4: Ubiquitous Defect Detection Flow Chart

The distinction as seen in figure 4, image d may be calculated by subtracting x' from x. To eliminate low-noise zones, median and morphological filtering are done to d. The filtered d is then subjected to a connected component analysis. If the connection area overdoes a specific level, x is deemed a poor sample. The identification of defect-related characteristics is guided by a coarse defect region. To boost detection performance, a central loss limitation is imposed.

C. Specific Defect Recognition:

For specific error discovery, birth of error- related features is considered important when the error region is not large enough. Thus, in order to ameliorate the discovery performance, a center- loss limitation is presented to ameliorate the discrimination power of the learned features and it's represented in the below figure 5.

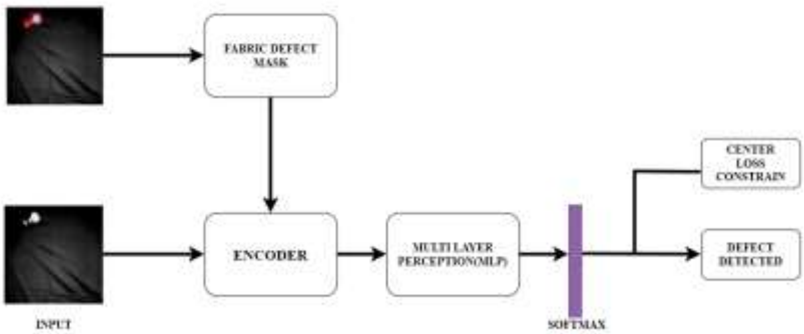


Fig5: Flowchart for recognizing specific defects

Since there are ubiquitous blights in the first- stage towel image, we zoom in on the imperfect regions by a factor of1.5 to induce a towel disfigurement mask FM containing possible disfigurement regions and three-dimensional environment information. It is necessary to determine the size of the point maps generated by the GE after the organization of the GE and the extent of the original image has been determined.

These article charts contain pictorial features of the towel from which we'd like to prize disfigurement- related features. thus, we custom the towel disfigurement mask FM as a guideline to prize disfigurement- linkedlandscapes from .This is achieved by resizing FM and inducing the mask sequence {FMi}.

FM = x, m = 1,., L (12)

In this model, ⊙ denotes element-wise addition, L is layer count in GE, And, F refers to features associated with uprooted disfigurement. Through a three-subcases MLP, F and labors F are further reused, with the number of disfigurement orders replacing the number of FM channels. For the training of the disfigurement classifier, this cross-entropy loss is applied, and we first feed F into a Soft Max activation function

S = ∑ m = 1 Kyi˜ log Si (13)

Where K is the number of deformity orders, yi denotes the one-shot embedding of the underlying verity marker, and Si denotes the Soft Max activity function's concern.

LC=12∑i=1N‖x’−cyi‖22 (14)

Where x’ indicates the MLP's learnt deep features affair ,The cyi denotes the yi- the order center of the deep structures. The common loss function is ladened and designed.

L = w1 (LS) w2(LC) (15)

Where weights w1 and w2 are set to0.9 and0.1, independently, grounded on a public dataset was used for cross-validation.

D. Model Lightweight Processing

The suggested solution is light and designed to fulfill the real-time needs of embedded devices. Since these modules are only utilized during the training period, this is the case. After preparation, we reconstruct the input fabric image using only the encoder GE. After that, in the convolution layers, we decrease the number of channels of the remaining modules to reduce the computational complexity. Trimming the suggested model makes processing easier. The circumcision procedure is separated into two parts: The suggested solution is light and designed to fulfill the real-time needs of embedded devices. Trimming the suggested model makes processing easier. The circumcision procedure is separated into two parts:

