StampOne: Addressing Frequency Balance in Printer-proof Steganography

Farhad Shadmand *
ISR-UC[1]
farhad.shadmand@isr.uc.pt

Iurii Medvedev
ISR-UC[1]
iurii.medvedev@isr.uc.pt

Luiz Schirmer
ISR-UC[1], Unisinos[2]
luizschirmer@unisinos.br

João Marcos
ISR-UC[1]
joao.marcos@isr.uc.pt

Nuno Gonçalves
ISR-UC[1]
nunogon@deec.uc.pt

[1] Institute of Systems and Robotics, University of Coimbra, Portugal
[2] University of the Sinos River Valley Rio de Janeiro, Brazil

Abstract

Robust steganography and invisible watermarking techniques in printed images are crucial for anti-counterfeiting systems within the multimedia industry for copyright protection, security of documents (e.g. passports), and brand protection graphic elements. Conventional steganography models, mainly designed for digital non-lossy media, encounter challenges in recovering messages from images degraded by printing and scanning or social media compression, particularly due to limitations associated with utilizing image regions characterized by the lowest and highest frequencies. In this paper we introduce StampOne, a novel printer-proof steganography model utilizing Generative Adversarial Networks (GANs). StampOne ensures balanced frequency density between encoder and decoder inputs, reducing disparities between original and encoded images. Our method, through integration with diverse U-shape networks (image-to-image), emphasizes the significance of frequency domain analysis in robust steganography. It facilitates the development of robust steganography models capable of withstanding diverse noise types, including JPEG compression, contrast variations, brightness fluctuations, aliasing, blurring, and Gaussian noises. It surpasses previous models in both quality of encoded images and printer-proof capabilities.

1. Introduction

The proliferation of generated (fake) content in digital, and also physical, media, brought not only opportunities but also many threats and challenges to the society. Particularly in physical objects, fake content can be used for the manipulation of ID documents (passports, driver's license among others) by attacking the document's portrait [6, 12, 24]. Other examples of manipulation of printed media with fake generated content include the proliferation of fake (printed) news or the attack on brand protection labels.

The integration of image watermarking and steganography [10, 36, 37] presents a promising avenue for addressing this problem, facilitating the robust embedding of an invisible signature within an image. Portable mobile devices can then be used for a 1st-level forensic verification of the integrity of the signature, and thus the veracity of the scanned image.

While our work shares similarities with both image watermarking and steganography, our primary focus lies specifically within the domain of steganography. Image steganography is a technique for concealing a confidential message within a cover image or video, while ensuring that the encoded content remains indistinguishable to the human eye from the original. We can classify steganography into two categories: robust and non-robust. Robust steganography models are capable of withstanding printer-scan and/or digital noise, whereas non-robust models are designed for noise-free digital environments. Furthermore, the main challenge of robust steganography models is maintaining high perceptual quality of the encoded images, since the necessary changes to the image at the pixel level, usually referred to as artifacts, need to be stronger to resist to the degradation imposed by the transmission channel.

Existing steganography models have several limitations mainly concerning the size of the embedded message, the level of similarity between encoded and real photos, decoding accuracy, and resistance against fraudulent techniques. Moreover, in robust steganography, the choice of neural net-

*This work has been supported by Fundaçao para a Ciência e a Tecnologia (FCT) under the project $UIDB/00048/2020$.

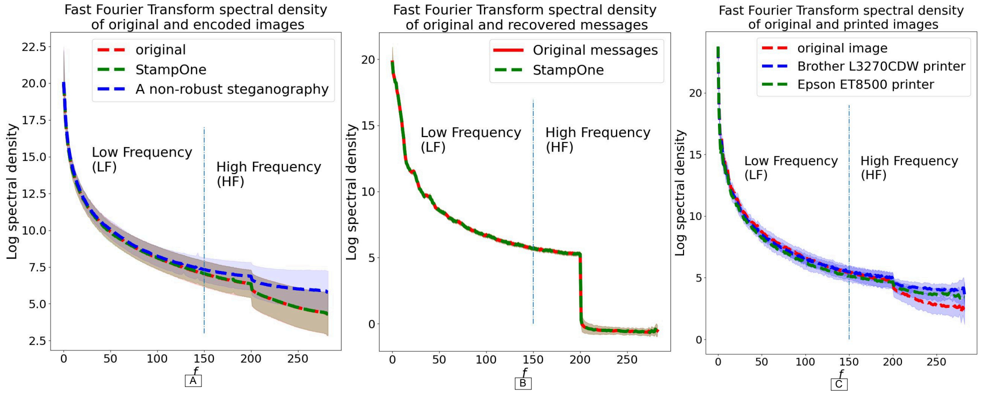

Figure 1. (A) compares the spectral density of original and encoded images using StampOne and a non-robust steganography algorithm. StampOne shows a higher correlation between high-frequency components in the encoded images and the original ones, unlike non-robust steganography GANs (such as [4]), which exhibit a lower correlation. (B) illustrates the spectral density of original and recovered messages using the StampOne decoder. The messages (2D binary) show a sharper decline in high-frequency components, which implies a harder task for the decoder in handling these frequencies. (C) presents the spectral density of original images and printed original photographs obtained from two different office printers (Brother $L3270CDW$ and Epson $ET8500$). Subsequently, the printed images were scanned for analysis. Printers introduce noise and elevate the spectral density of high-frequency components. Additional analysis on image gradients is available in the supplementary materials for further examination of the observed differences.

work architecture is more constrained than in other machine learning applications, due to the severe noise conditions that might be encountered [37].

