2023 IEEE Symposium Series on Computational Intelligence (SSCI)
Mexico City, Mexico. December 5-8, 2023

Enhancing Document Digitization: Image Denoising with a Cycle Generative Adversarial Network

Gerardo Lugo-Torres
Center for Computing Research (CIC-IPN)
Instituto Politécnico Nacional
Mexico City, Mexico
glugot2022@cic.ipn.mx

Diego A. Peralta-Rodríguez
Computational Cognitive Sciences Lab. - CIC
Instituto Politécnico Nacional
Mexico City, Mexico
dperaltar1100@alumno.ipn.mx

José E. Valdez-Rodríguez
Computational Cognitive Sciences Lab. - CIC
Instituto Politécnico Nacional
Mexico City, Mexico
jvaldezr2018@cic.ipn.mx

Hiram Calvo
Computational Cognitive Sciences Lab. - CIC
Instituto Politécnico Nacional
Mexico City, Mexico
hcalvo@cic.ipn.mx

***Abstract*—In the era of big data, we now live on, there is an increasing demand to convert large amounts of scanned documents, such as texts, medical records, and images into digital formats. However, often when scanning introduces imperfections such as salt-and-pepper or background noise, blurring caused by camera motion, watermarking, coffee stains, wrinkles, or faded text. These imperfections carry significant challenges to current algorithms of text recognition, leading to a decline in their performance. To date, a wide range of methods are aimed at reducing noise. This work compares the performance of a CycleGAN model concerning median filter, Wiener filter, adaptive threshold, morphological filtering, and a CNN-based autoencoder. While the CNN-based autoencoder technique gave us the best results, the CycleGAN model approach provided us with comparable results with only 50 training epochs in contrast to the 700 epochs of the CNN-based autoencoder and was superior to the rest of the other contrasted methods. Likewise, data preparation for the training is much simpler in the CycleGAN model due to its property of requiring only unpaired data for training.**

***Index Terms*—Cycle, Generative Adversarial Network, Image Denoising, Autoencoder, Filter, Median, Wiener, Adaptative Threshold, Morphology**

I. Introduction

Due to environmental factors, transmission channels, and various other influences, noise becomes unavoidable in image acquisition, compression, and transmission processes. Following image processing operations like video processing, image analysis, and tracking are negatively impacted by the distortion and loss of image information caused by this noise. Image denoising is therefore crucial in contemporary image processing systems [1]. Image denoising's main objective is to eliminate noise from an image and return it to its true representation. Due to the high-frequency nature of noise, edges, and textures, it might be difficult to discern them during the denoising process, potentially leading to the loss of some information in the denoised photos. The current challenge is to obtain high-quality results by removing noise from noisy photos while preserving significant information. Image denoising has been a persistent and extensively studied problem [2], yet it remains a complex and ongoing task. The primary challenge stems from the fact that image denoising is an inverse problem mathematically lacking a unique solution [3].

In recent years, a growing number of studies have concentrated on generative models for image synthesis [4]. These advances have increasingly influenced a lot of research fields, such as image noise processing [5]. Although using machine learning techniques, such as CNN-based autoencoders, is quite common in noise processing [6], they present specific challenges. These include the requirement for large volumes of data and significant training time to obtain quality results. Additionally, there is the issue of needing large amounts of labeled data to train high-performing CNNs. In this work, we investigate whether a CycleGAN model can denoise text images containing background noise such as smudges, folds, and grime in terms of the quality of the output results in comparison to more prevalent methods. The contributions of this work are:

- We contrasted the performance of median filtering, Wiener filtering, adaptative threshold, morphological filtering, and a CNN-based autoencoder concerning the CycleGAN model.
- Although the CNN-based autoencoder outperforms our CycleGAN model, the CycleGAN still outperforms the other methods while maintaining a close performance concerning the CNN-based autoencoder.

