IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 14 March 2026, accepted 30 March 2026, date of publication 2 April 2026, date of current version 10 April 2026.

Digital Object Identifier 10.1109/ACCESS.2026.3680290

APPLIED RESEARCH

DocSafe: Toward Practical Print-Proof Image Steganography via Frequency Decomposition and Covariance Alignment

FARHAD SHADMAND[1], IURII MEDVEDEV[1], LUIZ SCHIRMER[2], AND NUNO GONÇALVES[1], (Member, IEEE)

[1]Institute of Systems and Robotics, University of Coimbra, 3030-789 Coimbra, Portugal
[2]Colégio Técnico Industrial, Universidade Federal de Santa Maria, Santa Maria 97105-900, Brazil

Corresponding author: Farhad Shadmand (farhad.shadmand@isr.uc.pt)

ABSTRACT Data hiding techniques such as robust steganography and invisible watermarking are important for applications including aesthetic stamps, copyright protection, privacy-preserving communication, and content provenance. However, existing print-proof steganography methods often struggle to embed messages in small image regions or decode them from very low-resolution printed outputs. In this paper, we introduce DocSafe, a framework that embeds binary messages (up to 256 bits) into images while producing stamp-like outputs robust to digital and physical distortions. The method enables region-aware embedding using spatial masks, analyzes the impact of wavelet sub-band frequencies for improved robustness, and investigates covariance-based losses, identifying Stein divergence for the encoder and AIRM loss for the decoder as the most effective configuration. A lightweight architecture is further proposed to enhance both imperceptibility and decoding accuracy. We achieve 100% bit accuracy from printed images as small as 2×2 cm in the face dataset and at 3×3 cm on the object dataset, demonstrating state-of-the-art performance in print-proof robust steganography, even when messages are embedded in specific image regions using spatial masks. DocSafe also improves visual quality (ColorHisto 0.05, FID 4.11) and maintains strong decoding accuracy under various digital noise conditions.

INDEX TERMS Robust image steganography, digital steganography, watermark, print-proof, deep learning, GANs.

I. INTRODUCTION

With the rapid advancement of digital tools and techniques for generating synthetic content across both digital and physical domains, critical challenges have emerged in distinguishing original images from manipulated ones, watermarking and steganography AI-generated media, protecting brands from counterfeiting, and ensuring the integrity of identity documents and official records.

The associate editor coordinating the review of this manuscript and approving it for publication was Mehul S. Raval.

In response to these emerging threats, advanced deep learning models for image watermarking and steganography have been developed [1], [2], [3], [4], [5], aiming to enhance the security and authenticity of digital content while mitigating risks associated with forgery and unauthorized alterations. These models are designed to embed security identifiers into images with minimal perceptual differences between the encoded and original images, ensuring robustness and imperceptibility in practical applications.

Watermarking and steganography models are typically classified as either robust or non-robust. Robust models aim to preserve embedded messages under various distortions,

such as those introduced by printing, camera capture, or noisy digital transmission. In contrast, non-robust models focus on maximizing embedding capacity and imperceptibility but are highly susceptible to modifications [1].

Our work shares conceptual similarities with both image watermarking and steganography, while our primary focus lies within the domain of steganography [1], [3], [4], aiming to develop highly robust embedding techniques that ensure imperceptibility while maintaining resilience guard against various distortions and transformations.

Despite progress in robust steganography—especially [1], [3], [4], [6] several limitations remain that restrict industrial applicability. These include difficulties in preserving high visual similarity, reduced decoding accuracy under real-world conditions, lack of an integrated and deployable framework from encoding process, to detecting encoded part and decoding, absence of a fully integrated and deployable framework encompassing the entire pipeline from encoding to detection and decoding, poor performance on small images, and minimal control over the artifacts introduced by message embedding. Overcoming these issues is crucial for building a practical and reliable steganography model ready for real-world deployment.

In this study, we address the aforementioned challenges by introducing DocSafe—a robust and printable steganography framework consisting of both encoder and decoder components. The design of DocSafe is grounded in two foundational analyses: (1) the identification of effective frequency decomposition strategies for input processing, and (2) the formulation of an optimized covariance loss function to minimize distributional differences—between original and encoded images for the encoder, and between the input and recovered message for the decoder.

Building on the frequency decomposition approach introduced in StampOne [1], we further investigate the role of individual frequency components within both the encoder and decoder networks of our proposed model. All inputs—original images and binary messages in the encoder, and encoded images in the decoder—are first processed using a gradient operator [7] followed by a Haar wavelet transform [8], resulting in a multi-scale decomposition into distinct frequency sub-bands. To adaptively modulate the contribution of each frequency sub-band, we integrate a depthwise convolutional layer that learns importance weights based on the relevance of each spectral component. This enables the network to focus on the most informative bands during both embedding and extraction, improving robustness and fidelity.

To further evaluate the impact of individual sub-bands, we train multiple steganography models using selectively filtered inputs, enabling a controlled analysis of the contribution of each frequency component under various distortion scenarios.

The findings lead to an optimized frequency decomposition strategy, which is applied into the DocSafe framework.

Secondly, we incorporate an optimized covariance-based loss into both the encoder and decoder. To identify the most effective metric for this purpose, we evaluate four established covariance similarity measures: Log-Euclidean Distance, Affine-Invariant Riemannian Metric (AIRM), Jeffrey Divergence, and Stein Divergence, as outlined in the Log-D-CORAL framework [9].

Building on the foundational concepts of frequency decomposition and optimized covariance loss, we redesign a novel architecture for the DocSafe framework (Figure 1). Central to this design are three variants of the Message Embedding Network (MEN)—MEN v1, v2, and v3 (Figure 2). MEN v1 employs a simple linear layer, performing well on homogeneous datasets but lacking generalization to diverse sets like COCO [10]. To address this, MEN v2 and v3 introduce learnable embedding modules with different activation strategies: LeakyReLU in v2 and a Snake+GELU combination in v3, enhancing adaptability and non-linearity. Additionally, we design a lightweight, deployable decoder inspired by BiSeNet [11], featuring a dual-branch structure to process low and high frequency features separately. This architecture balances accuracy, memory efficiency, and speed, making it suitable for real-time and edge applications.

Experimental results and ablation studies confirm the superior performance of DocSafe over existing robust steganography models. It consistently preserves high perceptual quality across both face and object datasets. In print-based evaluations, DocSafe M-1 achieves a 100% decoding rate on face images down to 2×2 cm, while M-2 and M-3 attain 100% at 3×3 cm and up to 85% at 2×2 cm on complex object datasets—surpassing all previous print-proof models.

Furthermore, the newly proposed decoder significantly reduces memory usage: compared to the original U-shaped decoder from StampOne (approximately 235 MB), the redesigned decoder achieves similar or better performance using only 77 MB, enhancing its suitability for deployment in resource-constrained environments.

In summary, the main contributions of this work are:

- A detailed analysis and optimization of frequency decomposition as a preprocessing strategy for robust steganography.
- An in-depth evaluation and integration of covariance-based loss functions to minimize distributional discrepancies and improve generalization.
- A redesign of the Message Embedding Networks (MENs) and decoder architecture to enable efficient message hiding within small, targeted regions of the image.