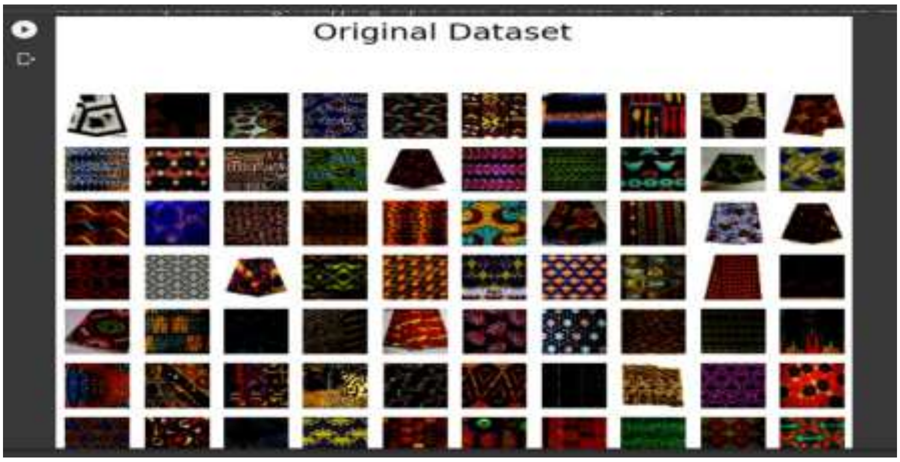
The decoder GD, encoder E, and discriminator D of the generator are first deleted. Since these modules are only utilized during the training period, this is the case. After training, we use the encoder GE to recreate the tissue image input.

Decrease the amount of channels in the following modules' convolutional layers to minimize computational complexity.

The dataset that is being used here contains almost 1059 image files which are uploaded before importing required library files. The dataset models are the African fabrics, which is named after the African people. These datasets are collected based on the fabrics and clothe materials that are worn by African country people. These fabrics constitutes several material types and defect types.

The material types includes synthetic, cotton fabric, silk, silk cotton, Nylon, Designer clothes, wool, leather and etc. It also varies among different colors and the defect types include yarn defects, isolated defects, pattern defects, wet processing defects, weaving defects, raising defects, milling defects and etc. This proposed dataset model stands best when compared with the existing dataset model which constitutes only certain type of defects such as knot, hole, weaving defects and so on. Since the existing dataset model is limited when comparing to our proposed methodology, it lacks accuracy and effectiveness. Images are subsequently resized to 64 × 64. For each data set/experiment, all algorithms are trained on the same data set.

Performance Analysis of Generation Model



In addition, he collected a dataset of 1.5 K defective fabric images of various material types. These images have a 256-pixel shortest side. This dataset was large and was employed in generative models, allowing for many assessments to the cutting edge. The figure 6 represents the original African fabrics dataset.Because of generative models have a more difficult time producing images that are equivalent to genuine photographs. This task is therefore assessing the quality of the composite image compared to a recent photo.

This dataset was large and was employed in generative models, allowing for many assessments to the sharp edge. Experimentations with fine-grained object class minimum and maximum sizes yielded worse results. We ascended these dimensions using the relevant scale factor to composite the image at the lower resolution. All techniques are developed on the same information set for every data set/experiment. In addition, he collected a dataset of 1.5 K defective fabric images of various material types. These images have a 256-pixel smallest side.

Table 1. Structure of Imrpoved GAN

Model: "sequential"

Layer (type)	Output Shape	Param #
dense (Dense)	(None, 16384)	1638400
batch_normalization (BatchNo	(None, 16384)	65536
leaky_re_lu (LeakyReLU)	(None, 16384)	0
reshape (Reshape)	(None, 8, 8, 256)	0
conv2d_transpose (Conv2DTran	(None, 16, 16, 256)	1048576
batch_normalization_1 (Batch	(None, 16, 16, 256)	1024
leaky_re_lu_1 (LeakyReLU)	(None, 16, 16, 256)	0
conv2d_transpose_1 (Conv2DTr	(None, 32, 32, 256)	1048576
batch_normalization_2 (Batch	(None, 32, 32, 256)	1024
leaky_re_lu_2 (LeakyReLU)	(None, 32, 32, 256)	0
conv2d_transpose_2 (Conv2DTr	(None, 64, 64, 256)	1048576
batch_normalization_3 (Batch	(None, 64, 64, 256)	1024
leaky_re_lu_3 (LeakyReLU)	(None, 64, 64, 256)	0
conv2d_transpose_3 (Conv2DTr	(None, 64, 64, 3)	6912