II. Related works

Text image denoising, a topic thoroughly examined over decades, addresses the diverse noise challenges text can face. Efforts, such as median filtering [7], tackle smudges and dirt. Advanced methods use adaptive median filtering [8] and wavelet transform [9] to retain text details. Although morphological methods have been less explored, they've been

used for text reconstruction and OCR [10]. Machine learning methods, particularly recurrent convolutional neural networks, have shown promise in document binarization and denoising, subsequently improving OCR results [11]. Autoencoders, like CNN-based variants, have been pivotal for noise reduction in both generic images [11], [12] and text denoising [13] given their robust data representation. Generative models, including the CR-GAN [14] and Super Resolution GAN [15], have been employed for image quality enhancement. Zhu's work employed a CycleGAN for background noise reduction in text, achieving a PSNR of 27.88 dB [16]. In specialized contributions, Neji's Blur2Sharp Cycle GAN demonstrated impressive results with a PSNR of 32.52 dB [17]. Nigam's VCGAN [18] was designed to rectify deformed handwritten text, and Gangeh's model integrated a deep MoE with a CycleGAN, outperforming methods like REDNet and DE-GAN [19], [20].

III. Methodology

A. Spatial domain filtering

Linear and non-linear filtering methods, such as the median filter, Wiener filter, and adaptive thresholding, are commonly employed for image denoising and enhancement [2]. A summary of each method is presented: 1) The median filter is a non-linear filtering technique utilized in image processing. It involves the replacement of each pixel in an image with the median value derived from the nearby pixels within a designated window or kernel. It is particularly effective in reducing impulse or "salt-and-pepper" noise, where a few pixels have extreme values. 2) Wiener Filter is a statistical filter that aims to minimize the mean square error between the original and filtered images. It is an optimal linear filter for restoring images corrupted by additive noise. The Wiener filter considers the noise and image power spectra to estimate the most suitable filter parameters for noise reduction. 3) Adaptive Thresholding is a technique where the threshold for image segmentation is determined locally based on the characteristics of the image. Instead of using a fixed threshold value for the entire image, adaptive thresholding adjusts the threshold dynamically according to local pixel intensities. It is important to note that while these filtering methods can be effective in certain situations, they also have limitations. For instance, they may encounter difficulties handling complex noise patterns or preserving fine details in the image.

B. Morphological filtering

Morphological filtering is a non-linear image processing technique that focuses on objects' shape, structure, and connectivity within an image. Morphological filtering operates on binary or grayscale images and utilizes two fundamental operations: dilation and erosion. These operations involve using a structuring element, a small, predefined shape, or a template. The structuring element defines the neighborhood around each pixel during the filtering process. Morphological filtering's primary operations are dilation and erosion, but it also includes more advanced operations such as opening, closing, and morphological gradients.

C. CNN-based autoencoder

A CNN-based autoencoder is a type of neural network architecture that combines convolutional neural networks (CNNs) with an autoencoder framework. Autoencoders are a type of unsupervised learning models that have been specifically developed to acquire effective representations of input data. This is achieved by encoding the data into a latent space with less dimensions and afterwards decoding it to restore its original form. Convolutional Neural Networks (CNNs), conversely, demonstrate a high degree of suitability in extracting spatial information from organized data, particularly in the context of photographs [6].

In a CNN-based autoencoder, the encoder part typically consists of convolutional layers followed by pooling or down-sampling layers. These layers capture hierarchical features and reduce the spatial dimensions of the input, resulting in a compressed representation. The decoder part of the network mirrors the encoder, using transposed convolutions or up-sampling layers to reconstruct the original input from the compressed representation. The essential advantage of using CNNs in autoencoders is their ability to capture local patterns and spatial relationships in the data. CNN-based autoencoders are particularly effective in image-related tasks, as they can learn to extract hierarchical features from images and reconstruct them with minimal loss .