In Section II, we review existing image steganography models and highlight the need for a more robust and deployable solution. Section III presents two key analyses foundation—frequency decomposition and covariance loss optimization—that guide our framework design. Section IV details the DocSafe architecture, including the encoder, decoder, and message embedding variants. Finally, Section V

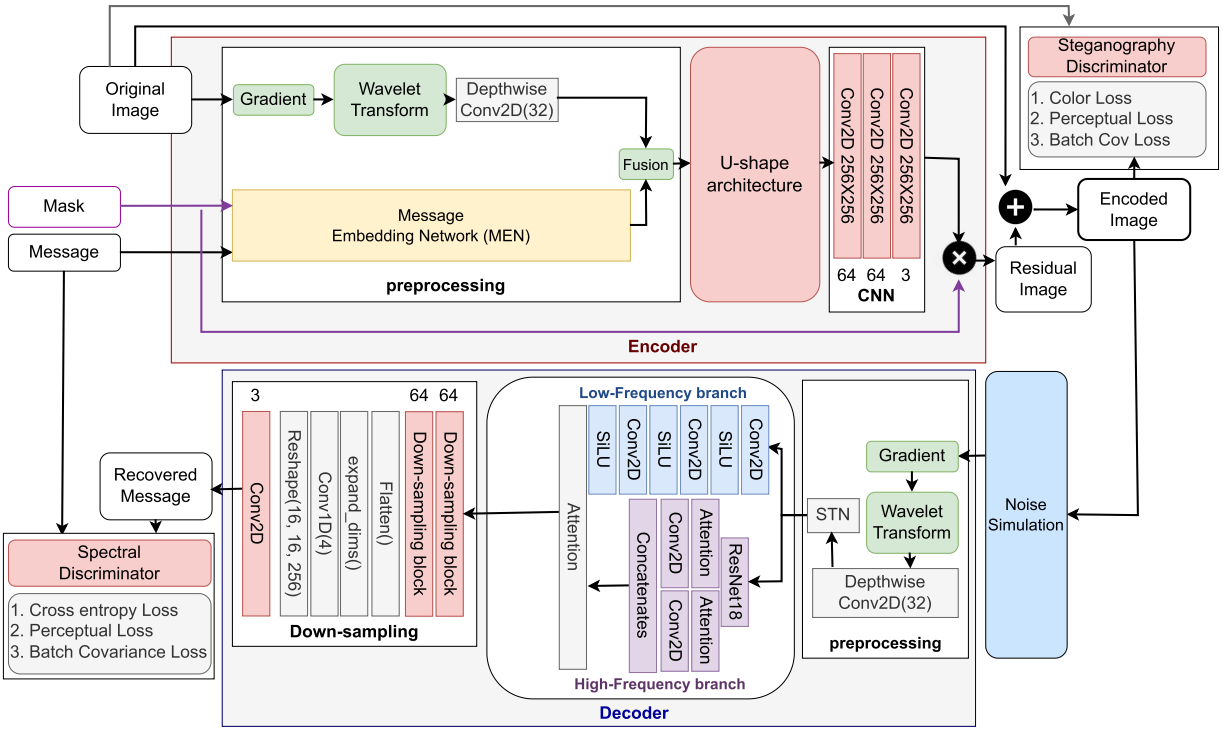

FIGURE 1. The full architecture of DocSafe consists of an encoder and a decoder. A dedicated preprocessing module is first applied to the inputs. The original image is transformed into gradient maps and wavelet sub-band representations to extract frequency-domain information. In parallel, the message input is processed through the Message Expansion Network (MEN). The resulting features are then fused and passed through the U-Net and convolutional layers of the encoder, which generates a residual image. This residual is added to the original image to produce the encoded image. To simulate real-world distortions, the encoded image is degraded using random noise transformations. Before decoding, a preprocessing stage enhances high-frequency components to facilitate message recovery. The decoder then reconstructs the embedded message from the distorted input. During training, different loss functions are applied to the encoder and decoder, and two discriminators are incorporated to further improve performance.

reports experiments evaluating DocSafe under digital and physical distortions, compared to state-of-the-art methods.

II. RELATED WORK

Our goal is to develop robust steganography models capable of generating a stamp-like guard layer to ensure ID authenticity and support brand protection. To this end, we review watermark and steganography methods with a focus on robustness to real-world distortions and applicability in security-critical scenarios.

Non-Robust steganography, Traditional steganography models that do not utilize deep learning techniques attempt to conceal messages within the spatial domain [12] or frequency domain [8], [13], [14], [15] of images. These approaches rely on hand-crafted features [14], learning-based techniques [3], [4], [16], or dynamically selecting the embedding bit-plane (LSB, Bit2SB, or Bit3SB) based on local image characteristics [17]. However, these methods generally lack robustness against a range of distortions and attacks introduced by real-world processes such as printing, scanning, photography, and digital transmission through social media platforms.

In contrast, deep learning-based steganography models demonstrate significantly higher robustness against multiple noise sources [18]. With the continued evolution of this field, recent models [1], [2], [3], [4] have been developed that exhibit high robustness and printer-proof capabilities, effectively resisting image distortions caused by digital transformations, printing processes, and camera capture artifacts.

Robust steganography, HiDDeN [18] is the first robust deep learning-based steganography model that incorporates random noise simulation and image distortions during training to achieve digital robustness. The applied distortions include Gaussian blurring, pixel-wise dropout, cropping, and JPEG compression, ensuring the model's ability to withstand common digital transformations. HiDDeN leverages Generative Adversarial Networks (GANs) to generate encoded images, embedding a bit-string array within a cover image while maintaining imperceptibility and resilience against various forms of degradation.

StegaStamp [4] introduces a highly robust steganography model with printer-proof capability, enabling the decoding of hidden messages even after printing and scanning. This robustness is achieved by incorporating various image

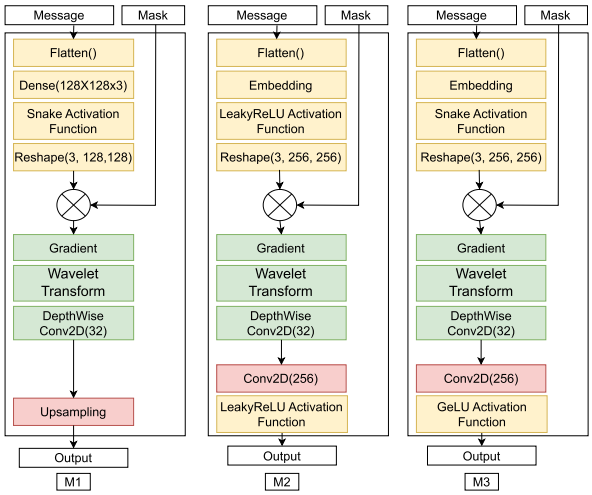

FIGURE 2. We explore three different message embedding networks. M1 is designed for Model v1, where a linear layer is employed to project the input message to a higher-dimensional space. M2 and M3 are used in Model v2 and Model v3, respectively. In both M2 and M3, a learnable embedding layer—commonly used in natural language processing models—is utilized to map discrete message tokens into continuous high-dimensional representations. The primary distinction between these two networks lies in their activation functions: M2 applies two successive LeakyReLU activations, while M3 incorporates a combination of Snake and GELU activations to better capture non-linear message transformations.

distortions into the training process, including warping, rotation, and color transformations, which simulate real-world degradations encountered during the printing process. To further enhance its performance, StegaStamp modifies the U-Net architecture of the encoder by removing the bottleneck, allowing for better feature preservation and information flow. Additionally, it employs the Learned Perceptual Image Patch Similarity (LPIPS) metric [19] as a perceptual loss function, significantly improving the visual quality of encoded images.

CodeFace [3] represents the next generation of StegaStamp, specifically designed for highly robust steganography in frontal face detection, with a primary application in identification cards (ID cards). CodeFace not only enhances the perceptual quality of frontal face-encoded images by minimizing facial feature discrepancies between original and encoded images—leveraging models such as FaceNet [20]—but also introduces a defined conditions, making it highly suitable for real-world industrial applications in secure identity verification.

RoSteALS [16] explores modifying the latent code of an image by introducing controlled variations, uncovering the possibility of embedding meaningful messages (secrets) directly into the latent space without perceptually altering the image content. The model enables message retrieval through an additional decoder network that extracts the embedded information from the encoded images. To achieve this, RoSteALS employs an autoencoder architecture where both the image and the message are passed through the downsampling phase of the autoencoder, mapping them into their respective latent representations, typically $4\times$ or $8\times$ smaller than the original inputs. The latent codes of the image and message are then combined before passing through the upsampling network to reconstruct the encoded image. A key strength of RoSteALS is its robustness against severe digital image perturbations. This resilience is achieved by introducing multiple noise sources from ImageNet-C [21] during training. The injected distortions include brightness, saturation, contrast variations, JPEG compression artifacts, and spatter noise, enhancing the model's ability to withstand digital image degradations while preserving message integrity.

