Noise Simulation for the Improvement of Training Deep Neural Network for Printer-Proof Steganography

Telmo Cunha[1], Luiz Schirmer[2], João Marcos[1] and Nuno Gonçalves[1,3]
[1]*Institute of Systems and Robotics, University of Coimbra, Coimbra, Portugal*
[2]*University of Vale do Rio dos Sinos, São Leopoldo, Brazil*
[3]*INCM Lab, Portuguese Mint and Official Printed Office, Lisbon, Portugal*

Keywords: Printer-Proof Steganography, Noise Simulation, Deep Learning, GAN.

Abstract: In the modern era, images have emerged as powerful tools for concealing information, giving rise to innovative methods like watermarking and steganography, with end-to-end steganography solutions emerging in recent years. However, these new methods presented some issues regarding the hidden message and the decreased quality of images. This paper investigates the efficacy of noise simulation methods and deep learning methods to improve the resistance of steganography to printing. The research develops an end-to-end printer-proof steganography solution, with a particular focus on the development of a noise simulation module capable of overcoming distortions caused by the transmission of the print-scan medium. Through the development, several approaches are employed, from combining several sources of noise present in the physical environment during printing and capture by image sensors to the introduction of data augmentation techniques and self-supervised learning to improve and stabilize the resistance of the network. Through rigorous experimentation, a significant increase in the robustness of the network was obtained by adding noise combinations while maintaining the performance of the network. Thereby, these experiments conclusively demonstrated that noise simulation can provide a robust and efficient method to improve printer-proof steganography.

1 INTRODUCTION

Nowadays, images have emerged as potent conveyors of information and knowledge, an interesting characteristic for both researchers and industries.

Steganography is a process that hides information within a common object. This method allows to hide sensitive information from unauthorized access, ensuring confidentiality as well as covering communication. With the addition of deep learning and machine learning methods, it is possible to enhance the steganography method for a more robust and reliable application. Examples of these methods are end-to-end solutions, such as HiDDeN (Hsu and Wu, 1999), SteganoGAN (Zhang et al., 2019), StegaStamp (Tancik et al., 2020), and CodeFace (Shadmand et al., 2021), where they provide a robust and secure method for hiding messages within digital images with the utilization of different methods, such as GANs or other networks, that take advantage of the use of an encoder and decoder to improve the concealed information within images. One example of the use of this application is exemplified in Figure 1, which uses the concept of steganography to increase the security of identity documents. However, these solutions, present some drawbacks. One of the issues is the limitation of their robustness. These end-to-end solutions are vulnerable to distortions that occur during the printing and scanning processes, and also show limitations when subjected to extreme compression and filtering.

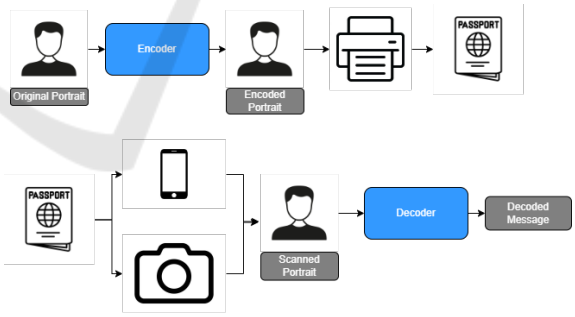


Figure 1: General pipeline of an end-to-end steganography solution with the purpose of improving the security measures of documents. Based on (Shadmand et al., 2021).

To overcome the limitation of robustness, in this paper we propose the improvement of printer-proof

Cunha, T., Schirmer, L., Marcos, J. and Gonçalves, N.
Noise Simulation for the Improvement of Training Deep Neural Network for Printer-Proof Steganography.
DOI: 10.5220/0012272300003654

In *Proceedings of the 13th International Conference on Pattern Recognition Applications and Methods (ICPRAM 2024)*, pages 179-186
ISBN: 978-989-758-684-2; ISSN: 2184-4313

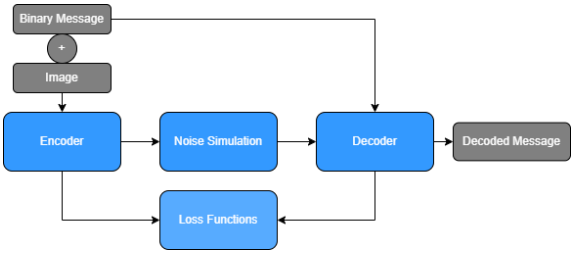


Figure 2: Overall architecture of the end-to-end Steganography solution.