1) CNN-based encoder architecture : Our work diverges from traditional autoencoders that rely on unsupervised training by emphasizing supervised learning where the model's output is evaluated against a target image. The goal is to produce a clear image from an initially distorted one using a network with five convolutional layers designed for extracting key image characteristics. This involves convolution operations, where a kernel matrix, learned through backpropagation with gradient descent, traverses input data, performing matrix multiplication to create feature maps. The first four convolutions use 64 unique 3x3x1 kernels, resulting in 64 channels per output, while subsequent convolutions employ 3x3x64 kernels. Padding with zeros maintains consistent image dimensions post-convolution. Non-linearity is introduced via the Leaky Rectified Linear Unit (ReLU) activation function. Max pooling compacts information in the encoder, while up-sampling in the decoder restores original image dimensions. Batch normalization layers are incorporated for enhanced model efficiency, and the final output predictions on pixel intensities are shaped by the sigmoid activation function. The model's training objective is to minimize the difference between the original input and reconstructed output using loss functions like mean squared error (MSE) or binary cross-entropy.

D. CycleGAN

CycleGANs are generative models used for unsupervised image-to-image translation, capable of learning domain mappings without paired training data. Introduced for tasks like

artistic style conversion or season-based image transformation, the key feature of CycleGANs is the cycle consistency loss. This ensures content preservation when an image is translated between domains and then reverted. During training, the model optimizes generators to produce realistic images and discriminators to differentiate between real and translated images. The cycle consistency loss further penalizes deviations from the original image after cyclic translations. In essence, CycleGANs offer a consistent and cyclic approach to unsupervised image translation.

1) CycleGAN architecture: The design of Cycle GAN distinguishes itself from other Generative Adversarial Networks (GANs) by incorporating two distinct mapping functions, F and G, that acts as generators and their corresponding discriminators, D_x and D_y.The generator mapping functions are given as a the mapping $G : X \rightarrow Y$ and $F : Y \rightarrow X$ where X is the input image distribution and Y is the desired output distribution. The cost function used is the sum of adversarial loss and cyclic consistent loss:

$$L(G, F, D_x, D_y) = L_{\text{advers}}(G, D_y, X, Y) + L_{\text{advers}}(F, D_x, Y, X) + \lambda L_{\text{cycl}}(G, F, X, Y) \quad (1)$$

with an objective function with the form of:

$$\min_{G,F} \max_{D_x, D_y} L(G, F, D_x, D_y) \quad (2)$$

The CycleGAN generator is composed of three distinct components, namely: 1) Encoder, 2) Transformer, and 3) Decoder. The U-NET architecture will be employed for the generator. In order to construct the generator, we establish our downsample and upsample techniques. The downsampling process decreases the two-dimensional dimensions, specifically the width and height, of the image by a factor known as the stride. The stride refers to the measurement of the distance covered by the filter in a single step. With a stride of 2, the filter is selectively applied to alternate pixels, resulting in a reduction of both the width and height by a factor of 2. In this work, instance normalization was employed as an alternative to batch normalizing. The process of upsampling is characterized by the enlargement of image dimensions, which stands in contrast to downsampling, where image dimensions are reduced. In this context, the Conv2D Transpose layer performs the inverse operation of a Conv2D layer. The architecture of the discriminator employs the PatchGAN discriminator. The PatchGAN is utilized to transform a 256×256 input into a 64×64 output array, where each element represents the authenticity of the corresponding patch in the image. This process involves applying a 4×4 convolution-InstanceNorm-LeakyReLU layer with 128, 256, and 512 filters, and a stride of 2. The application of Instance Normalization on the initial layer consisting of 64 filters is not implemented. Following the final layer, a convolution operation is performed to get a 1×1 output. Our model architecture is illustrated in Figure 2.