Total params: 4,839,648
Trainable params: 4,825,344
Non-trainable params: 34,304

E. The Analysis of the generated images:

The analyses were carried out on a workstation that had a TitanRTXgpu architecture and 24 gigabytes of random access memory (RAM) in order to develop a model which generates fabric defects. Jupyter notebooks were used to create the model, as well as TensorFlow (2.2.1) and Python3.6. The training dataset consisted of 2,480 paired photos, with a total of 1,100 of those images having belt yarn faults, 652 of those images displaying loop faults, and 637 of those images displaying stained defects. In addition, 936 sets of paired photos were chosen to serve as the test set in order to assess the performance of the model, which consisted of 310 images paired with belt yarn defects, as illustrated in Figure 7-9.

It was a large dataset that was used in generative models, which allowed for many comparisons with cutting-edge research. This finding is compared with that obtained when the object detector was trained with real-world images. A collection of 1,500 genuine pictures accompanied by handwritten annotations and bounding boxes are used to evaluate each trained model. As a result, this assignment involves comparing the quality of the composite image to a current photograph. Each and every one of Images of staged items is of high quality. Because of the high quality and professional aspect of these images, generative models have a more difficult time producing images that are equivalent to genuine photographs. All algorithms are trained on the same data set for each data set/experiment.

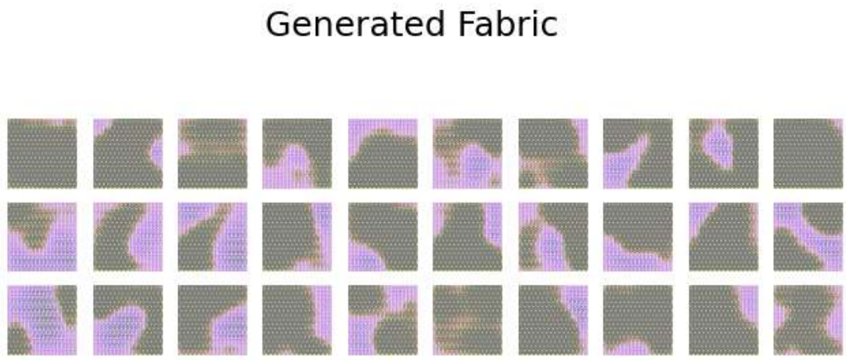


Fig7: A typical test set of fabric defects.