steganography solutions through the realization of noise simulations. In this paper, it is approached an end-to-end solution that uses GAN (Goodfellow, 2017; Goodfellow et al., 2014; Ledig et al., 2017), such as StegaStamp (Tancik et al., 2020). In this steganography solution, GANs have the purpose of hiding secret messages in the images while maintaining the same appearance. As can be seen in Figure 2, the general architecture is composed of four components: the encoder, decoder, noise simulation module, and loss functions. The main goal of the encoder is to hide messages in images. The decoder is incorporated into the whole architecture after applying the noise to the images and is designed to recover the hidden message. The noise simulation module allows for the simulation of the several noise sources that occur during the printing and scanning processes to train the network.

This research, through rigorous experimentation and analysis, provides new insights into the behavior and limitations of the noise simulation. Careful adjustments to the noise simulation module, involving the addition of various noise levels, improved both model performance and robustness. This approach effectively addresses noise in print-scan environments.

2 RELATED WORK

2.1 Image Steganography

Image steganography methods consist of hiding the existence of a secret message, audio, image, or video into a cover image, in a way that encoded, and cover images are not distinguishable from each other. The state-of-the-art methods that take advantage of traditional methods without the use of deep learning are discussed in (Pevný et al., 2010). This work focuses on deep learning-based steganography techniques.

Image steganography has seen significant advancements in recent years, with new techniques improving the robustness and security of steganography. These techniques are based on opposing networks (mainly GANs) to encode and decode information. The most relevant methods to achieve this approach are SteganoGAN (Zhang et al., 2019), HiDDeN (Hsu and Wu, 1999), StegaStamp (Tancik et al., 2020), and CodeFace (Shadmand et al., 2021). All the techniques mentioned add a noise simulation network to improve the ability to recover images with distortion. The HiDDeN noise simulation component is implemented between the encoder and the decoder. The authors propose the noise simulation for a discrete cosine transform, a JPEG compression, a JPEG-Mask, and a JPEG-Drop as distortion types for generating the noise samples. However, the noise simulation modules in SteganoGAN and HiDDeN have a rather simple formulation, and they do not entirely consider other noise sources introduced by physical printing and capturing with a digital camera.

StegaStamp (Tancik et al., 2020) was the first successful example of steganography with printed images, showing a robust decoding message under physical transmission. The noise distortions, aimed at approaching the printing process, are composed of Gaussian noise, transformations of color manipulation by printers, such as random constant, brightness, random affine color, and Hue shift distortions, and lastly, JPEG compression. Nevertheless, StegaStamp has some limitations, namely the possibility of a pattern (originating from the hidden message) becoming perceptible in large low-frequencies regions of the image and the excessive noise present in the encoded image when compared to the original image.

CodeFace (Shadmand et al., 2021) introduces a novel deep learning printer-proof steganography approach for document security systems. This new approach was inspired in StegaStamp model and introduces a new security system for encoding and decoding facial images that are printed on common identity documents. The noise simulation module is based on StegaStamp (Tancik et al., 2020) and HiDDen (Hsu and Wu, 1999). The resize network (that performs downsampling of the input image) enables the decoder to read a message from small face images in the decoding process. This end-to-end solution has introduced several new contributions in the field of steganography, namely it improved the perceptual quality of the encoded image and its compliance with modern FRS and document issuing requirements.

2.2 Noise Simulation

Noise simulation is a widely used method by researchers to assess image processing algorithms under realistic conditions in various fields of research. It involves introducing artificial noise into digital im-