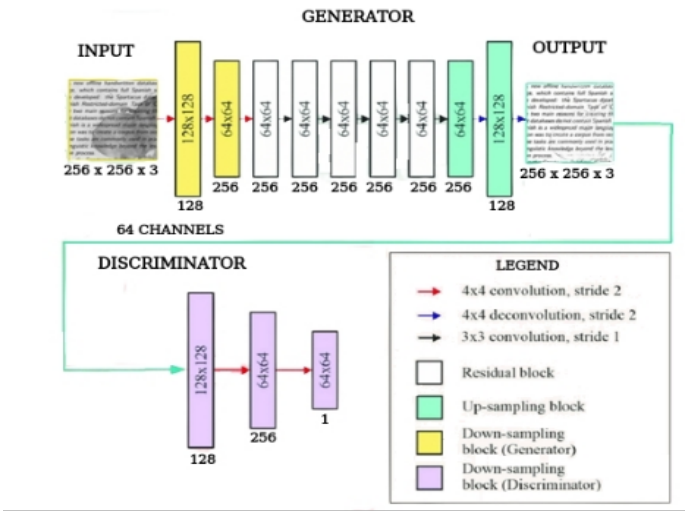

Figure 1. CycleGAN architecture

E. Evaluation metrics

The performance metrics for evaluating the CycleGAN model:

- Fréchet Inception Distance (FID):

$$\text{FID} = |\mu_1 - \mu_2| + \text{Tr}(\sigma_1 + \sigma_2 - 2\sqrt{\sigma_1 * \sigma_2})$$

  where μ_1 and σ_1 refer to the mean and covariance of the train data and μ_2 and σ_2 refer to the mean and covariance of the test data and Tr Tr refers to the trace.
- Inception Score (IS):

$$\text{IS} = e^{\frac{1}{N}\Sigma_{i=1}^{N} D_{KL} p(y|x^i)||\hat{p}(y)}$$

  where $p(y|x)$ is the conditional probability of image being the given object and $p(y)$ is the marginal probability that the given image is real, G refers to the generated image and $D_(KL)$ refers to KL Divergence of the mentioned probabilities.

IV. Results

For the present work, a dataset of 216 text images with background noise is used. The dataset is available publicly from the Kaggle competition "Denoising Dirty Documents Remove noise from printed text" . For the CycleGAN and CNN-based autoencoder a NVIDIA Tesla P100 16GB GPU was used while a 12th Generation Intel Core i7 with 16 GB of RAM was used for the rest of the experiments. From the 216 noisy text images dataset, a median filter with a window size 9x9 was used to obtain the 'background' of our images while preserving the 'foreground,' which contains the text we want to retain. This method proved successful as the background noise typically occupies more area than the text. We utilized median filtering and subtracted the resulting background from the original images. This method obtained an average SSIM of 0.845 with a maximum of 0.889 and a minimum of 0.77. Similarly, a Wiener filter with a window

size 9x9 was applied to the dataset since its use has been reported to obtain better results than filters such as median and mean [2], [3]. This method obtained an average SSIM with a maximum of 0.767 and a minimum of 0.65. The use of an adaptive threshold involved utilizing the local information within the image to compute thresholds for individual images. This approach is similar to the locally adaptive thresholding method, which incorporates neighborhood characteristics such as the mean and standard deviation of pixels. Being slower to apply a global threshold but obtaining a more precise result and overall better performance than the median and Wiener filters. With this method, we obtained an average SSIM of 0.771 with a maximum of 0.837 and a minimum of 0.67. Utilizing a morphological approach, we employed edge detection techniques to discern locations within the image where there is a significant alteration in brightness, afterwards organizing these points into distinct edges. The initial step involves the application of an external morphological gradient using a structuring element in the form of a circular shape with a radius of one and centered at the origin. This gradient is defined as the result of performing a dilation operation on the original image, followed by subtracting the original image itself. The purpose of this operation is to extract the outer edges of the image. Subsequently, it is necessary to remove the peripheral components of the noise. The initial step involves the use of a dilation operation, which serves to increase the thickness of lines by introducing additional pixels along their boundaries. As a consequence, the text is augmented by the process of ”filling in,” but the areas adjacent to the stains retain their empty or hollow appearance. Subsequently, by the implementation of the inverse process known as erosion, it becomes possible to eliminate slender lines entirely while retaining the thicker ones. With this method, we obtain an average SSIM of 0.777 with a maximum of 0.84 and a minimum of 0.68. In contrast to the base CNN model, the CNN stacker employs a CNN-based autoencoder that utilizes data from five distinct channels. These channels consist of the original picture, the output obtained via median filtering, edge detection, adaptive thresholding, and the CNN autoencoder. The stacker model effectively utilizes all five pieces of information in order to provide the ultimate outcome. Two-thirds of the dataset were allocated for training purposes, while one-third was reserved for model validation. The model was trained for 700 epochs with a batch size of 6, with Adam as the optimization algorithm and root-mean squared error (RMSE) as the loss function. With this method, we obtain an average SSIM of 0.952 with a maximum of 0.973 and a minimum of 0.92. In Figure 4, it can be observed the performance of the CycleGAN training.