This paper focuses on high-level robust steganography, such as it was proposed by StegaStamp [37] and Code Face [36], striking a balance between high-quality encoded images and decoding accuracy to overcome the aforementioned limitations. We have explored a hypothesis for concealing messages in images by transferring the message and the image (input of the networks) to a specific balanced frequency space. This idea is inspired by previous works that focused on frequency ranges to enhance GANs model performance [18, 26, 35, 45].

The analysis of the frequency domain brought a new perspective to the problem. On one hand, in Figure 1, specifically in plots (A) and (B), we demonstrate the frequency bias issue that can be found in steganography models and the impact of digital transformations on high-frequency components of the encoded images. On the other hand, the noises arising from several sources, such as social media (JPEG compression), camera sensors (lighting and blurring), and printers (dithering and Gaussian noise), just to mention some of the known noise sources, have often a direct impact on the high-frequency components of the encoded images. We illustrate these behaviors in Figure 1 (C), using printed images as examples.

StampOne was designed to address the frequency-related issues associated with image degradation in strong noisy conditions. Our method utilizes preprocessing models for the encoder and decoder, incorporating gradient transform, wavelet transform, and ”Depthwise” layer [38] to normalize and balance frequencies of the input data (original images, message, and encoded images). Furthermore, we have devised a dedicated network for message preparation, aiming to embed messages into original images and adjust the input size of the encoder network. For the sake of clarity along the document, we denote this network as the Message Preparation Network (MPN), as depicted in Figure 2.

StampOne effectively learns and generates the high and the low frequencies components (balance frequency density) in both the encoded images and the recovered messages, as demonstrated in Figures 1 (A) and (B). The quality of the encoded images is notably enhanced across multiple models. Remarkably, the decoder exhibits improved performance, even when decoding small encoded images. We validate our method by testing it with several variants of U-shape networks. Notably, our model with AttentionVNet [30] surpasses the previous state-of-the-art (StegaStamp) in 10% considering the decoding performance.

In summary, the main contributions of this work are:

1. We introduce a novel approach that utilizes gradient and wavelet processing to convert input data into a specific frequency range for both the encoder and decoder, ensuring consistent frequency normalization independent of the U-shape architecture.
2. We tackle the frequency density balance problem in GANs, achieving superior image decoding and better overall image quality.

In Section 2, we conduct a comprehensive review of current image steganography models, examining their limitations and drawbacks. This analysis forms the basis for our proposed approach. In Section 3, we present the impor-

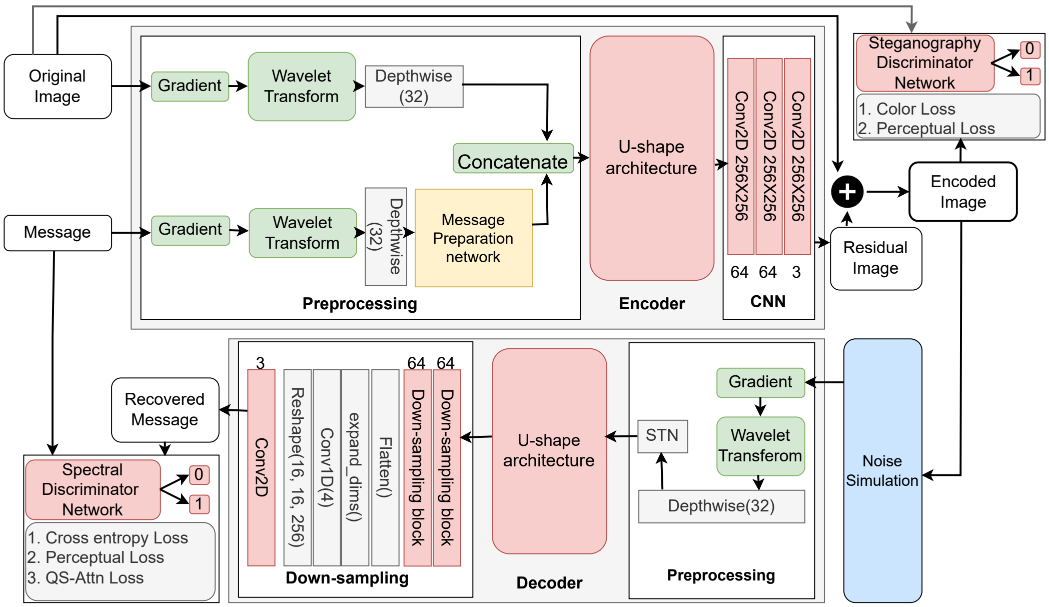

Figure 2. The complete end-to-end training networks of StampOne consist of several components, including StampOne preprocessor stages, a U-shaped encoder, a steganography discriminator, an encoder loss function, a U-shaped decoder, a spectral discriminator, and a decoder loss function. Additionally, the architecture incorporates the Spatial Transformer Network (STN) [23], that improves the decoder's ability to read the messages from warped or rotated encoded images.

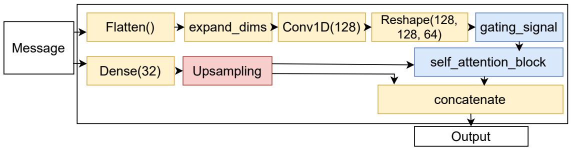

Figure 3. The depicted network is responsible for preparing messages to be embedded within images. The Message Preparation Network (MPN) implementations shown in the figure consistently yield reliable outcomes for both digital and printed images.

tance of exploring the frequency domain in deep learning networks and outline the specific architectures of our encoder and decoder. In Section 4, we extensively evaluate our results, comparing them with existing robust steganography models that have been previously published.