StampOne [1] is a highly robust steganography model designed with printer-proof capabilities. It focuses on developing a unified preprocessing strategy for both the encoder and decoder, enhancing high-frequency preservation to achieve a balanced representation of low- and high-frequency components in the output. A key advantage of StampOne is its modular design, allowing the use of any arbitrary U-shaped network for message embedding, as long as the proposed preprocessing framework is applied before the encoder and decoder. To optimize perceptual quality in the encoded images, StampOne introduces multiple Message Preparation Networks, ensuring superior imperceptibility while maintaining accurate message recovery. Additionally, the model incorporates a Spectral Discriminator within the decoder during training, which specifically enhances high-frequency components to improve message retrieval accuracy and robustness.

The Watermark Anything Model (WAM) [22] is a watermarking framework that leverages segmentation to embed watermarks specifically in regions of interest, while leaving the remaining parts of the image untouched. This selective embedding strategy enables WAM to hide and reliably decode watermarks even from small localized regions. Moreover, WAM demonstrates strong robustness against common image manipulations, including inpainting and splicing attacks.

RiemStega [6] is another high-level robust steganography model developed for print-proof applications. Structurally inspired by CodeFace and StegaStamp, RiemStega adopts their encoder-decoder architecture and inherits their robustness characteristics, enabling reliable message recovery even under severe distortions such as printing, scanning, geometric warping, and additive noise. A major contribution of RiemStega is the integration of a Riemannian loss function (RiemL), which minimizes the covariance distance between the feature distributions of the encoded and original images, thereby enhancing the model's domain robustness and retrieval accuracy.

Some related works focus more on watermarking. For example, Dzhanashia and Evsutin [23] proposed a U-Net encoder with a CNN-based decoder to embed and recover a 2-bit message in 32×32 images, showing robustness to geometric distortions. Similarly, Lakasek and Elhadi [24] introduced an imperceptible blind watermarking method

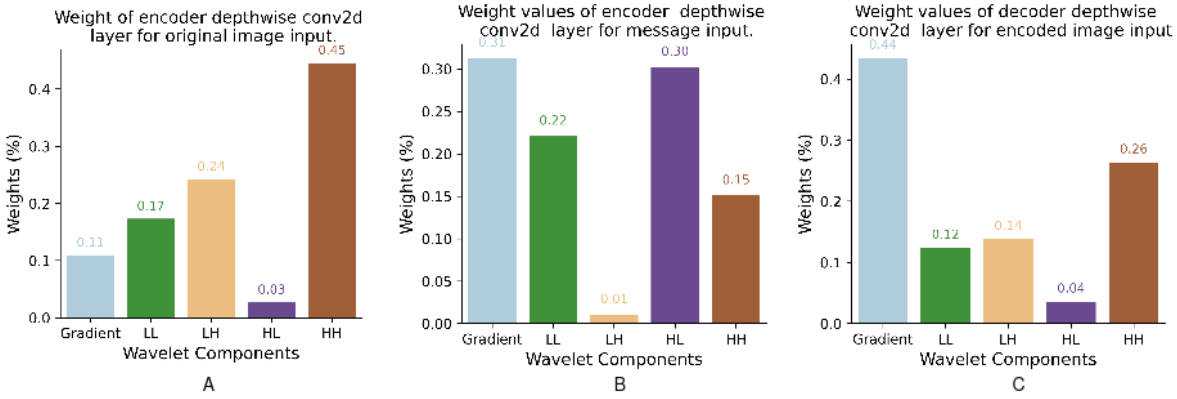

FIGURE 3. The input RGB image and message are first processed through a gradient operator and wavelet transform, expanding it from 3 to 12 channels (R, G, B per sub-band). A depthwise convolutional layer in both the encoder and decoder assigns learnable weights to each wavelet sub-band channel, adaptively controlling their contribution and indicating the significance of each frequency component during processing.(A) Controlling the passage of wavelet sub-band levels derived from the original image.(B) Modulating wavelet sub-band levels associated with the secret message during embedding.(C) Adjusting wavelet sub-band levels in the encoded image as part of the decoder's reconstruction process.

using the Non-Subsampled Shearlet Transform (NSST), embedding watermark information into the high-frequency HH directional sub-band to better preserve edge and texture information.

Inspired by StampOne and StegaStamp, we adopt a frequency-balance strategy; however, we conduct a detailed analysis of this mechanism and introduce targeted improvements. Drawing from RiemStega, we incorporate covariance-based loss functions into both the encoder and decoder, systematically evaluating different covariance formulations to identify the most effective configuration for each component. Finally, we design a novel decoder architecture and introduce an automatic masking mechanism in the encoder, enabling the message to be embedded selectively into semantically appropriate regions of the image.

III. FOUNDATIONS: FREQUENCY ANALYSIS, COVARIANCE LOSS OPTIMIZATION

This research makes two key contributions. First, it offers a detailed analysis of the role of low and high frequency components in robust steganography, revealing how each contributes to message integrity. Second, it presents a comprehensive evaluation of covariance-based loss functions for reducing distributional discrepancies during encoding and decoding. Together, these insights enable the development of a more robust and industrially applicable framework.

A. EFFECT OF FREQUENCY-LEVEL SELECTION

The attenuation of low or high frequency information presents a critical challenge for robust steganography models [1], due to three main factors:

- Vulnerability to image distortions: Various distortions—arising from digital transmission, printing, or camera capture—can degrade specific frequency components, compromising the integrity and quality of the embedded message.
- Decoder reliance on high-frequency or gradient features: Robust steganography decoders typically detect higher luminance transitions rather than relying on homogeneous luminance areas. Accurate message retrieval thus depends on preserving edge and gradient information, which resides in the high-frequency domain, rather than low-frequency color variations.
- Perceptual quality of encoded images: Maintaining a balanced frequency distribution in the encoder output is essential to ensure imperceptibility while preserving sufficient information for reliable message extraction. This balance supports both visual fidelity and decoding accuracy.

Building on prior efforts to address frequency imbalance in robust steganography models [1], [4], this work adopts a frequency-aware preprocessing strategy inspired by StampOne, applied to both the encoder and decoder inputs.

During the preprocessing stage, each input RGB image undergoes both gradient extraction and Haar Wavelet Transform decomposition. The wavelet transform splits each color channel into four frequency sub-bands—Low-Low (LL), High-Low (HL), Low-High (LH), and High-High (HH)—yielding a total of 12 wavelet channels across the three color channels. These are subsequently concatenated with a 3-channel gradient map, resulting in a 15-channel composite input representation.

To regulate the contribution of each component, a depthwise convolutional layer is introduced. This layer learns adaptive weights for the individual frequency sub-bands and gradient channels, modulating their influence during both encoding and decoding. Serving as a significance indicator, this mechanism allows the network to prioritize informative spectral components while attenuating less relevant ones. We gain insight into which wavelet sub-bands are consistently emphasized by analyzing the learned weights from this depthwise convolution across the training data. The learned

TABLE 1. Evaluating of encoded image quality and decoder performance across six models—M-ALL, M-Grad, M-LL, M-HL, M-LH, and M-HH—each defined by different frequency filtering strategies applied during preprocessing. Trained on a face image dataset, these models enable a controlled comparison of how specific spectral components affect visual fidelity and message retrieval accuracy.

Methods	Encoded images quality				(B) **Bit accuracy**			
	SSIM (⇑)	LPIPS (⇓)	ColorHisto (⇓)	FID (⇓)	5×5 cm	4×4 cm	3×3 cm	2×2cm
M-All	**0.91**	**0.03**	0.06	**11.21**	**92**	**80**	**62**	**12**
M-Grad	0.83	**0.03**	0.06	19.45	90	75	57	2
M-LL	0.71	0.11	0.09	16.70	32	27	20	0
M-HL	0.70	0.11	0.11	12.44	42	30	12	0
M-LH	0.64	0.15	0.11	20.74	62	57	50	2
M-HH	0.80	0.04	**0.05**	30.54	55	35	32	5

weight values for each frequency sub-band and gradient channel are illustrated in Figure 3.

In Figure 3 (A), we illustrate the learned weights assigned to the frequency components of the original image. The results show a clear preference for high-frequency sub-bands, with the HH and LH components receiving the highest weights—approximately 45% and 24%, respectively—followed by LL (17%), gradient features (11%), and HL (3%).