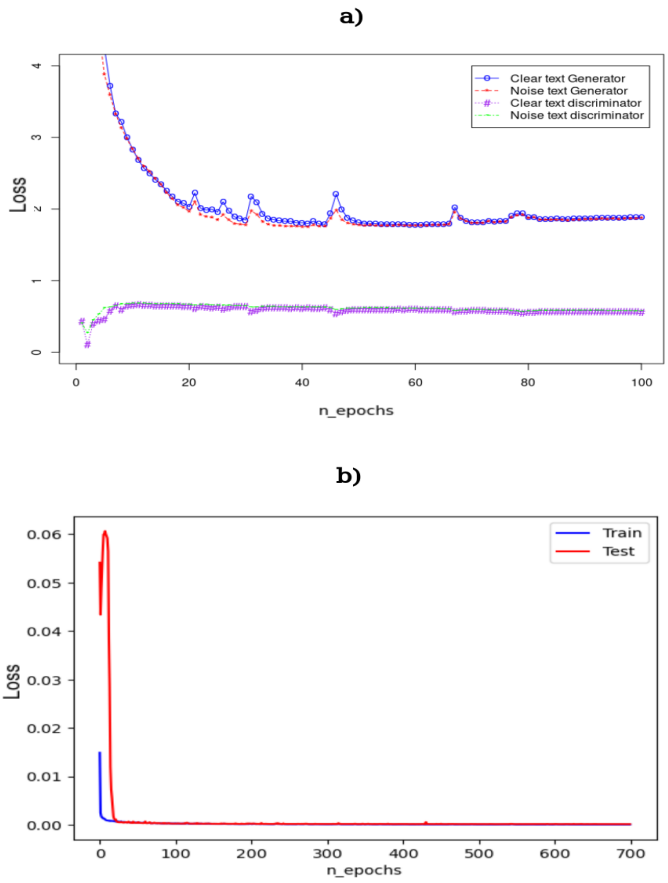

Figure 2. a) Loss function of CycleGAN model, b) Loss function of the CNN-based autoencoder.

A CycleGAN was trained for 100 epochs with denoised text image from which we produce a total of 216 synthetic “denoise” text images from the 216 noisy datasets. For each set of images, its corresponding FID and IS values were calculated as depicted in Table I.

Table I
SCORES OF THE GENERATED IMAGES OF THE CYCLEGAN MODEL.

Number of Epochs	Frechet Inception Distance	Inception Score
25	19.8168	1.145
50	15.4153	1.1458
75	17.4401	1.1522
100	18.3337	1.1511

With this method we obtain an average SSIM of 0.887 with maximum 0.945 and minimum of 0.77. The overall performance is summarized in I.

Table II
QUALITY SUMMARY OF DENOISED IMAGES.

Method	SSIM		PSNR(dB)		RMSE	
	μ	σ	μ	σ	μ	σ
Median	0.845	0.024	19.62	0.988	26.79	3.138
Wiener	0.719	0.028	15.74	0.906	41.82	4.381
Adaptative Threshold	0.771	0.036	14.61	0.894	47.69	4.901
Morphology	0.777	0.034	19.01	1.2	28.83	4.043
CycleGAN	0.887	0.040	23.21	2.238	18.21	4.759
CNN-based autoencoder	0.952	0.013	27.26	1.444	10.57	1.781

*μ: mean; σ: standard deviation

From the overall performance distribution that is shown in 3 we were able to notice that the CNN-based autoencoder gives the best results on average and also with less variability in its output.