2. Related Work

Within the realm of steganography, the insertion of secrets can take place in either the spatial [31] or frequency domain [5, 19, 20, 39] of the image, employing hand-crafted [19] or learning-based techniques [7, 36, 37]. We review the two different steganography types (Robust and Non-robust) related to our work:

Non-robust steganography, Barni et al. [5] proposed an approach that leverages the Discrete Wavelet Transform (DWT) to enhance encoded images loss function in specific regions of an image. To create the encoded image, the DWT weights of the message are adjusted and added to the DWT of the original image ($DWT_{encoded} = DWT_{image} + w \times DWT_{message}$).

Wavelet Obtained Weights (WOW) [19] and Universal Wavelet Relative Distortion (UNIWARD) [20] present an embedding algorithm that utilizes the Syndrome-trellis code [16] and Wavelet coefficients to hide messages in textured or noisy areas of the images. The embedding algorithm evaluates the costs associated with altering pixels between encoded and original images.

Another model based on Discrete Fourier Transform (DFT) is proposed by Matthieu Urvoy et al. [39]. In this model, a message is embedded in the Fourier domain of the images. The coefficients of encoding are organized into two symmetrical square patches, which can be represented as a sum of sine waves (sinusoidal grating).

Robust steganography, StegaStamp [37] is the first printer-proof steganography model capable of decoding messages from physically printed images. The authors introduced a novel noise simulation pipeline to replicate printing and digitization distortions based on HiDDeN [48]. To minimize the noticeable quality gap between the encoded and cover images during training, the method employs the LPIPS perceptual loss [46]. The authors eliminated the bottleneck block within the encoder architecture of UNet, which compels the network to conceal messages within the high-dimensional frequency domain of images. This approach significantly enhances the visual appearance of the encoded samples. However, the model faces limitations due to the structural constraints of its encoder. Consequently, developing a robust steganography model based on alternative U-shape structures becomes challenging, as the hidden message's integrity may be compromised in the presence of noise.

Code Face [36] is a comprehensive Machine Readable Coding (MRC) method for encoding and decoding secret messages in small face photos for ID documents. It improves the encoded image quality by minimizing the facial feature distance between encoded and original face photographs.

RoSteALS [7] leverages the diffusion model in conjunction with VQGAN [15] to translate both the cover image and message into latent code space. Subsequently, these latent representations are fused and processed through a generator to create encoded images, which is developed based on the diffusion model. While the model does not exhibit robustness in printer tests, its performance remains competitive with the state-of-the-art concerning robustness against digital noises like JPEG compression and changes in resolution.

3. Methodology

3.1. Importance of frequency balancing techniques

In the existing models [5, 14, 20, 31, 37, 43], it is commonly observed that hiding messages within the texture or background of an image and avoiding the edges, reduces the visual perception of artifacts for the majority of encoded images. Suppressing information at higher frequencies improves the invisibility of artifacts in the encoded images but reduces the robustness in the decoding process, especially considering printed images. Conversely, concealing messages in lower frequencies leads to more noticeable artifacts [19, 20], albeit improving the resilience of the hidden messages. Thus, a balance should be found respecting the frequency range in which the hidden information should be encoded. It is crucial to address the frequency bias issue in GANs; the spectral density of the high-frequency data in GAN-generated images significantly deviates from that of the original images [28, 32]. This discrepancy can have adverse effects on down-sampling and up-sampling blocks in the network, making the output prediction sensitive to minor changes in input images [18, 26, 35, 45]. This problem is exacerbated in models exhibiting outputs resembling chessboard patterns or binary code [11].

In Figure 1, we compute the Fast Fourier Transform (FFT) spectral density of the images and divide them into low and high frequencies, following a methodology akin to that of [14]. Figure 1 (A) demonstrates this frequency mismatch between the inputs and generated outputs of both robust and non-robust steganography models (such as [4]). The non-robust steganography model is designed from an Attention−VNet (pix2pix) [34], here mentioned for reference purposes. In Figure 1 (B), one can observe a significant decline in the amplitude of high frequencies in 2D binary messages, a phenomenon that poses challenges for decoding steganography networks with low-amplitude frequencies. To further investigate the impact of printing and digital types of noise, we computed the spectral density of printed and original images, as depicted in Figure 1 (C). These types of noise introduce additional patterns in high-frequency components, thereby increasing the spectral density.

To overcome these challenges, we propose a hypothesis that involves preprocessing the input data to achieve a balanced frequency density. This is accomplished by addressing three key factors in our encoder and decoder networks: Firstly, balancing input frequency in robust steganography GANs using Sobel operation. Secondly, concealing messages beyond lowest and highest frequency regions of cover images with DWT. Lastly, amplifying high frequency for the decoder more than the encoder using a spectral discriminator [18]. The binary 2D messages outputted by our decoder contribute to this imbalance, prompting us to employ a discriminator to amplify high frequencies.

3.1.1 Highlighting high frequency component

By passing the networks' inputs through a gradient operation [14, 31], we highlighted high frequency for the encoder and the decoder (we have only the edges of the images). In this way, the neural network has the capability to prioritize learning the highest frequency content over the lowest frequency content, to reduce undesirable aliasing effects, while preserving important content as much as possible. To reinforce the high frequencies, we can also use high-frequency pass filters [44] to remove the lowest frequency, but our experiments show that the Sobel operation is a more stable operation.