In contrast, the learned frequency weights for the message input, shown in Figure 3 (B), exhibit a markedly different distribution. The gradient component receives the highest weight (31%), followed by the HL sub-band (30%), LL (22%), HH (15%), and LH (1%). This distribution highlights the critical role of edge information—captured by the gradient and HL components—in effective message embedding and recovery. The model's emphasis on these features suggests that sharp transitions and directional structures are particularly informative for accurately hiding and recovering the message.

Continuing the analysis, Figure 3 (C) shows the learned frequency weights for encoded images as processed by the decoder. Notably, the distribution closely mirrors that of the message input in the encoder. The gradient component receives the highest weight (44%), followed by HH (26%), LH (14%), LL (12%), and HL (4%). This alignment indicates that the decoder heavily relies on high-frequency and edge-related features—particularly gradients and HH components—for accurate message reconstruction, further emphasizing the critical role of these spectral cues across the entire encoding-decoding pipeline.

These findings suggest that message embedding cannot rely solely on a fixed frequency range. Instead, the optimal frequency emphasis may vary depending on the input type. While high-quality encoded images benefit from preserving higher-frequency components such as HH, successful message embedding and recovery rely more heavily on the gradient information than on isolated wavelet sub-bands.

In the next phase of our analysis, we conducted a controlled evaluation to quantify the individual contributions of specific wavelet frequency components to steganography performance. To this end, we manually filtered selected sub-bands and trained six distinct models, each utilizing a different combination of frequency inputs. These include: M-ALL, which incorporates all components (Gradient, LL, LH, HL, HH); M-Grad, using only the gradient; M-LL, combining the gradient with the lowest low-frequency sub-band (LL); M-HL, pairing the gradient with the highest low-frequency sub-band (HL); M-LH, including the gradient and the lowest high-frequency sub-band (LH); and M-HH, combining the gradient with the highest high-frequency sub-band (HH). This experimental design allows for a direct comparison of each sub-band's effectiveness in supporting robust steganography models. The results are summarized in Table 1.

Notably, all other tested combinations of frequency components failed to converge during training, underscoring the importance of informed frequency selection for ensuring model stability and robustness.

The best overall performance is achieved by the M-ALL model, which utilizes the complete set of wavelet sub-bands along with gradient information. This configuration yields the highest scores in both perceptual quality of the encoded images and decoding accuracy on printed samples.

In terms of perceptual quality, the performance ranking from best to worst is: M-ALL, M-Grad, M-HH, M-LL, M-HL, and M-LH. For decoder accuracy, the ranking is: M-ALL, M-Grad, M-LH, M-HH, M-HL, and M-LL.

These results indicate two key insights: first, a complete set of frequency sub-bands is essential for achieving optimal image quality and message recovery; second, for improving decoding robustness, particular attention should be given to the LH (low-high) frequency sub-band, which appears to play a critical role in preserving essential features for message reconstruction.

B. DOMAIN ADAPTATION THROUGH COVARIANCE

Deep learning models frequently encounter the domain shift problem—also known as domain adaptation (DA)—where significant discrepancies between the training and testing data distributions lead to reduced performance [9], [25], [26]. This issue is especially critical in steganography, where

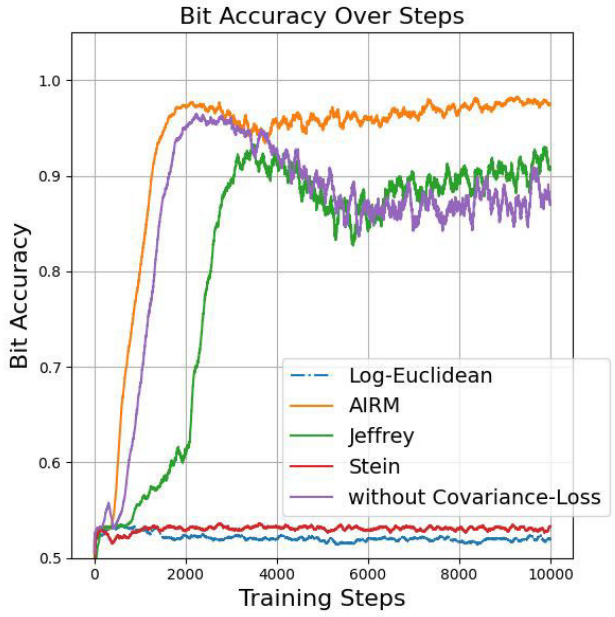

FIGURE 4. We conducted 10,000 training steps for five different steganography models, each employing a distinct formulation for covariance minimization between the input and recovered messages. Among these, the most stable and consistent results were achieved using the Affine-Invariant Riemannian Metric (AIRM).

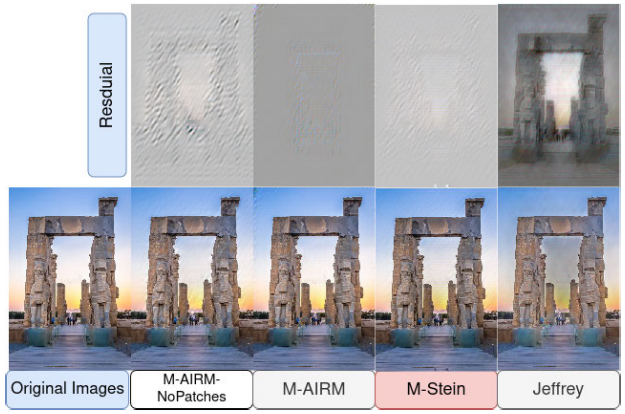

FIGURE 5. Shows examples of encoded images generated using different covariance loss functions. All networks share the same architecture and training conditions; the only variation lies in the type of covariance loss function applied between the original and encoded images in the encoder. Among the tested configurations, M-Stein produced the best visual results, demonstrating superior perceptual quality and fidelity.

synthetic training noise often fails to reflect the complexity and variability of real-world distortions [27], [28], [29].

To address domain shift and accelerate convergence, we propose a batch-wise covariance loss, inspired by the Log-D-CORAL framework [9] and extended by RiemStega [6]. This loss aligns the second-order statistics—specifically, covariance matrices—between paired input and target batches (e.g., original and encoded images).

To compute the loss, we first apply a patch embedding and positional encoding function F_p, following the Vision

TABLE 2. Evaluation of visual quality across four models—M-AIRM-NoPatches, M-AIRM, M-Stein, and M-Jeffrey—trained with different covariance losses. All share the StampOne architecture; the key variation is in the covariance similarity metric used.

Methods	covariance loss type for encoder SSIM (⇑)	LPIPS (⇓)	ColorHisto (⇓)	FID (⇓)
M-AIRM-NoPatches	0.79	0.04	0.05	30.54
M-AIRM	0.82	**0.03**	0.05	16.77
M-Stein	**0.86**	**0.03**	**0.04**	**14.83**
M-Jeffrey	0.80	0.06	0.12	25.30

Transformer (ViT) paradigm [30]. This function segments each image into non-overlapping patches, flattens them into vectors, and augments them with learnable positional encoding to preserve spatial structure. The resulting patch embeddings are denoted as $f_{in} = F_p(f_{in})$ and $f_{ta} = F_p(I_{ta})$, representing the input and target batches, respectively. This transformation reformulates each image as a sequence of patch-level feature vectors, facilitating the computation of second-order statistics, specifically covariance matrices, across the batch. These covariance matrices—denoted as C_{in} for the input and C_{ta} for the target—serve as the basis for the batch-wise covariance loss.

To identify the most effective approach for computing and minimizing covariance differences in the encoder and decoder networks, we evaluate the similarity between covariance matrices using four established metrics: Log-Euclidean Distance (d_{LE}), Affine-Invariant Riemannian Metric (AIRM) (d_{AIRM}), Jeffrey Divergence (d_{jeffrey}), and Stein Divergence (d_{stein}). They are computed both between input and recovered messages (decoder path) and between original and encoded images (encoder path), separately.

The Log-Euclidean Distance, as used in Log-D-CORAL [9], is computed by:

$$d_{\text{LE}}(C_{in}, C_{ta}) = \frac{1}{4d^2} \|\log(C_{in}) - \log(C_{ta})\|_F^2 \qquad (1)$$

where $\log(\cdot)$ denotes the matrix logarithm and $||\cdot||_F$ is the Frobenius norm, defined as $\|A\|_F = \sqrt{\sum_{i,j} |A_{ij}|^2}$.