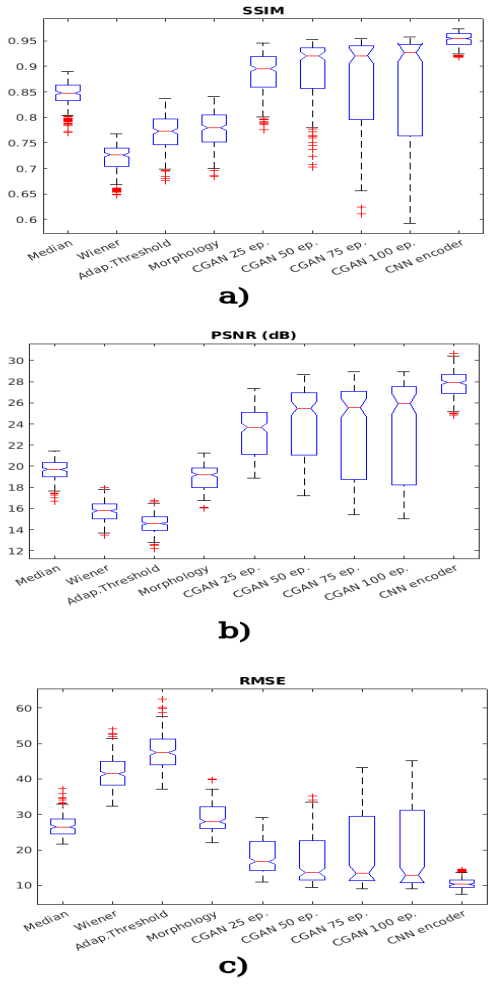

Figure 3. Comparative performance of the method for the 216 evaluated images: a) SSIM measure distribution where a closer value to 1 indicate a better denoising since it is closer to the denoised ideal image of the dataset, b) PSNR measure where a higher value indicates a better image quality, and c) RMSE measure where a lower value indicates a better image fidelity with respect to the ideal denoise image.

The performance of the implemented methods in a some sample images are shown in 4.