3.1.2 Discrete wavelet transform

We employ Haar wavelet transforms [5] to partition the input gradients into five distinct sub-bands. The top sub-band, denoted as I_G, corresponds to the gradients of the original images. The subsequent sub-band, LL, captures low-frequency details, while the remaining sub-bands (LH, HL, and HH) primarily represent vertical, horizontal, and diagonal edges, respectively, thereby capturing high-frequency information. By applying the wavelet transform to RGB images, the number of channels in the input data is increased from 3 to 15, considering that each of the five wavelet sub-bands contains three channels (RGB). This frequency representation of the gradient inputs allows us to emphasize the high-frequency features and facilitates an improved generalization of the model.

3.2. Encoder

The general architecture of the encoder is shown on the top of Figure 2. It consists of three elements, a preprocessing block, an U-shape network, and three layers of Convolution in the last block (CNN block).

Preprocessing block: to address the first two aforementioned factors explained in the previous section. We preprocess the inputs of the encoder and the decoder networks by reshaping the 256-bit binary sequences into a (16×16) 2D matrix in grayscale image format. The 2D message is then converted to 3D in RGB image format, and both the message and the cover image are subjected to gradient and wavelet operations. The wavelet transform is applied to obtain the dimensions of $16 \times 16 \times 15$ for the message and $256 \times 256 \times 15$ for the original image. The "Depthwise" layer is employed to assign distinct weights to each of the DWT sub-bands, following the wavelet transform. The highlighted message in the wavelet domain is then sent to the message preparation network (MPN).

MPN, which can be comprised of a set of two, three, or four layers, depending on the message size, is responsible for translating the message into a format compatible with the subsequent processing steps. We designed four MPN as detailed in Supplementary Material. The model demonstrating the most promising results is depicted in Figure 3. It shows robustness against various types of noise while preserving acceptable perceptual quality in its encoded images.

The high-level features from both RGB and frequency signals are added by concatenating the images and messages in the gradient and wavelet domains. This fused representation serves as the input for the next block, the U-shape network.

U$-$shape network: We leverage the advantage of our preprocessing method by integrating it into various image-to-image networks, including UNet [33], VNet [34], Eff$-$UNet [3], LeViT$-$UNet [42] ResUNet [41], Swin$-$UNet [8], Attention$-$UNet [30], Attention$-$VNet and UNet++ [47]. While these architectures were originally designed to operate in the spatial domain, we propose using them in the decomposed high-frequency domain, represented by the gradient and wavelet coefficients. Our approach demonstrates superior performance in generating encoded images with higher visual quality and more accurate high-frequency content, surpassing StegaStamp and Code Face. We refer UNet as a network that employs max-pooling in their down-sampling blocks [33], and VNet as a network that uses only Convolution layers, such as the pix2pix model [34], in its down-sampling blocks. The details of these networks are beyond the scope of this research.

CNN block, comprising three 2D convolution layers, is incorporated to enhance the network's ability to generate realistic encoded images. The activation functions utilized in the first two layers are "Leaky ReLU", while the activation function in the final layer is "Snake".

3.3. Decoder

Preprocessing block: the decoder follows a similar structure as the encoder. The gradient and wavelet transformers of the encoded images are passed through the "Depthwise" layer and Spatial Transformer Network (STN) [23]. In the "Depthwise" layer, each channel of the image frequency wavelet is assigned with a weight to emphasize the high-frequency components of the encoded image. The STN is used to prevent warping and rotation when printing and capturing encoded images by a camera sensor.

U$-$shape network: the U-shape network is applied, which involves an identical mirror design for both the encoder and the decoder.

CNN block: following the U-shape architecture, a CNN down-sampling operation is performed in the final block. The number of downsampling layers is determined by the size of the message. The output size of the U-shape network is $256 \times 256 \times 3$ resolution, which is subsequently resized by the downsampling blocks to match the size of the message ($16 \times 16 \times 3$ for a 256-bit message). To enhance the precision of the decoder, a convolutional 1D layer is employed at the end of the decoder network. The final convolutional 2D layer (filiters=3), using the Snake function [49], recovers the hidden message from the decoder's output. In our computational framework, we employ a distinct structure at the termination of the decoder network, deviating from conventional approaches such as a simplistic CNN akin to StegaStamp. This decision is driven by two key rationales. Primarily, avoiding a linear layer mitigates the network's exponential growth in proportion to the size of the messages, thereby optimizing computational resources. Secondly, empirical analyses affirm the superior efficacy of this structural alteration, demonstrating improved performance outcomes.

3.4. Steganography and spectral discriminators

In the case of steganography GANs, the discriminator should be trained to focus on minimizing the difference between the encoded and original images. With this goal in mind, we calculate Wasserstein adversarial loss (L_{SD}) [2] between the discriminator feature output of the encoded and the original images. This discriminator is introduced by the StegaStamp model.

The standard GAN frameworks do not perform well in capturing and reconstructing high-frequency information from specific image datasets, such as chessboard (binary) patterns. These patterns contain crucial edges and details, requiring the generation of accurate high-frequency components [11, 14, 18, 28, 35]. In tackling this challenge, we employ a spectral discriminator (F_{FD}) inspired by the SWAGAN model [18]. It is trained simultaneously with the encoder, decoder, and steganography discriminator to leverage both frequency space decomposition and pixel values of the recovered and actual messages. The SWAGAN model employed Wavelet Transform of the input. However, we have chosen to use Fast Fourier Transform (FFT) of the input instead of Wavelet Transform.

3.5. Loss functions

Encoder Loss Function: The encoder loss function (L_{en}) incorporates color histogram (l_{Color}) [1], perceptual loss (LPIPS) (l_P) [46], and steganography discriminator (l_{SD}). It is defined as follows:

$$L_{en} = \lambda_{Color} \times l_{Color} + \lambda_P \times l_P + \lambda_{SD} \times l_{SD} \quad (1)$$

where, the weights of each component, denoted by λ_{Colour}, λ_P, and λ_{SD}, are adjusted to balance their respective contributions.