The Affine-Invariant Riemannian Metric (AIRM), adopted in RiemStega [6], provides a more geometrically accurate distance between Symmetric Positive Definite (SPD) matrices and is defined as:

$$d_{\text{AIRM}}(C_{in}, C_{ta}) = \left\| \log(C_{in}^{-1} C_{ta}) \right\|_F \qquad (2)$$

This formulation preserves the intrinsic Riemannian manifold structure of the covariance space, enabling more robust alignment of second-order statistics between input and target domains. By respecting the geometric properties of symmetric positive definite (SPD) matrices, it enhances domain invariance and promotes better generalization in steganography applications.

The Jeffrey Divergence (JD), a symmetric version of the Kullback–Leibler divergence for Gaussian distributions with

equal means, is computed as:

$$d_{\text{jeffrey}}(C_{in}, C_{ta}) = \frac{1}{2}\,\text{tr}(C_{in}^{-1} C_{ta}) + \frac{1}{2}\,\text{tr}(C_{ta}^{-1} C_{in}) - d \quad (3)$$

where tr denotes the trace (sum of diagonal elements) of a matrix ($\text{tr}(A) = \sum_{i=1}^{d} A_{ii}$).

The Stein Divergence (also known as the Jensen–Bregman LogDet Divergence) is a determinant-based metric that is computationally efficient and well-suited for SPD matrices:

$$d_{\text{stein}} = \sqrt{\log\left|\frac{C_{in} + C_{ta}}{2}\right| - \frac{1}{2}\log|C_{in}| - \frac{1}{2}\log|C_{ta}|} \quad (4)$$

To determine the most effective covariance minimization loss for the decoder, the steganography networks were trained independently for 10, 000 steps, the results are summarized in Figure 4. The models employing Log-Euclidean Distance and Stein Divergence failed to converge, with bit accuracy remaining around 0.54. Other loss functions yielded performance comparable to the baseline (i.e., no covariance loss applied), suggesting limited utility in this context. In contrast, the AIRM demonstrated superior stability—particularly after the introduction of random noise distortions at step 2,000—and continued to show consistent improvement throughout extended training up to 310,000 steps. These results highlight AIRM as the most effective option for enhancing decoder robustness under noisy conditions.

The purple line in Figure 4 represents the case where AIRM is applied without batch-wise computation—similar to the configuration used in the RiemStega model. Under this condition, the decoder exhibits performance comparable to the baseline scenario without any covariance minimization loss, indicating that AIRM without batch-wise adaptation provides limited benefit.

After identifying AIRM as the most effective option for use as a decoder loss function, we evaluated five different covariance-based loss functions for the encoder. Each model was trained for 160, 000 steps. The comparative results—assessing both the perceptual similarity between encoded and original images, as well as the decoder's performance in recovering messages from printed encoded images—are summarized in Table 2. Qualitative examples of the encoded images generated by each configuration are presented in Figure 5.

IV. INTRODUCE AN EFFICIENT AND ADAPTABLE NETWORK

Building on our analysis of frequency selection and covariance loss, we introduce the optimized steganography model DocSafe, illustrated in Figure 1. The encoder builds upon StampOne with key enhancements, including a redesigned Message Embedding Network supporting spatial masking. The decoder is fully rearchitected—drawing inspiration from BiSeNet [11]—to enhance feature extraction and message reconstruction while remaining computationally efficient.

As illustrated in Figure 1, the encoder receives the original image, mask, and message as inputs. The image is first processed using gradient and wavelet transforms to enhance spatial–frequency representations. The message is passed through a linear embedding layer, reshaped to the image size, and combined with the transformed image features before being processed by a U-Net–based encoder that predicts a residual image. The mask is applied within the Message Enhancement Network (MEN) and again at the end of the encoder to constrain the residual. The encoded image is generated by adding the residual to the original image. During training, several distortions are applied to the encoded images before they are passed to the decoder, which uses a similar preprocessing stage to recover the embedded message. The model is trained using multiple loss functions for the encoder and decoder, along with two discriminators to improve robustness and visual quality.

A dedicated preprocessing module is first applied to the inputs. The original

The complete encoder–decoder pipeline is illustrated in Figure 6. As shown in the figure, we employ Bose–Chaudhuri–Hocquenghem (BCH) error correction coding, which plays a crucial role in improving the robustness and decoding accuracy of the system.

A. INTEGRATION OF FREQUENCY AND COVARIANCE FOUNDATIONS

Guided by our frequency analysis, we adopt the M-ALL input configuration for the DocSafe architecture—allowing all frequency components to pass through—while introducing a custom initialization scheme for the depthwise convolutional weights based on the learned importance of each sub-band. Specifically, weights are initialized as follows: for the original image, HH and LH are set to 0.6 and 0.4; for the message input, the gradient and HL components are set to 0.5 each; and for the encoded image in the decoder, the gradient and HH components are set to 0.6 and 0.4, respectively. All other weights are initialized to zero. This strategy biases the network toward the most informative components, promoting faster convergence and improved robustness.

In parallel, based on our covariance loss analysis, we employ Stein divergence as the loss function between original and encoded images during encoder training, and the Affine-Invariant Riemannian Metric (AIRM) to align covariance between the input and recovered message in the decoder.

B. ENCODER

The primary enhancements to the encoder are centered around the Message Embedding Networks (MENs), as depicted in Figure 2. Unlike the embedding mechanism used in StampOne, which distributes the message uniformly across the entire image, our goal is to enable selective message embedding—targeting semantically relevant regions such as faces or objects while avoiding less informative background areas. To achieve this, we introduce three newly designed and distinct architectures for the embedding

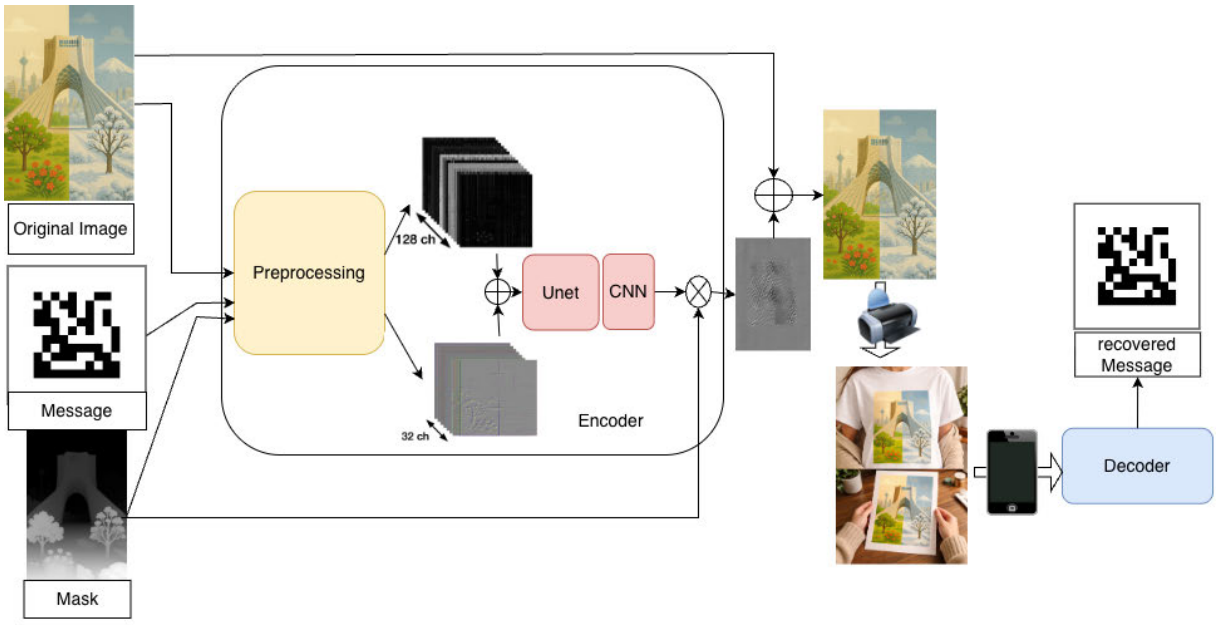

FIGURE 6. Full Encoder–Decoder Pipeline of DocSafe. The input string is converted into a 256-bit binary sequence using BCH error correction [31] and reshaped to 16 × 16. The original image and mask are resized to 256 × 256. The encoder takes the image, mask, and message as inputs. After preprocessing, message and image features are projected to $(B, 128, 256, 256)$ and $(B, 32, 256, 256)$, respectively, and passed through U-Net and convolutional layers to generate a residual image. This residual is added to the original image to obtain the encoded image. The encoded image is printed and captured by a smartphone camera. The captured image is then processed by the decoder to recover the embedded message.

module, referred to as MEN-1, MEN-2, and MEN-3, each tailored to improve control, flexibility, and generalization across diverse image domains.