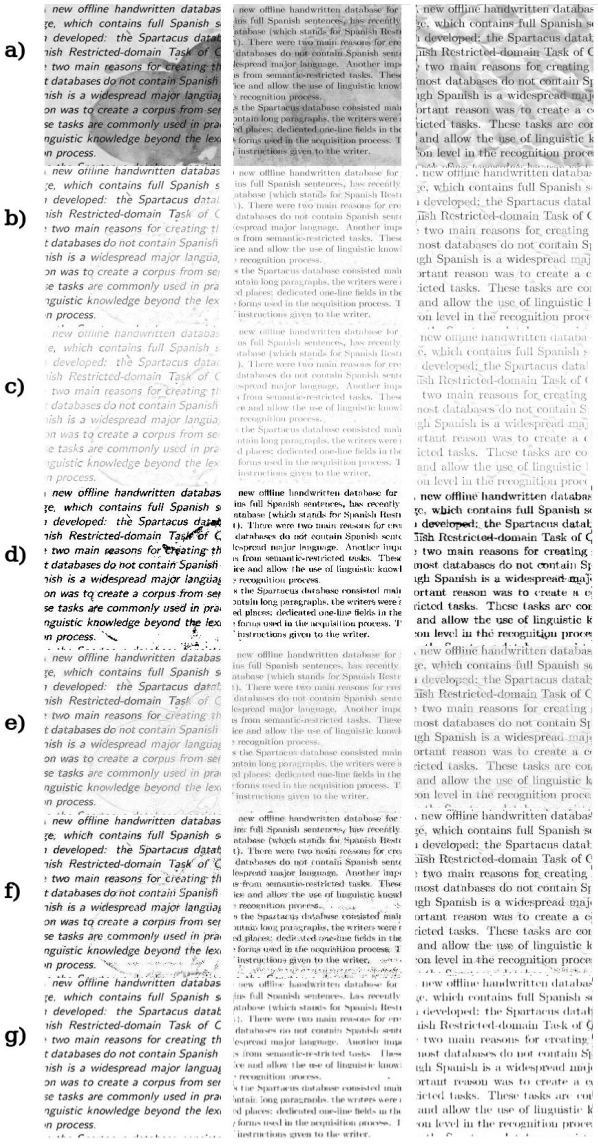

Figure 4. Results of image denoising: a) Image with background noise, b) Median filter, c) Wiener filter, d) Adaptative threshold, e) Morphological filtering, f) CycleGAN 50 epochs, g) CNN-based autoencoder

A. Conclusion & future works

In this work, we performed a comparison between various text denoising methods with respect to a CycleGAN model. We were able to show that the CycleGAN model has a better performance average wise with respect to the methods of median filter, Wiener filter, adaptative threshold and morphological filtering. Although the Cycle GAN model gives us a worse average SSIM of 0.065 with respect to the best performing method of CNN-based autoencoder, still it offered a significant advantage to the the autoencoder approach due to several advantages that were evident in this work. Firstly, data preparation for the training is much simpler in the CycleGAN model due to its property of requiring only unpaired data for training. Also, the number of epochs necessary to obtain

comparable results with the CNN-based autoencoder is much lower, taking as can be seen in Fig. 3 25 to 50 epochs to obtain comparable results to the autoencoder with 700 epochs.

However, one important flaw of the CycleGAN approach which can be seen in Fig. 3 is its high variability in the resulting output, which can be seen in the range of its distribution boxplot. This can be explained by the unsupervised nature of the method, opening the possibility of establishing a reinforced learning approach for this type of GAN models which can also help in the collapse mode problem inherent to this type of models . The use of other generative models such as Conditional GAN, stacked GAN pix2pix and super resolution GAN is sought to be implemented to carry out a comparative analysis based not on measures such as SSIM, PSNR or RMSE but on the number of words that can be extracted from these texts through NLP text extraction methods. The use of more datasets has been proposed to show the versatility of the CycleGAN model for the image enhancement task. It has been proposed the modification of the architecture of this work to 512x512 and 1024x1024 to allow processing images with better quality and in this way it is expected to obtain better results. Some preliminary results of these experiments for this future work with the Noisy and Rotated Scanned Documents datasets from Kaggle, CVC-MUSCIMA for staff elimination of music records, Blur dataset from Kaggle and our own dataset for eliminating lines from handwritten documents can be found in: https://github.com/Lugo1025/ExtendedDenoiseCGAN.

In summary, the use of text cleaning methods is extensive and the CycleGAN method described in this work shows important advantages with respect to the methods with which it was compared and thus is a valuable approach that can be used as pre-processing steps for various tasks such as OCR, natural language processing (NLP), computer vision, text extraction, and segmentation.

B. Dataset & Code

For the present work we use a public dataset of Noisy text images with 2 classes: 216 noisy (background noise) and the same images without noise. The code used in this paper and the dataset is available at: https://github.com/Lugo1025/Denoise-CGAN.

ACKNOWLEDGEMENTS

The authors wish to thank the support of the Instituto Politécnico Nacional (COFAA, SIP-IPN, Grant SIP 20230140) and the Mexican Government (CONAHCyT, SNI).

REFERENCES

[1] R. Rajni and A. Anutam, "Image denoising techniques-an overview," *International Journal of Computer Applications*, vol. 86, no. 16, pp. 13–17, 2014.

[2] A. Farahmand, H. Sarrafzadeh, and J. Shanbehzadeh, "Document image noises and removal methods," 2013.