Color Histogram is based on Log-Chroma space and computes the Euclidean norm of the histogram features (H) between encoded and original images. LPIPS uses a pre-trained pyramid network to extract image features from different layers, and the average of these features is used to measure perceptual differences.

Decoder Loss Function: The decoder loss function (L_{de}) includes cross-entropy (l_{SC}) and QS-Attn Contrastive loss (l_{QSDe}) [21] components, along with the spectral discriminator (l_{FD}). It is defined as follows:

$$L_{de} = \lambda_{SC} \times l_{SC} + \lambda_{QSDe} \times l_{QSDe} + \lambda_{FD} \times l_{FD} \quad (2)$$

where the weights of the loss function components, denoted by λ_{SC}, λ_{QSDe}, and λ_{FD}, are adjusted accordingly. The QS-Attn Contrastive loss function employs a self-attention network to select anchor features and encourages dissimilar anchors to spread apart, while grouping similar anchors.

3.6. Noise simulation

To enhance decoder robustness in real-world scenarios, we conducted perturbation or noise self-attack simulations during the network training. Various operations were applied to the images in order to simulate these types of noise, including those arising from digital sources, printers, and camera sensors [13]. Various operations were applied to the images in order to simulate these types of noise, including gray transfer, JPEG noise compression, Gaussian noise, affine noise transformation, sharpen transformation, linear pixel interpolation, color dithering, random brightness, random contrast, random hue shift, medium blur, and perspective warp.

4. Experiments

4.1. Datasets, training details and metrics

Datasets: To perform our training experiments, We utilized sub-sets of two main datasets for training: COCO [25] and DeepFashion [27] datasets, which consisted of approximately $123k$ and $800k$ images, respectively. To evaluate the performance of our encoder and decoder model under different noise, we randomly selected 1000 images from the COCO test dataset. For printer-proof testing, we selected two different datasets of various image datasets. The first dataset comprised randomly chosen 30 images from the BSDS500 dataset [29] and 10 images from the Urban dataset [22]. The results of this dataset are provided in the Supplementary Material. For the second test set, we specifically selected 40 face images from the VGGFace2 dataset [9], tailored for face image testing.

Training details: During the training process, we used a batch size of 10. The coefficients for the encoder loss function in Equation 1 were set as follows: λ_{Color} = 1, λ_P = 2, and λ_{SD} = 1. For the decoder loss function coefficients in Equation 2, we set λ_{SC} = 1, λ_{QSDe} = 1, and λ_{FD} = 1.

The models were trained on a single NVIDIA GeForce RTX 3090 GPU, utilizing the Adam optimizer with an initial learning rate of 0.0001. The learning rate was decreased by 10% every 20000 steps.

Metrics: In assessing the quality of the encoded output against the original cover image, we utilize a comprehensive set of metrics including Structural Similarity Index Measure (SSIM) [40], deep learning perceptual similarity score as measured by LPIPS [46] and Color Histogram (ColorHisto)[1]. For the secret decoding, we report standard bit accuracy (Bit acc), with cyclic error correction code using Bose–Chaudhuri–Hocquenghem (BCH) codes [17].

4.2. Baseline comparison

Code Face and **StegaStamp** achieved a message retrieval capacity of 0.13×10^{-3} bpp (100 bits in 400×400 pixels). This success relied on error-correcting codes.

RoSteALS model conceals a secret of 100 bits of length within an image of resolution $256 \times 256 \times 3$, where the model's capacity is 0.5×10^{-3}.

StampOne outperforms Code Face, StegaStamp, and RoSteALS with a capacity of 0.13×10^{-2} bpp (256 bits) into image of resolution $256 \times 256 \times 3$, which is approximately 10 times higher than Code Face and StegaStamp. We consider the two most robust models of StampOne, namely the models utilizing Attention−VNet (M1) and UNetPlus (M2) architectures.

Non-robust model is constructed using two Attention−Vnet architectures (utilized in pix2pix [34]) for both the encoder and decoder components. Additionally, it incorporates a steganography discriminator and loss func-

Table 1. (A) Encoded image quality Metrics. (B) Decoders' performances on 40 printed encoded images captured with Samsung S22 Ultra smartphone. M1 and M2 refer to StampOne models employing Attention−VNet and UNetPlus architectures, respectively. M3 denotes a non-robust model constructed using two instances of Attention−VNet. The initial four rows consist of high-level robust models, while the final two rows encompass non-robust models, serving as reference points when decoding messages from printed encoded images.

	(A) Encoded images quality			(B) **Bit acc (%)** - VGGFace2 [9]				
Methods	SSIM (⇑)	LPIPS (⇓)	ColorHisto (⇓)	6×6 cm	5×5 cm	4×4 cm	3×3 cm	2×2cm
StegaStamp [37]	0.93 ± 0.001	4.92 ± 1.6	6.11 ± 10.5	78	72	70	65	48
Code Face [36]	0.95 ± 0.0002	3.06 ± 0.9	7.32 ± 6.1	55	55	50	38	15
StampOne (M1)	**0.98** ± **0.00002**	**1.25** ± **0.4**	**5.38** ± **4.9**	**100**	**100**	**100**	**95**	**62**
StampOne (M2)	0.96 ± 0.00007	2.74 ± 2.38	6.30 ± 4.07	88	85	72	63	43
Non-robust (M3)	0.92 ± 0.001623	1.04± 1.69	2.80 ± 60.8	0	0	0	0	0
RoSteALS [7]	0.95 ± 0.0006	**0.04** ± **0.0003**	**0.09** ± **0.003**	0	0	0	0	0

Table 2. Impact of three types of image under different noise types. 1000 images from COCO test dataset are used for the decoder performance evaluation. Bit accuracy (%) during decoding from encoded images is evaluated under various types and levels of noise. M1 and M2 represent StampOne models utilizing the Attention−VNet and UNetPlus architectures, respectively. On the other hand, M3 refers to a non-robust model constructed through the utilization of two instances of Attention−VNet.