Message Embedding Network: MEN-1 uses a linear layer to project the input message into a $(3, 128, 128)$ feature map, activated by Snake and guided by a monocular depth map to focus embedding on semantically relevant regions. This is enhanced through frequency modules and upsampled to match the encoder input shape. While effective on structured datasets (e.g., faces or vehicles), MEN-1 shows limited generalization to diverse datasets like COCO or DeepFashion.

To improve generalization on heterogeneous datasets, MEN-2 and MEN-3 replace the linear layer with a learnable embedding layer that maps messages directly to $(3, 256, 256)$ feature maps. After frequency enhancement and a convolution with 1×1 kernel size to reach $(256, 256, 256)$, the output is fused with the cover image. MEN-2 uses LeakyReLU, while MEN-3 combines Snake and GeLU activations. Although they slightly underperform MEN-1 on domain-specific tasks, MEN-2 and MEN-3 offer better robustness and adaptability to diverse real-world data.

The visual outcomes of encoded images generated using the three proposed Message Embedding Networks are presented in Figures 7 and 8, corresponding to unmasked (full-image) and masked embedding, respectively. These comparisons highlight the influence of each design on the spatial localization and frequency characteristics of the outputs.

U−shape network: we exclusively use and evaluate the encoder with AttentionVNet [32] as the U-shaped network in DocSafe. AttentionVNet is particularly effective in enhancing high-frequency retention, making it a suitable choice for robust steganography embedding. However, it is important to note that the U-shaped network in DocSafe can be replaced with other architectures, depending on specific application requirements. At the final stage of the DocSafe encoder network, the residual output is element-wise multiplied by the mask to ensure that message embedding is confined to the regions of interest, leaving the non-target areas of the encoded image unaltered.

C. DECODER

For DocSafe, we propose a new decoder architecture, replacing the conventional U-shaped network with a dual-branch structure consisting of a low-frequency passing branch (Spatial Path) and a high-frequency passing branch (Context Path), as illustrated in Figure 1.

The low-frequency passing branch is composed of three 2D convolutional layers, each followed by the SiLU activation function and batch normalization to preserve spatial information while maintaining computational efficiency.

The high-frequency passing branch utilizes ResNet (ResNet-34 and ResNet-18) as its backbone, enhanced by a Signal Attention Network at the final stage. To adapt ResNet for our purpose, we remove its two final fully connected layers and extract two feature maps from the last 2D convolutional layer. These feature maps are then

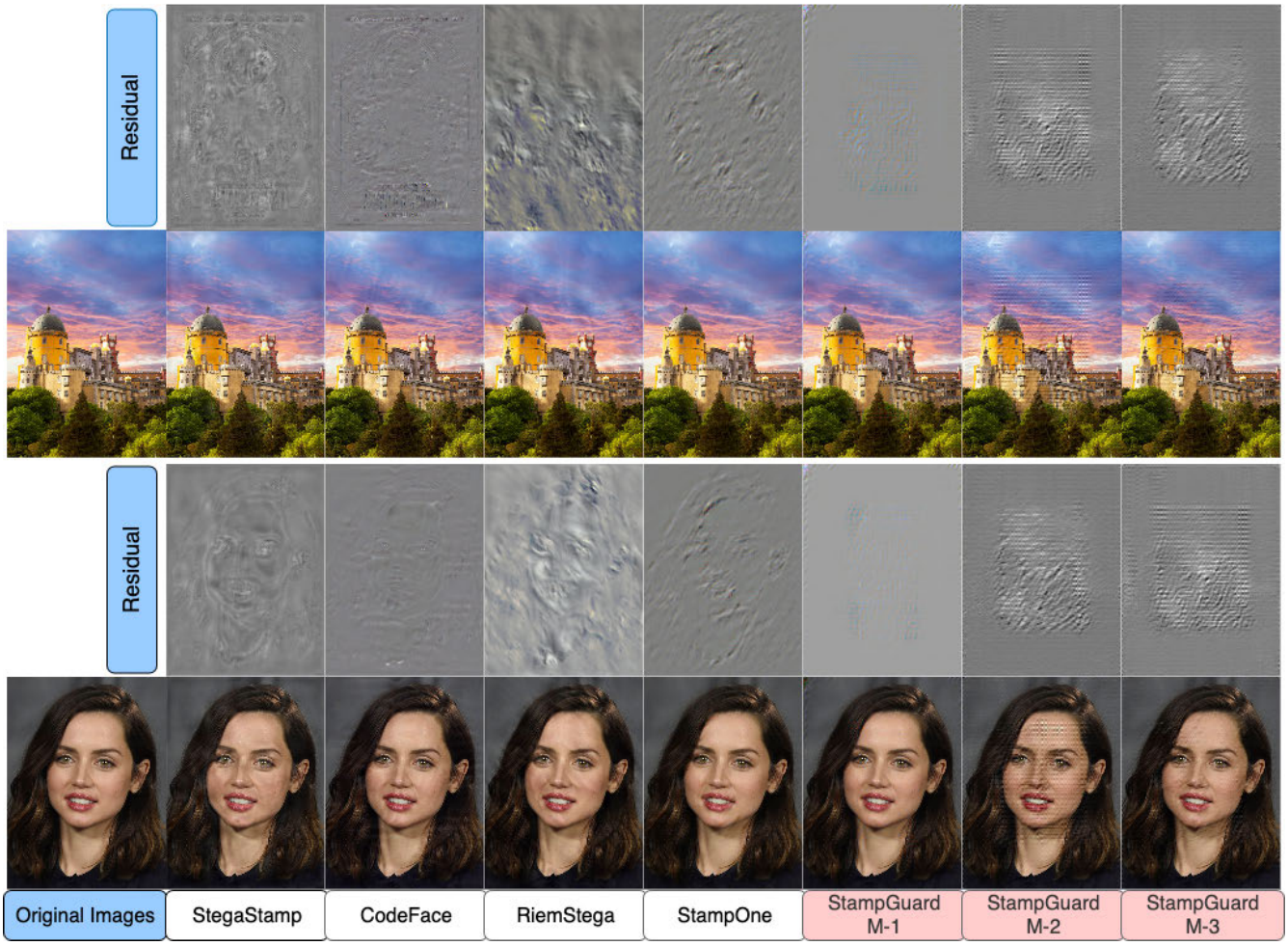

FIGURE 7. Representative samples of encoded images without spatial masking are presented, where the message is uniformly embedded across the entire image, leading to more globally distributed visual modifications. Model M-1 produces high-quality encoded images; however, its decoder performs well only on the face image dataset. In contrast, Models M-2 and M-3 generate encoded images with lower visual quality compared to M-1, but demonstrate improved and acceptable decoder performance on the object image dataset.

processed through two 1D convolutional layers (kernel size = 1), followed by an Attention Refinement Module (ARM) [11], which helps refine and enhance high-frequency features. The outputs are then merged to form the final representation of the high-frequency passing branch.

Finally, the low-frequency and high-frequency branches are fused using a signal-gated attention mechanism [32], similar to the approach used in the encoder. This design achieves an optimal balance between spatial detail retention and high-frequency feature enhancement, while also reducing the network size and improving the model's efficiency and processing speed. As a result, it significantly enhances message reconstruction quality and robustness in steganography applications.

D. DISCRIMINATORS

Encoder discriminator: We employ a standard discriminator in the encoder training pipeline, which receives both the original image (I_{org}) and the encoded image (I_{enc}) as inputs. To optimize the discriminator's behavior more effectively, we utilize two separate Adam optimizers with distinct objectives.