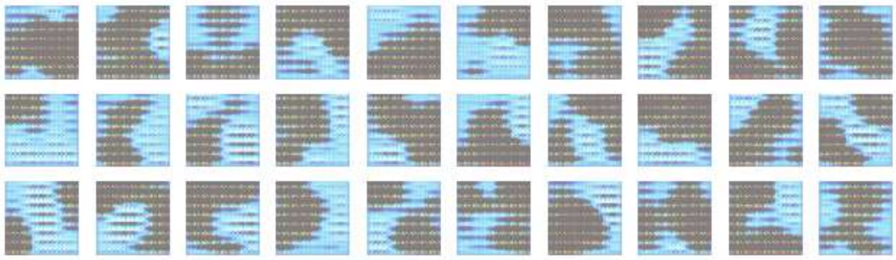


Fig8: Results of the reconstruction of the samples

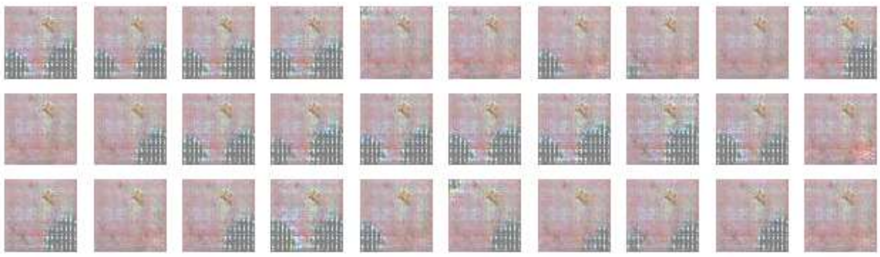


Fig9: An analysis of a public dataset for defect detection.

The initial test was performed to evaluate the reliability of the produced fabric fault image. This project employed GAN, ResNet, and RCNN to produce fabric flaws for comparison. An assessment of the quality of the produced image was conducted by measuring the peak signal to noise ratio (PSNR) and the structural similarity index measure (SSIM). The PSNR is a popular metric for assessing the quality of two images by compressing the difference between them. By contrast, the PSNR focuses only on pixel changes between two images and ignores factors such as contrast, brightness, and geographical perception. Consequently, the SSIM value was used in this study in conjunction with the PSNR value to determine the quality of the image produced. The SSIM value as a means of comparing two images in terms of their structural similarity. In light of the fact that defect shapes vary, the SSIM value may provide a more scientific indication of the disparity between the produced image and the ground truth compared to the PSNR value. When comparing the two indicators, a higher PSNR value indicates a higher quality image. The higher the SSIM score, the better the quality of the image produced. It is possible to compute PSNR and SSIM as follows:

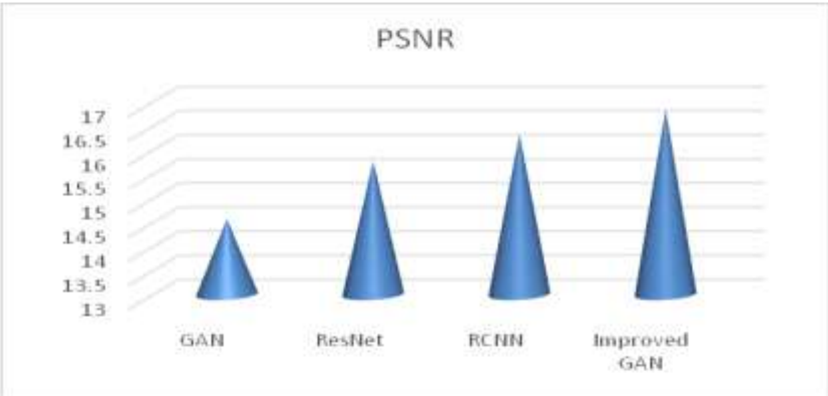
$$\begin{cases} MSE = \frac{1}{G \times W} \sum_{i=2}^{H} \sum_{j=1}^{W} (X(a,b) - Y(a,b)^2)) \\ PSNR = 10 \log_{10} \left(\frac{(2^n - 1)^2}{MSE} \right) \end{cases} \quad (16)$$

$$\begin{cases} l(x,y) = \dfrac{2xy + C1}{x + y + C1} \\ c(a,b) = \dfrac{2ab + C2}{a + b + C2} \\ s(m,n) = \dfrac{2mn + C3}{m + n + C3} \\ SSIM(n) = l(x,y) * c(a,b) * s(m,n) \end{cases} \quad (17)$$

A measure of the mean square error is MSE, whereas X(a,b) and Y(a,b) correspond to gray values at the image point (i, j) for x and Y, respectively. In this research, n is eight. l(a,b) represents image brightness, c(a,b) represents image contrast, and s(a,b) represents image structural similarity. For the denominator, C1, C2, and C3 are constant values to avoid zeros in the standard deviations, mean values, and image covariance, respectively.