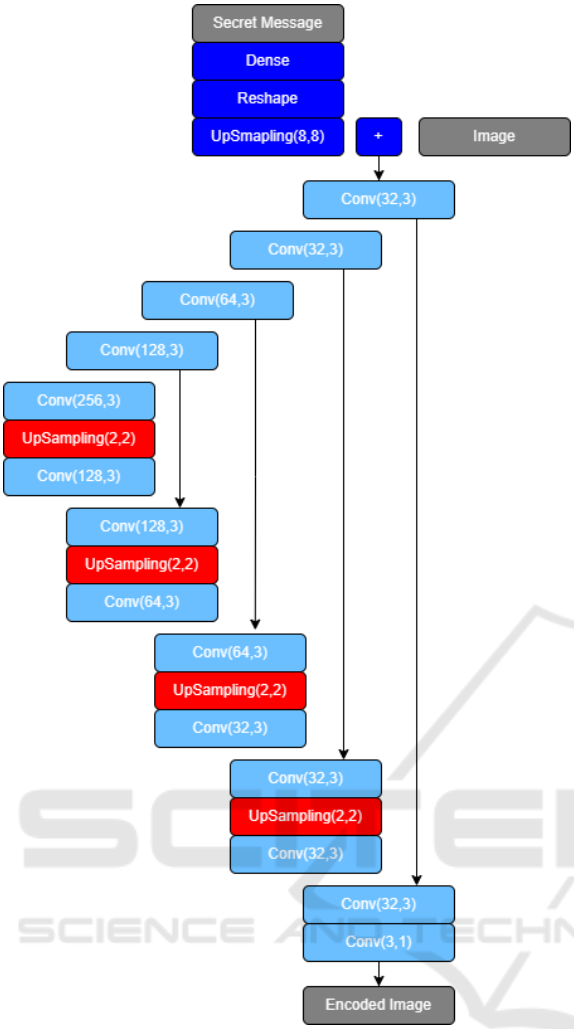


Figure 3: The encoder network is based on a U-net network with no pooling layers.

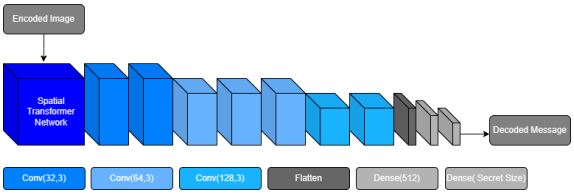


Figure 4: The decoder network is composed of an STN module followed by a CNN.

ages or signals to simulate real-world noise effects. This approach evaluates algorithm resilience to varied environmental conditions in image processing, and it is also used in deep learning for enhancing neural network robustness. The state-of-the-art that leverage traditional methods are discussed in (Misra and Wu, 2020), which explores the use of Gaussian Blur and in (Raid et al., 2014), an approach predating deep learning that reviews JPEG compression with the use of Discrete Cosine Transform.

