Automation and Artificial Intelligence in the Textile Industry: A Study on Generating Samples of Defective Fabrics with Generative Adversarial Networks

Angelo Leite Medeiros de Góes[1,⋆], *Adrião* Duarte Dória Neto[1,⋆⋆], *Thiago* Theiry de Oliveira[1,⋆⋆⋆], and *Matheus* Gomes Diniz Andrade[2,⋆⋆⋆⋆]

[1] *Electrical and Computer Engineering Postgraduate Program - PPGEEC*
[2] *Federal University of Rio Grande do Norte - UFRN*

Abstract. This research focuses on the impact of automation on production systems and the role of human operators in inspection and quality verification. Despite the application of statistical techniques, fabric defects continue to be a significant challenge in the textile industry. To address this issue, the study proposes the utilization of a deep convolutional neural generative adversarial network (DCGAN) trained on textile defect databases to generate new images for defect detection models. The objectives of this study include gaining a thorough understanding of the adversarial network, selecting and preprocessing training images, documenting the training process, evaluating the results obtained, and assessing the practicality of the model. After 2000 epochs, the network exhibited a notable ability to generate novel images with distinct defects, which could be utilized to augment existing fabric datasets. The research aims to enhance automated inspection processes by developing intelligent defect detection systems.

1 Introduction

The rapid progress in computing and robotics has brought about a revolutionary change in production systems, profoundly influencing the entire process of product design and manufacturing, primarily due to automation. Consequently, this technological advancement has significantly altered the role of human operators in the manufacturing environment, shifting them from highly specialized individuals operating in low-tech systems to adaptable and versatile workers operating in high-tech systems [1].

Nevertheless, although automation is extensively employed in product construction and assembly, the crucial tasks of inspection and quality verification continue to rely predominantly on human labour. Acknowledging the inherent limitations of relying solely on human inspection to identify all defective items, the industry has embraced a sampling inspection approach supplemented by statistical quality control [2]. However, despite the considerable benefits derived from effectively implementing statistical methods, they are not infallible.

In the textile industry, fabric defects represent a significant portion of reported issues, accounting for approximately 85% of total defects. This leads to substantial profit reductions for manufacturers, ranging from 45% to 65%, as a result of reselling or lower-quality products [3][4]. Furthermore, studies have shown that human inspectors typically only identify around 80% of these defects [1]. Given the recent advancements in computer vision and artificial intelligence, manufacturers have increasingly focused on inspection systems based on these technologies. However, despite these advancements, ensuring reliable inspection remains an ongoing challenge. As a result, in critical applications where defect presence must be minimized, the most widely adopted approach is reinspection, where items are repeatedly checked until all defective products are completely removed [2].

To improve the dependability of automated inspection processes, the development of resilient intelligent systems for defect detection is paramount. Yet, acquiring images of flawed fabrics for training purposes can pose challenges, as obtaining them directly from factory systems, without the necessary infrastructure and setup, is not always feasible. In light of this, we suggest leveraging fabric defect images, acquired in real-world settings, to train a generative adversarial network (GAN) and generate novel samples.

The primary aim of this study is to train an autoencoder using publicly available databases of textile defects, such as AITEX and TILDA, with the purpose of generating fresh images depicting defective fabrics. These images will be used to train and validate more robust models for defect detection. The specific objectives include gaining an in-depth understanding of the autoencoder network's operation, selecting and preprocessing the images designated for autoencoder training, conducting and documenting the network training process, evaluating the obtained results, and ultimately arriving at a conclusive assessment regard-

⋆e-mail: angelo.leite.056@ufrn.edu.br
⋆⋆e-mail: adriao@dca.ufrn.br
⋆⋆⋆e-mail: thiago.oliveira.016@ufrn.edu.br
⋆⋆⋆⋆e-mail: matheus.diniz.122@ufrn.edu.br

ing the feasibility and effectiveness of the proposed model in addressing the underlying problem.