The first optimizer is dedicated solely to reinforcing the discriminator's ability to recognize real images, by minimizing the following objective:

$$DL_{real} = Mean(log(D(I_{org}))) \tag{5}$$

The second optimizer operates jointly with the rest of the network's loss functions and aims to minimize the difference in discriminator outputs between the original and encoded images. This is formulated as:

$$L_{real} = Mean(log(D(I_{org})) - log(D(I_{enc})) \tag{6}$$

This two-optimizer setup allows for stable adversarial training while also guiding the encoder to generate visually convincing outputs that remain indistinguishable from the original images in the discriminator's feature space.

Decoder discriminator: We incorporate a spectral discriminator into the decoder training, inspired by StampOne,

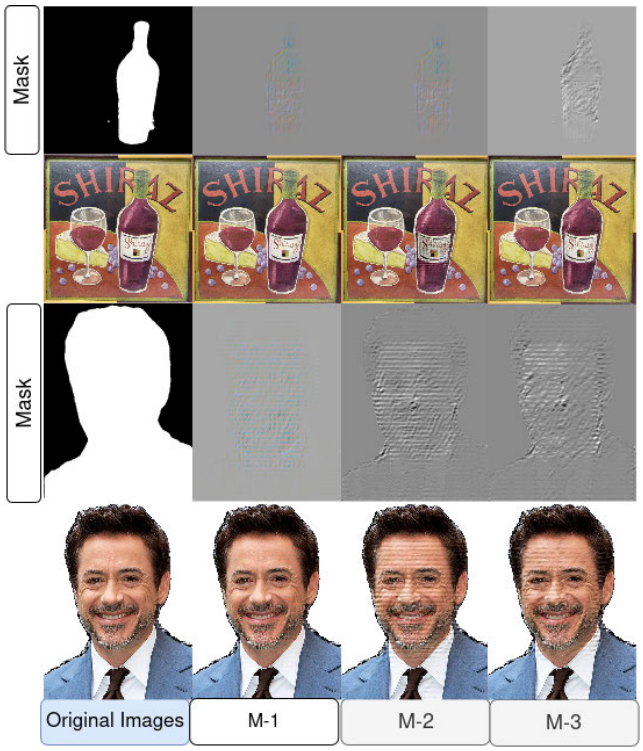

FIGURE 8. Examples of encoded images where the message is embedded exclusively within the masked regions.

to enhance the recovery of high-frequency details essential for accurate message reconstruction.

In this design, both the original message (m) and the reconstructed message ($m_{\text{rec}} = \text{Dec}(I_{\text{enc}})$) are transformed into the frequency domain via the Fast Fourier Transform (FFT). A high-pass filter is applied to suppress low-frequency components, resulting in $f_{\text{FFT}}(m)$ and $f_{\text{FFT}}(m_{\text{rec}})$, which are then input to the spectral discriminator D [7].

The adversarial loss is defined as:

$$\begin{aligned}\mathcal{L}\text{adv} = \mathbb{E}m \sim p_{\text{data}}\left[\log D(f_{\text{FFT}}(m))\right] \\ + \mathbb{E}m_{\text{rec}} \sim p_{\text{rec}}\left[\log(1 - D(f_{\text{FFT}}(m_{\text{rec}})))\right]\end{aligned} \tag{7}$$

where, $f_{\text{FFT}}(\cdot)$ denotes the composition of the FFT and high-pass filtering operations. This adversarial loss encourages the decoder to accurately reconstruct high-frequency components of the embedded message, which is crucial for robustness under real-world degradations.

E. LOSS FUNCTIONS

Covariance loss functions: the AIRM formulation (L_{AIRM}) is selected as the covariance loss function for the decoder and the Stein divergence (L_{Stein}) for the encoder.

Histogram color loss function: To minimize color discrepancies between the original and encoded images, we adopt a color representation strategy inspired by prior work in chroma-aware image processing [33], [34], [35]. To computing the Histogram color loss function (L_{ColorH}), we convert images from the RGB color space to a two-dimensional log-chroma space, which offers a perceptually meaningful representation of chromatic content. In this space, we compute histograms of chromatic information—denoted as H_{org} for the original image and H_{enc} for the encoded image. To quantify the color similarity between the original and encoded images, we define a color histogram loss function, L_{ColorH}, based on the Euclidean distance between the square roots of the two histograms:

$$L_{\text{ColorH}} = \frac{1}{\sqrt{2}}\left|H_{\text{org}}^{1/2} - H_{\text{enc}}^{1/2}\right|_2 \tag{8}$$

Here, the square root is applied element-wise to ensure the metric is both symmetric and differentiable. This loss function encourages the encoded image to preserve the chromatic characteristics of the original, thereby enhancing perceptual fidelity—an essential factor in visually imperceptible steganography.

Perceptual loss functions: For encoder, we also use the Learned Perceptual Image Patch Similarity (LPIPS) metric [19] (L_{LPIPS}) and Mean Squared Error (MSE) (L_{MSE}) between the original and encoded images as the primary perceptual loss functions.

Encoder loss functions: Full loss function for encoder is,

$$\begin{aligned}L_{\text{Encoder}} = \lambda_{Stein} \times L_{Stein} \\ + \lambda_{Color} \times L_{\text{ColorH}} \\ + \lambda_{LPIPS} \times L_{LPIPS} \\ + \lambda_{\text{MSE}} \times L_{\text{MSE}}\end{aligned} \tag{9}$$

where, the weights of each component, denoted by λ_{Stein}, λ_{Color}, λ_{LPIPS}, and λ_{MSE}, are adjusted to balance their respective contributions.

Decoder loss functions: The primary objective function used to train the decoder is the binary cross-entropy loss (L_{BCE}), which measures the difference between the original and recovered binary messages. To further enhance robustness—particularly in the presence of domain shifts and noise—we also incorporate the batch-wise AIRM covariance loss (L_{AIRM}), which aligns the second-order statistics between the input and the decoded messages.

The combined decoder loss function is defined as:

$$L_{\text{Decoder}} = \lambda_{\text{AIRM}} \times L_{\text{AIRM}} + \lambda_{\text{BCE}} \times L_{\text{BCE}} \tag{10}$$

where λ_{AIRM} and λ_{BCE} are scaling coefficients that control the contribution of each loss term. These hyperparameters are carefully tuned to ensure a proper balance between statistical alignment and message reconstruction accuracy, allowing the decoder to perform reliably under a variety of real-world conditions.

F. NOISE AUGMENTATION

To improve decoding robustness, DocSafe employs progressive noise augmentation during training, including Gaussian noise, JPEG artifacts, grayscale conversion, affine transformations, cropping, and other distortions [36], [37]. This strategy helps the decoder adapt to real-world noise and enhances overall resilience.

TABLE 3. Encoded image quality metrics for the face dataset presented in two parts: (A) reports the results when the message is embedded across the entire image. (B)presents the results when a spatial mask is applied, restricting message embedding to specific facial regions. The results are evaluated on the object dataset, which consists of 2,000 face images randomly selected from the FFHQ test set.

Methods	(A) no mask					(B) using mask				
	PSNR (⇑)	SSIM (⇑)	LPIPS (⇓)	ColorHisto (⇓)	FID (⇓)	PSNR (⇑)	SSIM (⇑)	LPIPS (⇓)	ColorHisto (⇓)	FID (⇓)
StegaStamp [4]	27.82	0.69	0.18	0.43	36.92	-	-	-	-	-
Code Face [3]	**32.31**	**0.93**	0.30	0.08	20.70	-	-	-	-	-
RiemStega [6]	32.01	0.90	**0.02**	0.06	21.66	-	-	-	-	-
StampOne [1]	28.61	0.81	0.30	0.10	33.63	-	-	-	-	-
DocSafe(M-1)	31.01	0.85	0.03	**0.05**	**16.77**	31.17	**0.90**	**0.03**	**0.05**	**16.77**
DocSafe(M-2)	27.39	0.80	0.06	0.05	35.75	28.42	0.82	0.06	0.05	35.72
DocSafe(M-3)	29.06	0.86	0.09	0.08	22.02	30.68	0.88	0.04	0.04	22.0
RoSteALS [16]	**32.81**	**0.95**	0.04	0.09	13.49	-	-	-	-	-

TABLE 4. Encoded image quality metrics for the object dataset presented in two parts: (A) shows the results when the message is embedded across the full image. (B) presents the results when a spatial mask is applied and the message is hidden only within the masked region. The evaluation was conducted using the object dataset, comprising 4,000 images randomly sampled from the COCO test set.