3 IMPLEMENTATION DETAILS

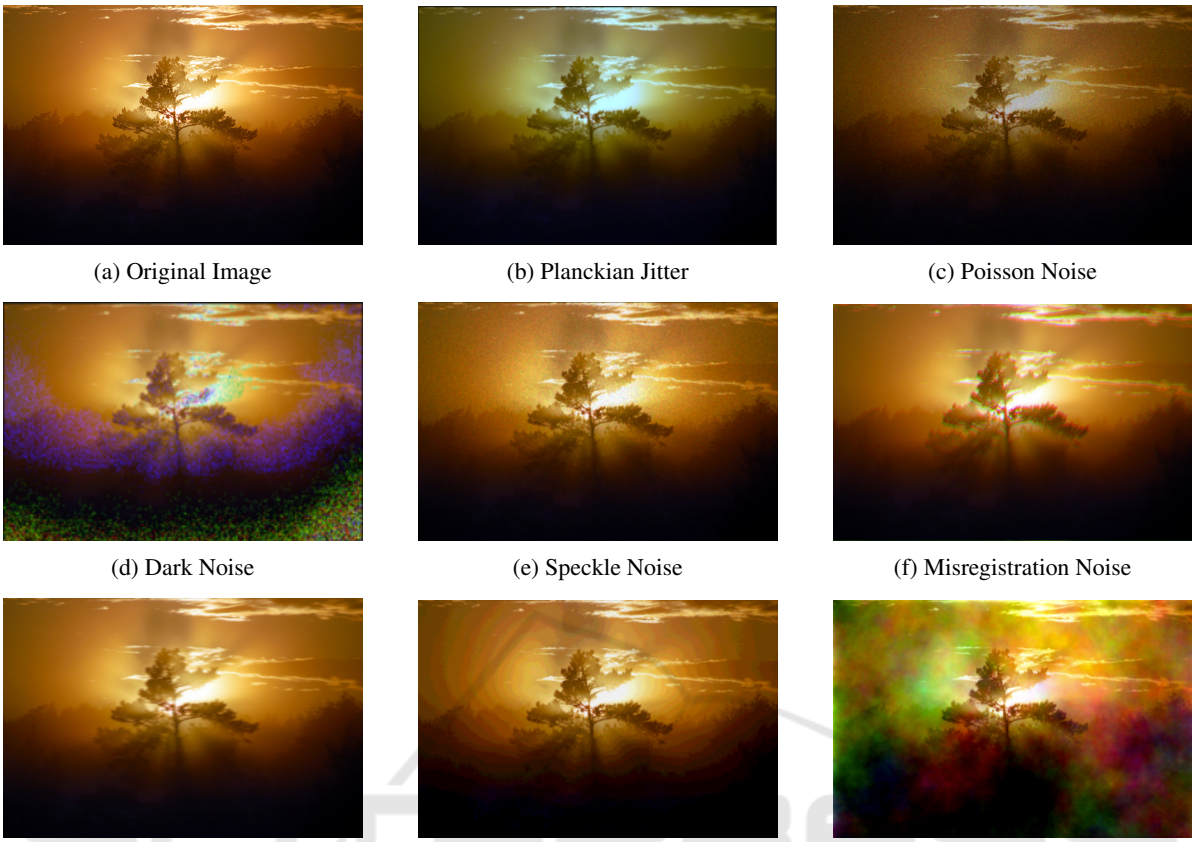
3.1 Baseline

The work and study performed throughout the development of the research was performed on the StegaStamp algorithm with some modifications. Compared to the original model, the modified algorithm does not use the Detector component, and some parameters and activation functions of the networks are different. The architecture of the algorithm can be described by the following components: an encoder, a spatial transformer network, a decoder, and a discriminator. The encoder has the main objective to embed a message into an image while minimizing perceptual differences between the input and encoded image. For this, the network of the encoder is based on a U-Net style architecture (Ronneberger et al., 2015) that receives a $400 * 400 * 3$ pixel cover image and a 100 bits secret message as input and generates an encoded residual image at the output, (see Figure 3). The decoder has the main goal to recover the hidden message from the encoded image. The decoder architecture, shown in Figure 4 is composed of a Spatial Transformer Network (STN) (Jaderberg et al., 2015) followed by a CNN (Convolutional Neural Network). The STN component develops robustness against small perspective changes that are introduced while capturing and rectifying the encoded image. Lastly, the discriminator, as the name suggests, has the objective of distinguishing between real and fake images. The discriminator network is composed of five convolutional layers with a kernel size of three, where each one is followed by the ReLU activation function with the exception of the final layer. Furthermore, with the addition of the Dual Contrasctive loss, a final linear layer was added, since the addition of Dual Contrastive loss, requires the use of contrastive learning with adversarial learning.

3.2 Noise Simulation Module

For simplicity reasons, the following Figure 5 represents examples of different noises sources implemented in the noise simulation module.

Planckian Jitter. Planckian Jitter (Zini et al., 2023), represented in Figure 5b, has the aim to simulate the thermal noise that can occur in image systems, allowing the model to be more robust to illumination



(a) Original Image (b) Planckian Jitter (c) Poisson Noise

(d) Dark Noise (e) Speckle Noise (f) Misregistration Noise

(g) Motion Blur (h) Posterization (i) Plasma Brightness

Figure 5: Examples of the different sources of noise integrated in the noise simulation module.

changes. One example of the presence of this distortion is when the color of the paper has a bluish or orange tone the moment a picture is taken. To perform this modification to an image, the method exploits Planck's law, which describes the spectral radiance of an ideal black body at a given temperature.

Poisson Noise. Poisson noise (Hasinoff, 2014), aka Photon noise, is a basic form of uncertainty associated with the measurement of light, inherent to its quantized nature and the independence of photon detection, as shown in Figure 5c. Its expected magnitude varies with signal strength and is the primary source of image noise in light conditions. This noise arises from the randomness of individual photon arrivals, a signal-dependent uncertainty inherent to the signal.

Dark Noise. The dark noise (Hui, 2020), shown in Figure 5d, can be defined as a random variation of the dark current signal since it results from statistical fluctuations in the number of thermally generated electrons, which contributes to the uncertainty in the dark current value at a given pixel location. The dark current refers to the electric current that flows through a semiconductor device, such as a CCD or CMOS sensor (used in digital cameras and smartphones), even in the absence of light.

Speckle Noise. Speckle noise (Arulpandy and Pricilla, 2020), presented in Figure 5e, is a granular noise texture that degrades the quality of an image as a consequence of the interference among wavefronts in imaging systems. The speckle effect is a result of the interference of many waves of the same frequency with different phases and amplitudes, with a resultant wave whose amplitude and therefore intensity vary randomly. Unlike other types of noise, speckle noise causes uneven pixel distribution.