In Section 2, this article will provide the essential theoretical groundwork. In Section 3, the methodology will delve into the construction of the database and the training of the network. In Section 4 the results will be presented, and lastly, Section 5 will encompass the conclusions drawn from the study as well as future perspectives for the proposed approach.

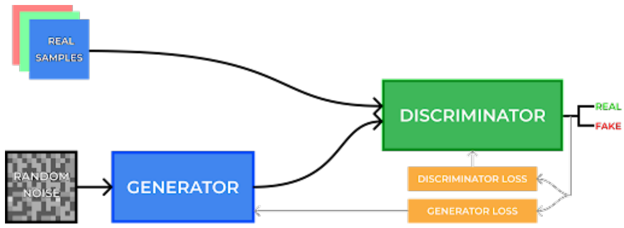

Figure 1. Overview of GAN Structure.

2 Theoretical foundation

The developed system has its theoretical technical foundation based on Generative Adversarial Networks (GAN) and Deep Convolutional GAN (DCGAN).

2.1 Generative Adversarial Networks

Generative Adversarial Networks (GANs) involve an innovative approach in the field of machine learning. GANs are a type of generative model that consists of two distinct neural networks, the generator and the discriminator, which operate in an adversarial learning scenario.

The generator is responsible for creating new synthetic samples that resemble real data from the training set. It takes random noise as input and transforms it into an output that resembles the desired data. The goal of the generator is to deceive the discriminator into incorrectly classifying the generated samples as real.

On the other hand, the discriminator is trained to distinguish between real and synthetic samples generated by the generator. It takes a sample as input and estimates the probability of it being real or synthetic. The objective of the discriminator is to accurately differentiate between real and generated samples.

The interaction between the generator and the discriminator occurs during the training of GANs. The generator aims to improve its ability to deceive the discriminator, while the discriminator aims to enhance its ability to distinguish between real and synthetic samples. This adversarial learning process leads to the simultaneous improvement of both the generator and the discriminator. Figure 1 provides a comprehensive overview of the GAN structure.

The main advantage of GANs is their ability to generate high-quality synthetic samples that resemble real data. This makes them especially useful in tasks such as image generation, as developed in this system.

2.2 Deep Convolutional Generative Adversarial Networks

Deep Convolutional Generative Adversarial Networks (DCGANs) represent an advancement in traditional GANs, with a key distinction lying in their network architecture. DCGANs employ convolutional layers in both the generator and discriminator, enabling the modeling of spatial patterns and the extraction of significant image features. This convolutional structure is especially effective for image data, as it considers local structures and spatial correlations among pixels. Furthermore, DCGANs incorporate batch normalization techniques, such as Batch Normalization, which contribute to training stability and enhance the model's ability to generalize.

In DCGANs, pooling layers, typically employed in convolutional networks for data dimensionality reduction, are replaced with transposed convolution layers (also known as deconvolution). This substitution allows for an expansion of data dimensionality and the generation of higher-resolution images. Compared to traditional GANs, DCGANs have demonstrated more reliable and superior outcomes in image generation [5]. They excel at learning and replicating intricate and detailed features, leading to the production of more realistic and visually captivating images. The general architecture of the employed network can be seen in Figure 2.

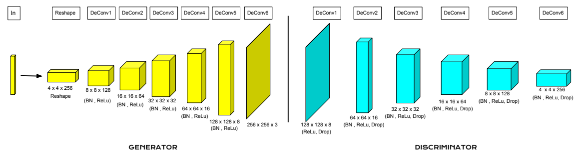

Figure 2. The overall architecture of Deep Convolutional Generative Adversarial Network.

3 Methodology

The experiment was conducted using the TILDA database, developed by the German Texture Analysis Group of the DFG (Deutsche Forschungsgemeinschaft). This database consists of a collection of images in the context of the textile industry, aimed at studying and analyzing different types of defects present in fabrics.