Table 2 shows the assessment findings of the created fabric fault image

Generation Model	**Performance Metrics**		
	PSNR	**MSE**	**SSIM**
GAN	14.5632	17.3254	0.2145
ResNet	15.7682	16.7825	0.1354
RCNN	16.3248	17.2453	0.1546
Improved GAN	16.8365	13.2546	0.2689



.

Fig10: PSNR Value

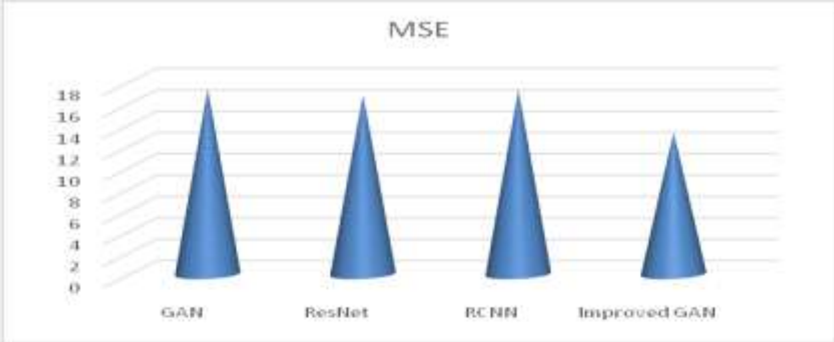


Fig11: MSE value

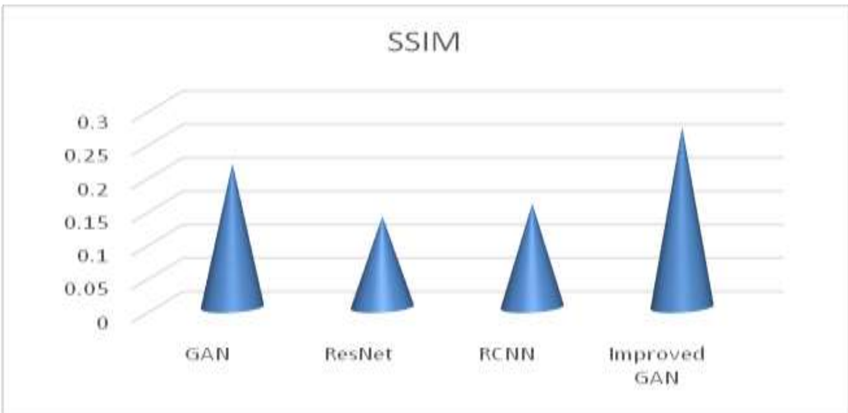


Fig12: SSIM Value

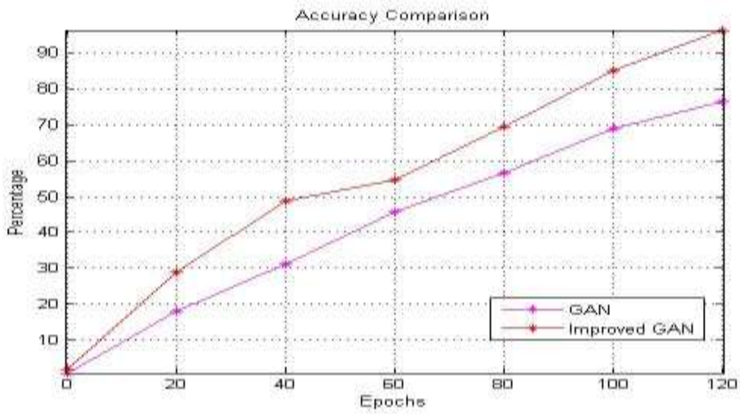


Fig13: Accuracy

It can be found from fig 10-13that the proposed improved GAN can achieve the best performance than existing methods.