Misregistration Noise. The misregistration noise (Townshend et al., 1992) simulates the noise that arises from the misalignment of image channels, as can be seen in Figure 5f. This type of noise can occur for several reasons and factors in the registration process or printing process, since perfect alignment may not be achieved. One of the causes could be camera movement, such as small movements and vibrations during image capture, or variations in focal lengths and camera settings. Another factor stems from issues during the printing process.

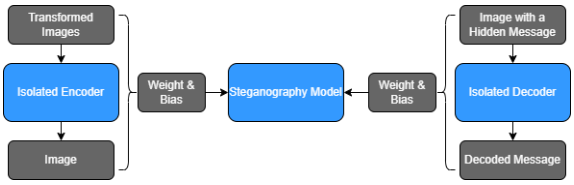


Figure 6: Self-supervised learning approach performed for this model.

Motion Blur. Motion blur (Karl, 2005; Kurimo et al., 2009), shown in Figure 5g, is a well-known noise and is one of the most significant reasons for image quality to decrease. Motion blur is a common optical effect in photographs and videos that occurs when the positions of objects change with respect to the camera, during the interval in time where the camera shutter is open. If the objects are moving rapidly or the shutter interval is long enough, then the objects leave a blurred streak in the direction of motion.

Posterization. Image posterization has the aim of converting an image that has a large number of tones into an image with distinct flat areas, reducing the tones, giving it a simplified and graphical appearance, such as a poster or painting (see Figure 5h).

Plasma Brightness. Plasma Brightness noise (Nicolaou et al., 2022) refers to a random brightness variation that affects an image, resulting in an image with random and visually patterns with a particular appearance (see Figure 5i). This distortion allows to simulate the effect of RGB spots or visible bands of colors that resemble rainbows or gradient transitions that comes from the limitation of colors in printers.

3.3 Self-Supervised Learning

Self-Supervised learning, a subcategory of unsupervised learning, leverages unlabeled data. The key idea is to allow the model to learn data representation without manual labels (refer to the discussion in (Ericsson et al., 2022)). For the use of this technique, it was implemented in two important components of the model architecture, which are the encoder and decoder.

The encoder network primary function is to embed a secret message within an image. Given that the encoder is composed of a U-net architecture, it is chosen for an information restoration task by reconstructing the original image from its transformed version. In contrast, the decoder network is responsible for extracting the concealed message. Thus, a self-supervised learning approach is employed to enhance the decoding process, utilizing a dataset containing hidden messages. See Figure 6.

3.4 Metrics

To perform a better evaluation of the performance, two metrics evaluate the quality of images during the process of image generation with a hidden message.

Structural Similarity Index (SSIM) (Wang et al., 2004; Nilsson and Akenine-Möller, 2020) evaluates the structural information, luminance, and contrast similarities between the reference image and the distorted image. Unlike traditional methods measuring pixel differences, SSIM aligns with human visual perception, focusing on identifying structural information and differences based on extracted information.

Peak Signal-to-Noise Ratio (PSNR) (Horé and Ziou, 2010) evaluates the quality of an image by comparing its pixels to those of a reference image, and it provides insights into the amount of noise or distortion present in the image.

The decoding rate (DR) is a simple metric used to quantify the number of encoded images that were successfully decoded, thereby measuring the efficacy of the hidden message retrieval process.

3.5 Datasets

Throughout the development of the research, we used the MIRFLICKR dataset to train the StegaStamp model. The MIRFLICKR dataset (Huiskes and Lew, 2008), which is composed of 25,000 images, is a dataset with a wide-range of diversity, covering a variety of categories including humans, animals, urban landscapes, and more.

Furthermore, in order to apply self-supervised learning to the model, specifically to create a task to use in the decoder network, we use the JMiPOD dataset (Cogranne et al., 2020). The JMiPOD dataset is designed for steganalysis, which, in short, is the process of detecting hidden information within digital media. This dataset is composed of modified JPEG images where the compression technique has been changed to incorporate hidden data.

4 EXPERIMENTS

The overall development and testing of the study conducted in this paper were performed in a digital environment, encompassing all the stages of the steganography process on digital devices. The images used for encoding and decoding maintained the same dimensions as those employed in the training phase, measuring 400*400 pixels. However, it is worth noting that the resolutions differed. During training and encoding stages, input images had resolutions varying

from 72 to 300 dpi (dots per inch). On the other hand, the input image for the decoding process consistently had a resolution of 96 dpi, obtained through the encoding process. This resolution is suitable for use in digital and print applications.