	JPEG (%)			Gaussian (Std 0 to 1)			Resolution (Pixel)		
Methods	70	60	50	0.08	0.06	0.04	(60 × 60)	(80 × 80)	(100 × 100)
StegaStamp [37]	**100**	**100**	**100**	**100**	**100**	**100**	55	80	91
Code Face [36]	80	88	88	55	75	86	2	11	36
RoSteALS [7]	87	90	94	23	35	53	**96**	97	98
StampOne (M1)	**100**	**100**	**100**	98	**100**	**100**	74	**98**	**100**
StampOne (M2)	97	99	**100**	88	96	99	72	94	99
Non-robust (M3)	0	0	0	13	46	84	0	0	22

Table 3. The bit accuracy (%) during decoding from encoded images is assessed across varying degrees of contrast and brightness. M1, M2, and M3 refer to the models listed in Table 2.

	Contrast (0 to 1)			Brightness (-1 to 1)	
Methods	0.05	0.1	0.15	-1	1
StegaStamp [37]	2.0	39	77	**100**	**100**
Code Face [36]	0	1	30	90	90
RoSteALS [7]	20	67	85	91	95
StampOne (M1)	**100**	**100**	**100**	**100**	**100**
StampOne (M2)	**100**	**100**	**100**	**100**	**100**
Non-robust (M3)	0	0	15	0	0

tions similar to those employed in StampOne. However, during its training phase, we did not incorporate MPN preprocessing or a spectral discriminator. This model is mentioned for reference.

4.3. Impact on perceptual quality

Table 1 (A) illustrates the quality of encoded images. Among Code Face and StegaStamp evaluation, StampOne exhibits the best performance, with AttentionVnet and UNetPlus referred in the table. Additional details regarding StampOne with other networks can be found in the supplementary material. Regarding the SSIM of the encoded images, StampOne consistently outperforms other existing robust models. While StampOne ranks second for ColorHisto and LPIPS metrics, RoSteALS surpasses it in these metrics. However, RoSteALS is unable to recover any message from printed encoded images.

4.4. Robustness

Forty selected images from VGGFace2 underwent encoding and subsequent printing at diverse dimensions, spanning from 2 cm by 2 cm to 6 cm by 6 cm (width×height), employing a consumer Brother $L3270CDW$ printer. To simulate real-world conditions, we conducted the decoding tests in an uncontrolled lighting environment and recorded videos using a Samsung S22 smartphone. The performance of our decoders, including AttentionVNet and UNetPlus, was compared against other models such as StegaStamp and Code Face. Table 1 (B) illustrates that our model with AttentionVNet achieved superior performance compared to the other networks, demonstrating its effectiveness in printer-proof steganography applications. We repeat the test with other smartphones in the supplementary material.

To evaluate the performance of the decoder, we conducted experiments under various noise distortions, including JPEG compression, Gaussian noise, different resolutions, and contrast and brightness variations. The decoder's

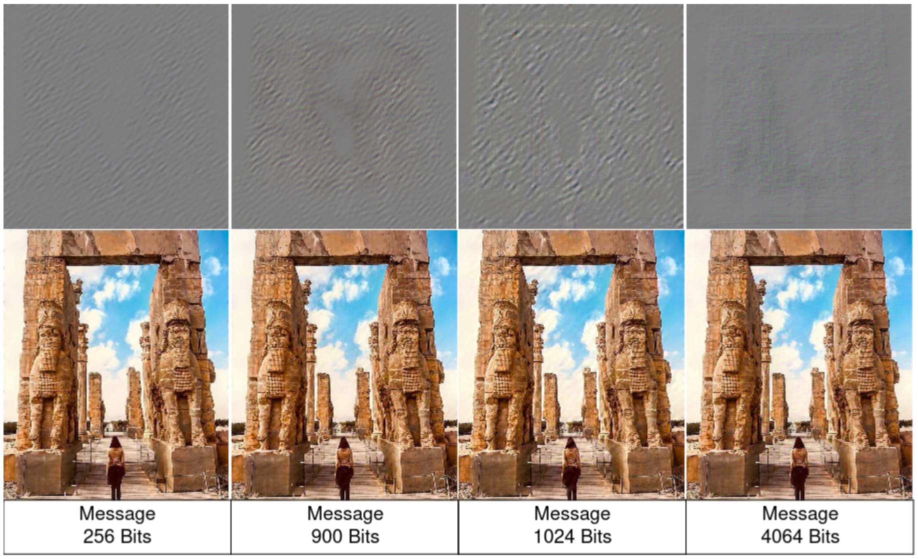

Figure 4. The results of encoding a message with capacities of 256 bits, 900 bits, 1024 bits, or 4096 bits are presented. The first row depicts the residual image added to the original images to generate the encoded images.

effectiveness was measured by calculating the percentage of accurately decoded messages from the encoded images. The results are detailed in Tables 2 and 3. Our preprocessing techniques, implemented with AttentionVNet and UNetPlus, consistently exhibited the best performance across these experiments. Notably, StegaStamp demonstrates superior robustness, equaling our StampOne model, specifically under JPEG compression, Gaussian, and brightness (both darkest and lightest). See supplementary material for results with other Ushape in StampOne.