3. Conclusion

This paper proposes an automated fabric flaw identification approach based on an upgraded GAN. Taking into account the numerous fabric defects and the unbalanced distribution of data, we presented the GAN as a method for discovering ubiquitous faults in an unsupervised manner.During the training step, the GAN learnt how to rebuild fabric images from normal samples. As a result, since it learnt nothing about fabric faults during the training stage, it was unable to recreate them. To re-encode the rebuilt sample, a new encoder was added in contrast to the standard GAN. Latent vector learning is enhanced by the suggested encoding loss. In order to identify ubiquitous flaws, the difference between the input fabric images and the reconstructed outcomes produced by the well-trained GAN was calculated. A coarse defect mask was constructed based on the difference images, which was then used to retrieve defect-related information from the feature maps generated by the generator. In order to detect the category of the particular defect, the output of the MLP was entered into the Soft-max algorithm. To enhance the discriminability of the learned features, the center loss constraint was applied. XuelangTianchi AI Competition dataset was used to assess the practicality and accuracy of the suggested method. This technology has been shown to be effective in attaining automated quality detection in textile manufacturing as well as enhancing textile production efficiency.

4. References

[1] Rui, J., &Qiang, N. (2022). Research on textile defects detection based on improved generative adversarial network. Journal of Engineered Fibers and Fabrics, 17, 15589250221101382.

[2] Wang, C., Wang, S., Wang, L., Cao, C., Sun, G., Li, C., & Yang, Y. (2022). Framework of nacelle inverse design method based on improved generative adversarial networks. Aerospace Science and Technology, 121, 107365.

[3] Xiang, J., Pan, R., &Gao, W. (2022). Online detection of fabric defects based on improved centernet with deformable convolution. Sensors, 22(13), 4718.

[4] Yue, X., Wang, Q., He, L., Li, Y., & Tang, D. (2022). Research on tiny target detection technology of fabric defects based on improved Yolo. Applied Sciences, 12(13), 6823.

[5] He, X., Chang, Z., Zhang, L., Xu, H., Chen, H., & Luo, Z. (2022). A survey of defect detection applications based on generative adversarial networks. IEEE Access, 10, 113493-113512.

[6] Hao, Z., Wang, Z., Bai, D., Tao, B., Tong, X., & Chen, B. (2022). Intelligent detection of steel defects based on improved split attention networks. Frontiers in Bioengineering and Biotechnology, 9, 1478.

[7] Fan, Y., Li, Y., Shi, Y., & Wang, S. (2022). Application of YOLOv5 neural network based on improved attention mechanism in recognition of Thangka image defects. KSII Transactions on Internet and Information Systems (TIIS), 16(1), 245-265.

[8] Wang, B. (2022). A study on spot welding quality judgment based on improved generative adversarial network and auto-encoder. Mechanical Systems and Signal Processing, 165, 108318.

[9] Zhang, H., Qiao, G., Lu, S., Yao, L., & Chen, X. (2022). Attention-based Feature Fusion Generative Adversarial Network for yarn-dyed fabric defect detection. Textile Research Journal, 00405175221129654.

[10] Huang, Y., & Xiang, Z. (2022). RPDNet: Automatic Fabric Defect Detection Based on a Convolutional Neural Network and Repeated Pattern Analysis. Sensors, 22(16), 6226.

[11] Li, D., Li, J., Fan, Y., Lu, G., Ge, J., & Liu, X. (2022). Printed label defect detection using twice gradient matching based on improved cosine similarity measure. Expert Systems with Applications, 204, 117372.

[12] Jing, J., Wang, Z., Rätsch, M., & Zhang, H. (2022). Mobile-Unet: An efficient convolutional neural

network for fabric defect detection. Textile Research Journal, 92(1-2), 30-42.

[13] Cheng, H., Liang, J., & Liu, H. (2022). Image restoration fabric defect detection based on the dual generative adversarial network patch model. Textile Research Journal, 00405175221144777.

[14] Yu, T., Chen, W., Junfeng, G., &Poxi, H. (2022). Intelligent Detection Method of Forgings Defects Detection Based on Improved EfficientNet and Memetic Algorithm. IEEE Access, 10, 79553-79563.