Before analyzing the results and the influence of each noise described in Section 3, it is presented the baseline to provide a basis for assessing the effectiveness and resilience of the techniques incorporated in this study, as shown in the following Table 1.

Table 1: Decoding rate for the baseline result.

Test	Epochs	Decoding rate
Base	140,000	70.3%

The results obtained during the evaluation of each individual noise were satisfactory as present in Table 2. From this set of results, it is possible to observe that the nature and influence of each individual noise bring a modest to a higher increase in model performance as well as robustness, showing an increase of $4pp$ to $14pp$ in the decoding rate metric. By looking at the values of the SSIM and PSNR metrics, it becomes feasible to evaluate the overall quality of the images generated by these models relative to the base model. The overall values of SSIM are below $0,70$, which indicates noticeable differences between images. However, certain values surpass this threshold, meaning images with good quality and fewer deviations from the images produced by the base model. Regarding the PSNR values, two observations may arise. Values in the range of $[50,60]$ indicate images with acceptable quality, suggesting a presence of degradation in image aspects. Conversely, values above 60 are indicative of images with good quality. If the values obtained were below 50, it would suggest a significant presence of degradation in the images.

From the previous results, it is possible to observe what noises increase the robustness of the model; however, it is not knowledgeable how the model will behave in the presence of several noises, demanding for a long process to investigate the right influence of each noise on an image and its position on the noise module. Table 3 presents the results of three different groups of noise combinations. Group 1 is composed of Posterization, Planckian Jitter, and Poisson noise, in this particular order. Group 2 is formed by Posterization, Planckian Jitter, Misregistration and Poisson noise. Group 3 consists of Posterization, Planckian Jitter, Misregistration and Poisson noise and Motion Blur. With the increase in the number of noises that are incorporated into the model, the performance of the model decreases, justified by the decoding rate metric. Thus, it is possible to conclude that the increase in different noise sources is accompanied by a reduction in the model overall performance and decoding rate, indicating that the model can decode messages with simpler noises, while struggling to decode the hidden message with more complex and sophisticated combinations of noise. Nevertheless, with larger noise combinations, the model attains heightened robustness, stemming from its exposure to diverse noise sources during training.

Table 2: Metrics of each individual noise.

	Noise		
	Posterization	Planckian Jitter	Poisson
DR	75.1%	78.7%	79.9%
ΔBase	↑ 4.8 pp	↑ 8.4 pp	↑ 9.6 pp
SSIM	0.71	0.72	0.68
PNSR	52.0	52.3	51.6
	Dark	Plasma Brightness	Motion Blur
DR	77.1%	74.8%	78.7%
ΔBase	↑ 6.8 pp	↑ 4.5 pp	↑ 8.4 pp
SSIM	0.69	0.68	0.70
PNSR	63.7	63.5	64.1
	Speckle	Misregistration	
DR	80.0%	84.6%	
ΔBase	↑ 9.7 pp	↑ 14.3 pp	
SSIM	0.69	0.69	
PNSR	63.9	51.8	

Table 3: Metrics of the set of noise combinations.

Group	Epochs	DR	ΔBase	SSIM	PSNR
1	140,000	81.4%	↑ 11.1 pp	0.71	64.4
	180,000	78.3%	↑ 8.0 pp	0.69	51.7
	140,000	83.6%	↑ 12.3 pp	0.69	57.9
2	180,000	81.3 %	↑ 11.0 pp	0.69	63.9
3	160,000	79.1%	↑ 8.8 pp	0.71	51.8
	180,000	74.6%	↑ 4.3 pp	0.70	51.9

To overcome the limitation mentioned, data augmentation techniques was used, to enhance the performance while maintaining robustness. The dataset was increased to its double in the first approach. By looking at Table 4, the increase in the decoding rate was modest, increasing approximately $2pp$.

Table 4: Results from data augmentation with double size, performed with the best results of the noise groups.

Group	DR	ΔBase	SSIM	PSNR
1	83.0 %	↑ 12.7 pp	0.73	52.4
2	82.3 %	↑ 12.0 pp	0.69	51.9
3	81.7 %	↑ 11.4 pp	0.70	52.0

Thus, a subsequent expansion of the dataset was performed, intended to increase the number of samples to 100,000 and 200,000. From the obtained results presented in Table 5 it was noticed that there was an insignificant increase in the decoding rate for the dataset with 100,000 samples. On the other hand, the other dataset shows a decrease in the metric. This occurrence points to an instance of overfitting brought on by an inadequately diversified dataset.