The concept of high-level robust steganography, ie. being able to decode messages from encoded images with digital and printed noise, can be understood through the Table 1 (B), where the messages can be successfully decoded from printed encoded images.

4.5. Effects of the message size on the quality of encoded images

Our model is capable of concealing up to 4096 bits of information within a cover image of size (256×256 pixel), which is 20 times more than the current robust steganography models, StegaStamp, Code Face and RoSteALS [7, 36, 37]. We presented encoded image samples with different capacity sizes in Figure 4. When increasing the capacity, the Structural Similarity Index (SSIM) between original and encoded images decreases in encoded images as shown in Table 4. However, LPIPS and ColorHisto are greater for 900 bits model than for 1024 bits model.

Table 4. The encoded image quality of StampOne AttentionVnet is assessed across different message sizes encoded within them.

Message Size	SSIM (⇑)	LPIPS (⇓)	ColorHisto (⇓)
256 (bits)	**0.98 ± 0.00002**	**1.25 ± 0.4**	**5.38 ± 4.9**
900 (bits)	0.94 ± 0.0003	5.2 ± 11.1	10.0 ± 7.0
1024 (bits)	0.92 ± 0.0004	4.14 ± 8.5	8.7 ± 5.70
4096 (bits)	0.91 ± 0.001	11.1 ± 18.2	14.2 ± 12.8

5. Conclusion

Despite the recent advancements in robust steganography models, they still exhibit limitations such as the maximum size of the encoded message, the trade off between the decoding accuracy and the perceptual quality of encoded images, and restricted flexibility in neural network architecture selection. To overcome these limitations, we introduce StampOne, a novel approach that performs a preprocessing on the frequency domain that can be used in diverse network architectures to improve printer-proof steganography models compared to existing methods, achieving encoder and decoder performance enhancements of up to 10% against previous state-of-the-art models, StegaStamp and Code Face. The proposed model can conceal messages up to 4096 bits, although our current optimization focuses on robust preprocessing network hyperparameters for messages up to 256 bits. Future work will extend this optimization for larger message capacities and explore various robust steganography models with balanced network input frequencies. Additionally, we aim to investigate alternative methods like diffusion model and normalizing flows, to enhance robustness and versatility.

References

[1] Mahmoud Afifi, Marcus A Brubaker, and Michael S Brown. Histogan: Controlling colors of gan-generated and real images via color histograms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7941–7950, 2021. 6

[2] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017. 5

[3] Bhakti Baheti, Shubham Innani, Suhas Gajre, and Sanjay Talbar. Eff-unet: A novel architecture for semantic segmentation in unstructured environment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 358–359, 2020. 5

[4] Shumeet Baluja. Hiding images in plain sight: Deep steganography. *Advances in neural information processing systems*, 30, 2017. 2, 4

[5] Mauro Barni, Franco Bartolini, and Alessandro Piva. Improved wavelet-based watermarking through pixel-wise masking. *IEEE transactions on image processing*, 10(5): 783–791, 2001. 3, 4

[6] Jon Bateman. *Deepfakes and synthetic media in the financial system: Assessing threat scenarios*. Carnegie Endowment for International Peace., 2022. 1

[7] Tu Bui, Shruti Agarwal, Ning Yu, and John Collomosse. Rosteals: Robust steganography using autoencoder latent space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 933–942, 2023. 3, 4, 7, 8

[8] Hu Cao, Yueyue Wang, Joy Chen, Dongsheng Jiang, Xiaopeng Zhang, Qi Tian, and Manning Wang. Swin-unet: Unet-like pure transformer for medical image segmentation. In *European conference on computer vision*, pages 205–218. Springer, 2022. 5

[9] Qiong Cao, Li Shen, Weidi Xie, Omkar M. Parkhi, and Andrew Zisserman. VGGFace2: A dataset for recognising faces across pose and age. In *International Conference on Automatic Face and Gesture Recognition*, 2018. 6, 7

[10] Jianbo Chen, Xinwei Liu, Siyuan Liang, Xiaojun Jia, and Yuan Xun. Universal watermark vaccine: Universal adversarial perturbations for watermark protection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2321–2328, 2023. 1

[11] Yuanqi Chen, Ge Li, Cece Jin, Shan Liu, and Thomas Li. Ssd-gan: Measuring the realness in the spatial and spectral domains. In *Proceedings of the AAAI Conference on Artificial Intelligence*, number 2, pages 1105–1112, 2021. 4, 6

[12] Bobby Chesney and Danielle Citron. Deep fakes: A looming challenge for privacy, democracy, and national security. *Calif. L. Rev.*, 107:1753, 2019. 1

[13] T. Cunha, L. Schirmer, J. Marcos, and N. Gonçalves. Noise simulation for the improvement of training deep neural network for printer-proof steganography. In *Proceedings of the 13th International Conference on Pattern Recognition Applications and Methods*, pages 179–186, 2024. 6

[14] Ricard Durall, Margret Keuper, and Janis Keuper. Watch your up-convolution: Cnn based generative deep neural networks are failing to reproduce spectral distributions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7890–7899, 2020. 4, 6

[15] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021. 4

[16] Tomáš Filler, Jan Judas, and Jessica Fridrich. Minimizing additive distortion in steganography using syndrome-trellis codes. *IEEE Transactions on Information Forensics and Security*, 6(3):920–935, 2011. 3

[17] George Forney. On decoding bch codes. *IEEE Transactions on information theory*, 11(4):549–557, 1965. 6

[18] Rinon Gal, Dana Cohen Hochberg, Amit Bermano, and Daniel Cohen-Or. Swagan: A style-based wavelet-driven generative model. *ACM Transactions on Graphics (TOG)*, 40(4):1–11, 2021. 2, 4, 6