The TILDA database comprises eight classes, with seven of them representing different types of defects found in fabrics, while one class is dedicated to samples without any defects. Each class contains a total of 50 images, resulting in a dataset with a total of 400 images. The images in this database have dimensions of 768 x 512 pixels.

In terms of implementation, the experiment code was executed in the Google Colab testing environment. Colab is a cloud-based platform that provides free computing resources for Python code development and execution. It is widely used for machine learning and data science projects due to its easy access and integration with popular libraries such as TensorFlow and PyTorch.

At the beginning of the experiment, the 400 images from the TILDA database were loaded and underwent preprocessing. In this stage, the images were resized to the 256x256 pixel format. Next, pixel value normalization was applied, scaling the values between -1 and 1. This normalization is a common practice in GAN models as it helps improve training stability and convergence. Additionally, this value range is suitable for the use of activation functions, such as the hyperbolic tangent, in the model's generator.

After the preprocessing of the images, the data was organized into batches of size 32. This step aims to group the images into smaller sets for model training. Grouping the images into batches allows the model to process multiple samples simultaneously, which can speed up training time and improve computational efficiency.

Organizing the data into batches of size 32 means that, at each training iteration, the model will receive a set of 32 images to process and adjust its parameters. This approach is widely adopted in machine learning algorithms to optimize performance and ensure a balanced distribution of examples during training.

The collected data was ultimately employed to train the networks, utilizing two distinct architectures as previously mentioned. These architectures were based on the original DCGAN framework proposed by Radford et al. [6], with modifications made to accommodate the new input and output size requirements. The subsequent sections outline the specific architectures utilized for each network.

3.1 Generator

The architecture of the generator network consists of a sequence of layers that transform a one-dimensional input of 100 values into an output tensor with dimensions 256x256x3. This transformation is achieved through various layers, including batch normalization, Leaky ReLU activation and transpose convolution.

The initial input of 100 values undergoes a dense layer. Subsequently, batch normalization is applied to ensure training stability and improve convergence. The Leaky ReLU activation function is then applied to introduce nonlinearities into the network. The resulting tensor is reshaped into a 4x4x256 format using a reshape layer. This layer does not introduce additional parameters but enables easier handling of subsequent convolutional layers.

From there, several transpose convolution layers are applied. Each of these layers gradually increases the size of the tensor, starting from a 4x4x256 size and ending with 256x256x3. The transpose convolution layers have a kernel size of 5x5, a stride of 2x2, and use padding to maintain the spatial dimensions of the feature maps. Each transpose convolution layer is followed by batch normalization and a Leaky ReLU activation function. The last transpose convolution layer utilizes the hyperbolic tangent activation function to map the pixel values to the valid range of -1 to 1.

3.2 Discriminator

The architecture of the discriminator model is defined as a sequence of convolutional layers, followed by batch normalization layers, Leaky ReLU activation functions, and Dropout layers. The main objective of the discriminator is to distinguish between real images and images generated by the generator.

The model begins with a convolutional layer that receives an input with dimensions 256x256x3. This layer uses a kernel size of 5x5, a stride of 2x2, and padding to maintain the spatial dimensions of the feature maps. A Leaky ReLU activation function is applied, which allows for a small slope in the negative region. Additionally, a regularization technique called Dropout with a rate of 0.3 is applied to reduce overfitting.

The architecture of the discriminator continues with a sequence of layers similar to those mentioned above, aiming to progressively reduce the spatial size of the input tensor. After the convolutional layers, a flatten layer is added, which transforms the tensor into a one-dimensional vector. Next, a dense layer, also known as the logits layer, is added, which produces a one-dimensional output for image classification. This layer does not use an activation function.

4 Results

In order to conduct a comprehensive evaluation of the obtained results, an approach was adopted that involved generating 16 vectors of size 100, composed of random noise. These vectors were specifically created to visualize the output generated by the model's generator.

This strategy allowed for a detailed and thorough analysis of the resulting images, enabling a more comprehensive understanding of the generator's performance. By examining these images, it was possible to assess the model's ability to produce realistic and coherent samples, evaluate the diversity of the generated images, and identify potential patterns or undesirable distortions. The Figure 3 shows the output of the network immediately after the first iteration.