Table 5: Result of data augmentation with a total size of 100.00 and 200.000 samples using Group 1.

Dataset	DR	ΔBase	SSIM	PSNR
100,000	83.7 %	↑ 13.4 pp	0.71	52.1
200,000	80.0 %	↑ 9.7 pp	0.73	52.4

Hence, it is possible to conclude that, using conventional methods, the overall performance of the model and decoding rate yield a small growth. To overcome this impasse, another data augmentation technique was implemented. For this, it was used neural style transfer (Gatys et al., 2015). In short, it generates a new image by extracting the content and style of different images from a pre-trained deep neural network. This method allows to mitigate the problem of a dataset with insufficient diversity. Nonetheless, when testing this technique, it was noticed that it was not suitable for the task at hand since the quality displayed by the created images was of lower quality or exhibited structures that were unsuitable for the problem of this research, as shown in Figure 7.

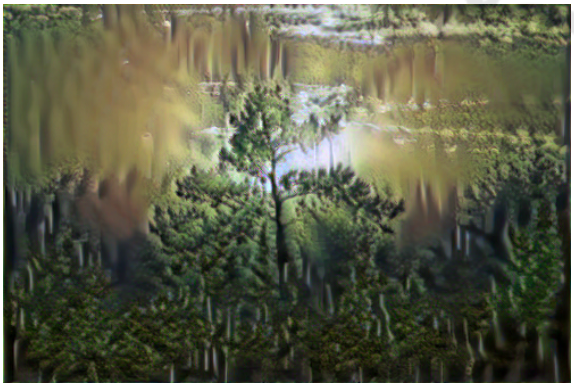


Figure 7: Image created with the use of the method of neural style transfer. Original image is the same used in Fig. 5.

In this way, in an effort to improve the model performance, self-supervised learning was introduced. It is important to note that the implementation of SSL is in its initial stages of development. Looking at Table 6, it is possible to affirm that the use of SSL for the encoder yields satisfactory results since, in an initial approach, it was obtained with a value close to its counterpart. Besides the fact that the value obtained for the metric is lower, the reason for this result is the suboptimal selection of parameters for the use of SSL.

Table 6: Result of SSL with pre-training the encoder, using the noise combination of Group 1 (Original result present in Table 3, with decoding rate of value 83.6%).

Decoding rate	ΔBase	SSIM	PSNR
82.7 %	↑ 12.4 pp	0.71	58.3

On the other hand, some hurdles were encountered during the implementation of SSL on the decoder, stemming from two main reasons. The first reason comes from the used dataset. While the dataset consisted of images with hidden messages, the size of each hidden message varied. The second reason may be caused by the inadequate performance of the message retrieval process. In assessing the model performance and accuracy, it becomes essential to have prior knowledge of the hidden message. Consequently, adopting a method for retrieval message could lead to incorrect interpretation of the retrieved message, thereby jeopardizing the training process. This aspect represents one of the future areas intended for further development and evaluation.

5 CONCLUSIONS

In this paper, a path is presented to improve the robustness of printer-proof steganography solutions. Within the approach presented, we have not only enhanced the noise simulation module but also improved the model performance in the face of increasing robustness against diverse real-word noise sources. Furthermore, through the implementation of data augmentation techniques and deep learning methods, such as SSL, the model performance has improved while maintaining its robustness. The achievements presented offer an effective path for several applications in the real world, such as the security measures of documents.

During the development of this study, promising grow paths emerge, notably with SSL. In comparison with the state-of-the-art approaches, the use of SSL is a novel approach. In this paper, this technique is in its initial stages, demonstrating a potential path for development. This work is one of many approaches to improve printer-proof steganography, and there are many open challenges and opportunities for future research in this field.

ACKNOWLEDGEMENTS

This work has been supported by Fundação para a Ciência e a Tecnologia (FCT) under the project UIDB/00048/2020 - DOI 10.54499/UIDB/00048/2020.

REFERENCES

Arulpandy, P. and Pricilla, M. (2020). Speckle noise reduction and image segmentation based on a modified mean filter. *Computer Assisted Methods in Engineering and Science*, 27(4).

Cogranne, R., Giboulot, Q., and Bas, P. (2020). Alaska#2: Challenging academic research on steganalysis with realistic images. In *2020 IEEE International Workshop on Information Forensics and Security (WIFS)*.