[15] Zhou, Z., Deng, W., Zhu, Z., Wang, Y., Du, J., & Liu, X. (2022). Fabric defect detection based on feature fusion of a convolutional neural network and optimized extreme learning machine. Textile Research Journal, 92(7-8), 1161-1182.

[16] Rahimunnisa, K. (2022). Textile Fabric Defect Detection. Journal of Innovative Image Processing, 4(3), 165-172.

[17] Yu, M., Yue, H., Chen, C., Zhou, C., Huang, Y., Wang, D., & Deng, F. (2022, August). An Accurate Defect Detection Model for Photosensitive Elements Based on Improved YOLOV4 Deep Neural Network. In 2022 5th International Conference on Pattern Recognition and Artificial Intelligence (PRAI) (pp. 337-345). IEEE.

[18] Akhyar, F., Furqon, E. N., & Lin, C. Y. (2022). Enhancing precision with an ensemble generative adversarial network for steel surface defect detectors (EnsGAN-SDD). Sensors, 22(11), 4257.

[19] Liu, Z., Gao, C., Li, C., Huang, N., &Guo, Z. (2022, May). Unsupervised fabric defect detection based on DCGAN with component-encoder. In Pattern Recognition: 6th Asian Conference, ACPR 2021, Jeju Island, South Korea, November 9–12, 2021, Revised Selected Papers, Part I (pp. 557-568). Cham: Springer International Publishing.

[20] Baizhan Xia,Hao Luo,and Shiguang Shi. (2022, May). Improved Faster R-CNN Based Surface Defect Detection Algorithm for Plates. Journal of Computational Intelligence and Neuroscience, University of Electronic Science and Technology of China, Zhongshan 528402, China.

[21] Junjun Fan, Wai Keung Wong, and Zhihui Lai (2022, March). Fabric Defect Detection Using Deep Convolution Neural Network. AATCC Journal of Research(Volume 8).

[22] Cheng, L., Tong, Z., Xie, S., &Kersemans, M. (2022). IRT-GAN: A generative adversarial network with a multi-headed fusion strategy for automated defect detection in composites using infrared thermography. Composite Structures, 290, 115543.

[23] Unsupervised Fabric Defect Detection Based on DCGAN with Component – Encoder, Published as a part of the Lecture Notes in Computer Science book series (volume 13188), Published by springer on November 2021.

[24] Fabric Defect Detection Based on Lightweight Neural Network, Published as a part of the Lecture Notes in Computer Science book series (volume 11857), Published by springer on October 2019.

[25] Fabric Defect Detection by Applying Structural Similarity Index to the Combination of Variational Auto Encoder and Generative Adversarial Network, Published as a part of Smart Innovation, System and Technologies book series(volume 314), Published by springer on November 2022.

[26] A Survey of Defect Detection Applications Based on Generative Adversarial Networks, Published in IEEE Access International Journal (volume 10) on September 2022.

[27] Multistage GAN for Fabric Defect Detection, Published in IEEE Transactions on Image Processing (volume 29) International Journal, on December 2020.

[28] A Fabric Defect Detection Method Based on Deep Learning, Published in IEEE Access (volume 10), on January 2022.

[29] Shadow Compensation for Synthetic Aperture Radar Target Classification by Dual Parallel Generative Adversarial Network, Published in IEEE Sensors Letters (volume 4, Issue 8) on August 2020.

[30] Conditional Wasserstein Generative Adversarial Network Gradient Penalty based approach to alleviating imbalanced data classification, Published in Information Sciences: an International Journal Volume 5512 on February 2020.

[31] Defect Detection of Fabrics with Generative Adversarial Network Based Flaws modelling, Published in Chinese Automation Congress: an International Conference, on January 2021.

[32] GAN -Based Defect Synthesis for Anomaly Detection in Fabrics, 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), on November 2020.