As the epochs progressed, a significant improvement in the generator network became evident. A notable milestone was reached at epoch 200, where a resemblance between the generated images and real fabrics became apparent. At this stage, the images (Figure 4) started to exhibit patterns, realistic textures, and refined details, effectively resembling textile samples.

At the epoch 2000 (Figure 6), the presence of fabric flaws became visible in the generated images.

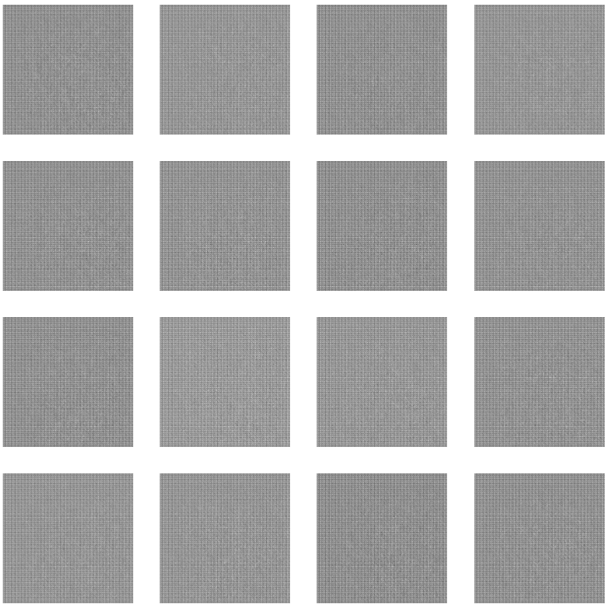

Figure 3. Generator output at epoch 0.

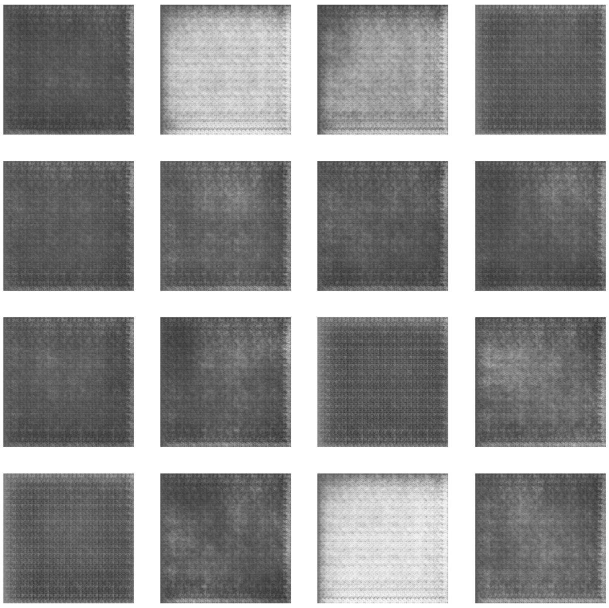

Figure 4. Generator output at epoch 200.

This is considered positive, as the motivation behind this process is to generate more data with defects to expand the defect databases. By intentionally introducing these flaws into the generated images, it is possible to create a more comprehensive and diverse dataset that can be used to train and test defect detection algorithms.

5 Conclusion

Based on the findings, this study provides a comprehensive analysis of the application of Generative Adversarial Networks (GANs) in automated inspection within the textile industry. The results demonstrate the promising potential