[3] M. C. Motwani, M. C. Gadiya, R. C. Motwani, and F. C. Harris, "Survey of image denoising techniques," in *Proceedings of GSPX*, vol. 27, 2004, pp. 27–30.

[4] M. Sharma, A. Verma, and L. Vig, "Learning to clean: A gan perspective," in *Computer Vision–ACCV 2018 Workshops: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers 14*. Springer, 2019, pp. 174–185.

[5] Y. Yuan, S. Liu, J. Zhang, Y. Zhang, C. Dong, and L. Lin, "Unsupervised image super-resolution using cycle-in-cycle generative adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2018, pp. 701–710.

[6] A. E. Ilesanmi and T. O. Ilesanmi, "Methods for image denoising using convolutional neural network: a review," *Complex & Intelligent Systems*, vol. 7, no. 5, pp. 2179–2198, 2021.

[7] P. Milanfar, "A tour of modern image filtering: New insights and methods, both practical and theoretical," *IEEE signal processing magazine*, vol. 30, no. 1, pp. 106–128, 2012.

[8] K. Venkatachalam, "A novel algorithm for image denoising using modified adaptive median filter," 2017.

[9] A. S. Yaseen, R. S. Zamel, and J. H. Khlaief, "Wavelet-based denoising of images," *Engineering and Technology Journal*, vol. 37, no. 2, pp. 54–60, 2019.

[10] L. Mennillo, J. Cousty, and L. Najman, "A comparison of some morphological filters for improving ocr performance," in *Mathematical Morphology and Its Applications to Signal and Image Processing: 12th International Symposium, ISMM 2015, Reykjavik, Iceland, May 27-29, 2015. Proceedings 12*. Springer, 2015, pp. 134–145.

[11] L. Yasenko, Y. Klyatchenko, and O. Tarasenko-Klyatchenko, "Image noise reduction by denoising autoencoder," in *2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT)*. IEEE, 2020, pp. 351–355.

[12] K. Bajaj, D. K. Singh, and M. A. Ansari, "Autoencoders based deep learner for image denoising," *Procedia Computer Science*, vol. 171, pp. 1535–1541, 2020.

[13] W. Zuo, K. Zhang, and L. Zhang, "Convolutional neural networks for image denoising and restoration," *Denoising of photographic images and video: Fundamentals, open challenges and new trends*, pp. 93–123, 2018.

[14] Y. Tian, X. Peng, L. Zhao, S. Zhang, and D. N. Metaxas, "Cr-gan: learning complete representations for multi-view generation," *arXiv preprint arXiv:1806.11191*, 2018.

[15] P. Vamsi Kiran Reddy and V. Sajith Variyar, "Image enhancement using gan (a re-modeling of sr-gan for noise reduction)," in *Information and Communication Technology for Competitive Strategies (ICTCS 2020) Intelligent Strategies for ICT*. Springer, 2021, pp. 721–729.

[16] Y. Zhou, S. Zuo, Z. Yang, J. He, J. Shi, and R. Zhang, "A review of document image enhancement based on document degradation problem," *Applied Sciences*, vol. 13, no. 13, p. 7855, 2023.

[17] H. Neji, T. Hamdani, M. Halima, J. Nogueras-Iso, and A. M. Alimi, "Blur2sharp: A gan-based model for document image deblurring," Tech. Rep., 2021.

[18] S. Nigam, A. P. Behera, S. Verma, and P. Nagabhushan, "Deformity removal from handwritten text documents using variable cycle gan," 2022.

[19] M. J. Gangeh, M. Plata, H. R. M. Nezhad, and N. P. Duffy, "End-to-end unsupervised document image blind denoising," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 7888–7897.

[20] P. Jadhav, M. Sawal, A. Zagade, P. Kamble, and P. Deshpande, "Pix2pix generative adversarial network with resnet for document image denoising," in *2022 4th International Conference on Inventive Research in Computing Applications (ICIRCA)*. IEEE, 2022, pp. 1489–1494.

[21] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.