Ericsson, L., Gouk, H., Loy, C. C., and Hospedales, T. M. (2022). Self-supervised representation learning: Introduction, advances, and challenges. *IEEE Signal Processing Magazine*, 39(3).

Gatys, L. A., Ecker, A. S., and Bethge, M. (2015). A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*.

Goodfellow, I. J. (2017). NIPS 2016 tutorial: Generative adversarial networks. *arXiv preprint arXiv:1701.00160*.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial networks. *Advances in neural information processing systems*, 27.

Hasinoff, S. W. (2014). Photon, poisson noise. In Ikeuchi, K., editor, *Computer Vision: A Reference Guide*. Springer US.

Horé, A. and Ziou, D. (2010). Image quality metrics: Psnr vs. ssim. In *2010 20th International Conference on Pattern Recognition*.

Hsu, C.-T. and Wu, J.-L. (1999). Hidden digital watermarks in images. *IEEE Transactions on Image Processing*, 8(1).

Hui, R. (2020). Chapter 4 - photodetectors. In Hui, R., editor, *Introduction to Fiber-Optic Communications*. Academic Press.

Huiskes, M. J. and Lew, M. S. (2008). The mir flickr retrieval evaluation. In *2008 ACM International Conference on Multimedia Information Retrieval*. ACM.

Jaderberg, M., Simonyan, K., Zisserman, A., and Kavukcuoglu, K. (2015). Spatial transformer networks. *Advances in neural information processing systems*, 28.

Karl, W. (2005). 3.6 - regularization in image restoration and reconstruction. In BOVIK, A., editor, *Handbook of Image and Video Processing (Second Edition)*, Communications, Networking and Multimedia. Academic Press, second edition edition.

Kurimo, E., Lepistö, L., Nikkanen, J., Grén, J., Kunttu, I., and Laaksonen, J. (2009). The effect of motion blur and signal noise on image quality in low light imaging. In Salberg, A.-B., Hardeberg, J. Y., and Jenssen, R., editors, *Image Analysis*. Springer Berlin Heidelberg.

Ledig, C., Theis, L., Huszar, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., and Shi, W. (2017). Photo-realistic single image super-resolution using a generative adversarial network. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. arXiv.

Misra, S. and Wu, Y. (2020). Chapter 10 - machine learning assisted segmentation of scanning electron microscopy images of organic-rich shales with feature extraction and feature ranking. In Misra, S., Li, H., and He, J., editors, *Machine Learning for Subsurface Characterization*. Gulf Professional Publishing.

Nicolaou, A., Christlein, V., Riba, E., Shi, J., Vogeler, G., and Seuret, M. (2022). Tormentor: Deterministic dynamic-path, data augmentations with fractals. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Nilsson, J. and Akenine-Möller, T. (2020). Understanding ssim. *arXiv preprint arXiv:2006.13846*.

Pevný, T., Filler, T., and Bas, P. (2010). Using high-dimensional image models to perform highly undetectable steganography. In Böhme, R., Fong, P. W. L., and Safavi-Naini, R., editors, *Information Hiding*. Springer Berlin Heidelberg.

Raid, A. M., Khedr, W. M., El-dosuky, M. A., and Ahmed, W. (2014). Jpeg image compression using discrete cosine transform - a survey. *arXiv preprint arXiv:1405.6147*.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference*.

Shadmand, F., Medvedev, I., and Gonçalves, N. (2021). Codeface: A deep learning printer-proof steganography for face portraits. *IEEE Access*, 9.

Tancik, M., Mildenhall, B., and Ng, R. (2020). Stegastamp: Invisible hyperlinks in physical photographs. In *IEEE/CVF conference on computer vision and pattern recognition*.

Townshend, J., Justice, C., Gurney, C., and McManus, J. (1992). The impact of misregistration on change detection. *IEEE Transactions on Geoscience and Remote Sensing*, 30(5).

Wang, Z., Bovik, A., Sheikh, H., and Simoncelli, E. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4).

Zhang, K. A., Cuesta-Infante, A., Xu, L., and Veeramachaneni, K. (2019). Steganogan: High capacity image steganography with gans. *arXiv preprint arXiv:1901.03892*.

Zini, S., Gomez-Villa, A., Buzzelli, M., Twardowski, B., Bagdanov, A. D., and van de Weijer, J. (2023). Planckian jitter: countering the color-crippling effects of color jitter on self-supervised training.