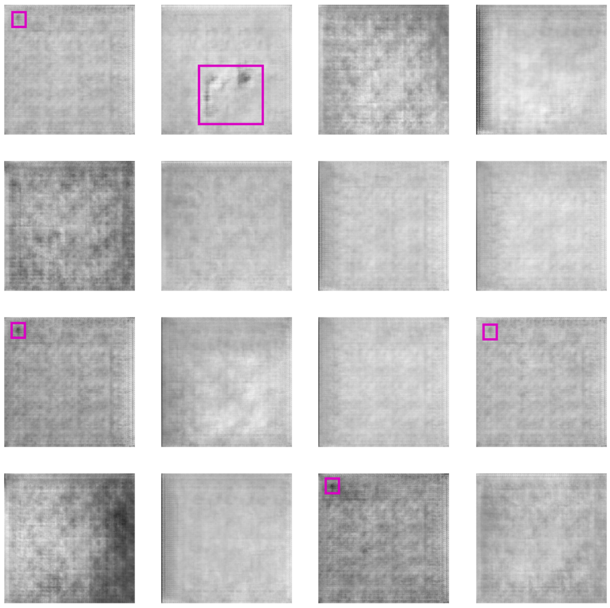

Figure 5. Generator output at epoch 2000.

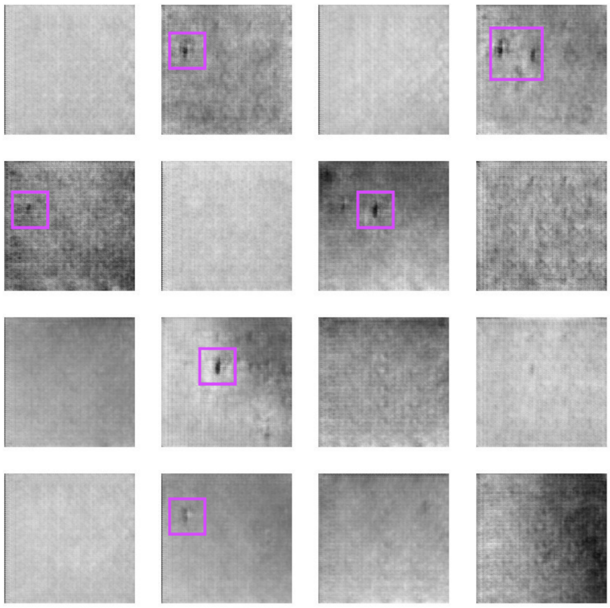

Figure 6. Generator output at epoch 3000.

of GANs in generating realistic fabric samples, including accurate replication of defects. Notably, as the training progressed, the generator exhibited substantial improvement in generating fabric-like images. By epoch 200, a good similarity between the generated images and real fabric samples was observed, showcasing authentic patterns and realistic textures.

As the training progressed near epoch 2000 and beyond, the generated images increasingly resembled defective samples. This can lead to the development of a more diverse and comprehensive dataset suitable for training and evaluating better defect detection networks. This also emphasizes the effectiveness of GANs in producing fabric representations of good quality, demonstrating their potential in facilitating automated inspection processes.

In summary, this approach has the potential to significantly improve defect databases, leading to more accurate and efficient inspections within the textile industry. The realism achieved holds the promise of enhancing the efficiency of quality control, thereby reducing financial losses linked to textile defects. The successful generation of fabric samples, defects included, showcases the valuable contribution of DCGANs as a promising tool for enhancing defect detection systems in contemporary manufacturing settings.

References

[1] J. Pesante-Santana, J. Woldstad (2000)

[2] C. Sylla, IFAC Proceedings Volumes **33**, 313 (2000), 2nd IFAC Conference on Management and Control of Production and Logistics (MCPL 2000), Grenoble, France, 5-8 July 2000

[3] N. Sandhya, N.M. Sashikumar, M. Priyanka, S. Maria, K.K. Wenisch, Textile & Leather Review **4**, 315 (2021)

[4] A.W. P. Sengottuvelan, A. Shanmugam, Research Journal of Applied Sciences pp. 26–31 (2008)

[5] V.M.M. Alvarez, *Image-generation-gan-vs-dcgan*, https://github.com/vmartinezalvarez/Image-generation-GAN-vs-DCGAN (2019)

[6] A. Radford, L. Metz, S. Chintala, *Unsupervised representation learning with deep convolutional generative adversarial networks* (2016), 1511.06434