[19] Vojtěch Holub and Jessica Fridrich. Designing steganographic distortion using directional filters. In *2012 IEEE International workshop on information forensics and security (WIFS)*, pages 234–239. IEEE, 2012. 3, 4

[20] Vojtěch Holub, Jessica Fridrich, and Tomáš Denemark. Universal distortion function for steganography in an arbitrary domain. *EURASIP Journal on Information Security*, 2014 (1):1–13, 2014. 3, 4

[21] Xueqi Hu, Xinyue Zhou, Qiusheng Huang, Zhengyi Shi, Li Sun, and Qingli Li. Qs-attn: Query-selected attention for contrastive learning in i2i translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18291–18300, 2022. 6

[22] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5197–5206, 2015. 6

[23] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. *Advances in neural information processing systems*, 28, 2015. 3, 5

[24] Pavel Korshunov and Sébastien Marcel. Deepfakes: a new threat to face recognition? assessment and detection. *arXiv preprint arXiv:1812.08685*, 2018. 1

[25] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 6

[26] Pengju Liu, Hongzhi Zhang, Kai Zhang, Liang Lin, and Wangmeng Zuo. Multi-level wavelet-cnn for image restoration. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 773–782, 2018. 2, 4

[27] Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1096–1104, 2016. 6

[28] Salma Abdel Magid, Yulun Zhang, Donglai Wei, Won-Dong Jang, Zudi Lin, Yun Fu, and Hanspeter Pfister. Dynamic high-pass filtering and multi-spectral attention for image super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4288–4297, 2021. 4, 6

[29] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. 8th Int'l Conf. Computer Vision*, pages 416–423, 2001. 6

[30] Ozan Oktay, Jo Schlemper, Loic Le Folgoc, Matthew Lee, Mattias Heinrich, Kazunari Misawa, Kensaku Mori, Steven McDonagh, Nils Y Hammerla, Bernhard Kainz, et al. Attention u-net: Learning where to look for the pancreas. *arXiv preprint arXiv:1804.03999*, 2018. 2, 5

[31] Tomáš Pevnỳ, Tomáš Filler, and Patrick Bas. Using high-dimensional image models to perform highly undetectable steganography. In *International workshop on information hiding*, pages 161–177. Springer, 2010. 3, 4

[32] Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. In *International Conference on Machine Learning*, pages 5301–5310. PMLR, 2019. 4

[33] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 5

[34] Pegah Salehi and Abdolah Chalechale. Pix2pix-based stain-to-stain translation: A solution for robust stain normalization in histopathology images analysis. In *2020 International Conference on Machine Vision and Image Processing (MVIP)*, pages 1–7. IEEE, 2020. 4, 5, 6

[35] Katja Schwarz, Yiyi Liao, and Andreas Geiger. On the frequency bias of generative models. *Advances in Neural Information Processing Systems*, 34:18126–18136, 2021. 2, 4, 6

[36] Farhad Shadmand, Iurii Medvedev, and Nuno Gonçalves. Code face: A deep learning printer-proof steganography for face portraits. *IEEE Access*, 9:167282–167291, 2021. 1, 2, 3, 4, 7, 8

[37] Matthew Tancik, Ben Mildenhall, and Ren Ng. Stegastamp: Invisible hyperlinks in physical photographs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2117–2126, 2020. 1, 2, 3, 4, 7, 8

[38] Yi Tay, Mostafa Dehghani, Dara Bahri, and Donald Metzler. Efficient transformers: A survey. *ACM Computing Surveys*, 55(6):1–28, 2022. 2

[39] Matthieu Urvoy, Dalila Goudia, and Florent Autrusseau. Perceptual dft watermarking with improved detection and robustness to geometrical distortions. *IEEE Transactions on Information Forensics and Security*, 9(7):1108–1119, 2014. 3

[40] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 6

[41] Xiao Xiao, Shen Lian, Zhiming Luo, and Shaozi Li. Weighted res-unet for high-quality retina vessel segmentation. In *2018 9th international conference on information technology in medicine and education (ITME)*, pages 327–331. IEEE, 2018. 5

[42] Guoping Xu, Xuan Zhang, Yin Fang, Xinyu Cao, Wentao Liao, Xinwei He, and Xinglong Wu. Levit-unet: Make faster encoders with transformer for biomedical image segmentation. *Available at SSRN 4116174*, 2022. 5

[43] Kai Xu, Minghai Qin, Fei Sun, Yuhao Wang, Yen-Kuang Chen, and Fengbo Ren. Learning in the frequency domain. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1740–1749, 2020. 4

[44] Shin'ya Yamaguchi and Sekitoshi Kanai. F-drop&match: Gans with a dead zone in the high-frequency domain. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6743–6751, 2021. 4

[45] Richard Zhang. Making convolutional networks shift-invariant again. In *International conference on machine learning*, pages 7324–7334. PMLR, 2019. 2, 4

[46] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. pages 586–595, 2018. 3, 6

[47] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. Unet++: A nested u-net architecture for medical image segmentation. In *Deep learning in medical image analysis and multimodal learning for clinical decision support*, pages 3–11. Springer, 2018. 5

[48] Jiren Zhu, Russell Kaplan, Justin Johnson, and Li Fei-Fei. Hidden: Hiding data with deep networks. pages 657–672, 2018. 3

[49] Liu Ziyin, Tilman Hartwig, and Masahito Ueda. Neural networks fail to learn periodic functions and how to fix it. *Advances in Neural Information Processing Systems*, 33:1583–1594, 2020. 5