Methods	(A) no mask					(B) using mask				
	PSNR (⇑)	SSIM (⇑)	LPIPS (⇓)	ColorHisto (⇓)	FID (⇓)	PSNR (⇑)	SSIM (⇑)	LPIPS (⇓)	ColorHisto (⇓)	FID (⇓)
StegaStamp [4]	22.36	0.66	0.20	0.28	15.07		-	-	-	-
Code Face [3]	**25.37**	**0.90**	0.30	0.08	16.7		-	-	-	-
RiemStega [6]	24.61	0.89	**0.04**	0.07	8.66		-	-	-	-
StampOne [1]	24.36	0.81	0.33	0.10	16.23		-	-	-	-
DocSafe(M-1)	23.83	0.80	0.10	0.12	**4.11**	24.21	0.81	0.10	0.08	**4.10**
DocSafe(M-2)	24.12	0.79	**0.04**	**0.05**	13.24	24.32	0.81	**0.03**	**0.05**	13.23
DocSafe(M-3)	24.37	0.84	0.13	0.08	7.17	**24.64**	**0.85**	0.12	0.06	**7.15**
RoSteALS [16]	25.12	**0.90**	**0.04**	0.09	20.28		-	-	-	-

V. EXPERIMENTS

A. DATASETS, TRAINING DETAILS AND METRICS

Datasets: The DocSafe model was trained separately on face and object datasets to optimize domain-specific performance. The face-domain model (M-1) used 51, 990 diverse images from datasets Flickr-Faces-HQ (FFHQ) [38], PICS Face, Color FERET [39], the AT&T Face Database [40], VGGFace2 [41], and the BioID Face Database [42] datasets.

For object-domain training (M-2 and M-3), subsets of COCO [10] and DeepFashion [43]—comprising over 250, 000 images—were used.

Evaluation for digital robustness employed 4, 000 COCO and 2, 000 FFHQ test image datasets. For printer-proof testing, 40 representative images were selected from each COCO and FFHQ datasets, ensuring variation in facial and object characteristics.

Depth monoclonal mask: To generate and apply spatial masks during training, we utilized Depth-Anything-V2 [44] to extract depth-aware regions of interest. Throughout training, we randomly alternated between using these depth-derived masks and applying full-image message embedding, enabling the model to learn effectively under both masked and unmasked conditions. become sure at least mask for hiding message cover 30% of images. Furthermore, to ensure sufficient space for message embedding, we constrained the selected mask to cover at least 30% of the image area.

Training details: During the training process, we used a batch size of 10. The coefficients for the encoder loss function in Equation 9 were set as follows: $\lambda_{Color} = 0.8$, $\lambda_{LPIPS} = 2$, $\lambda_{MSE} = 1$, $\lambda_{Stein} = 0.6$ and $\lambda_{SD} = 1$. For the decoder loss function coefficients in Equation 10, we set $\lambda_{FD} = 1$, $\lambda_{BCE} = 10$, and $\lambda_{AIRM} = 4$. These settings were inspired by StampOne and RiemStega.

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: To evaluate the quality of the encoded output relative to the original cover image, we employ a comprehensive set of metrics. These include the Structural Similarity Index Measure (SSIM) [45] to assess perceptual similarity, Peak Signal-to-Noise Ratio (PSNR) [46], the Learned Perceptual Image Patch Similarity (LPIPS) [19] for deep learning-based perceptual similarity evaluation, and Histogram Color loss (L_{ColorH}) [35] to analyze color distribution consistency. Additionally, to measure the overall quality of image collections within the dataset, we utilize Fréchet Inception Distance (FID) [47].

For evaluating secret message decoding performance, we report the standard metrics Bit Accuracy (BitAcc) and String Accuracy (StrAcc). For BitAcc, a randomly generated bit message is embedded into the original image and then decoded from the encoded image. The decoded message is compared with the original message using binary cross-entropy loss (BCE). StrAcc is defined as the percentage of decoded strings that are perfectly recovered from the encoded images across 10 to 20 frames. Specifically, we first generate a random string (5-8 correct) and convert it into a binary

TABLE 5. Decoder performance - String Accuracy (StrAcc) - for the face dataset is presented in two parts: (A) reports the results when the message is embedded across the entire image. (B) presents the results when a spatial mask is applied, restricting message embedding to specific facial regions. The evaluation was conducted using the face dataset, comprising 40 printed encoded images randomly sampled from the FFHQ dataset.

Methods	(A) no mask				(B) using mask			
	5×5 cm	4×4 cm	3×3 cm	2×2cm	5×5 cm	4×4 cm	3×3 cm	2×2cm
StegaStamp [4]	72	70	65	48	-	-	-	-
Code Face [3]	55	50	38	15	-	-	-	-
RiemStega [6]	100	100	85	25	-	-	-	-
StampOne [1]	100	100	95	62	-	-	-	-
DocSafe(M-1)	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**
DocSafe(M-2)	**100**	**100**	**100**	83	**100**	**100**	**100**	64
DocSafe(M-3)	**100**	**100**	55	45	100	100	75	45
RoSteALS [16]	0	0	0	0	0	0	0	0

TABLE 6. Decoder performance - String Accuracy (StrAcc) - for the object dataset is presented in two parts: (A) reports the results when the message is embedded across the entire image. (B) presents the results when a spatial mask is applied, restricting message embedding to specific object regions. The evaluation was conducted using the object dataset, comprising 40 printed encoded images randomly sampled from the COCO dataset.

Methods	(A) no mask				(B) using mask			
	5×5 cm	4×4 cm	3×3 cm	2×2 cm	5×5 cm	4×4 cm	3×3 cm	2×2 cm
StegaStamp [4]	82	82	82	72	-	-	-	-
RiemStega [6]	100	100	85	22	-	-	-	-
StampOne [1]	**100**	**100**	95	62	-	-	-	-
DocSafe(M-1)	60	25	25	0	22	0	0	0
DocSafe(M-2)	**100**	**100**	**100**	**85**	**100**	**100**	**85**	**68**
DocSafe(M-3)	**100**	**100**	87	60	**100**	**100**	72	45
RoSteALS [16]	0	0	0	0	0	0	0	0

representation using Bose–Chaudhuri–Hocquenghem (BCH) error correction [31]. BCH codes are cyclic error-correcting codes constructed using polynomial algebra over finite fields, enabling reliable detection and correction of multiple random bit errors in digital data. In our experiments, for models with a 256-bit message capacity (including DocSafe and StampOne), BCH encoding is configured with a generator polynomial of 476, a block correct size of 23 bits, and the capability to correct up to 8 erroneous bits per string. For models with 100-bit message capacity, BCH encoding uses a generator polynomial of 137, a block correct size of 8 bits, and supports correction of up to 5 erroneous bits per string.

B. BASELINE COMPARISON

Code Face, StegaStamp and RiemStega 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} bpp.

StampOne embeds a payload of 256 bits, corresponding to a capacity of 0.13×10^{-2} bits per pixel (bpp), into images of resolution $256 \times 256 \times 3$. The variant of StampOne utilizing the Attention-VNet architecture for both the encoder and decoder is selected as the baseline comparison model in our experiments.

DocSafe M-1 refers to the DocSafe model trained specifically on the face dataset. When attempting to train this configuration on more complex datasets, such as COCO, the model fails to converge and is unable to learn meaningful message decoding. M-1 utilizes MEN-1 as its Message Embedding Network. The embedding capacity of this configuration is 0.13×10^{-2} bpp, corresponding to 256 bits embedded in an image of resolution $256 \times 256 \times 3$.

DocSafe M-2 and M-3 are our DocSafe models that use MEN-2 and MEN-3, respectively, as their Message Embedding Networks. These configurations are successfully trained on the COCO object dataset. Both maintain the same embedding capacity as M-1: 0.13×10^{-2} bpp (256 bits) for images of size $256 \times 256 \times 3$.

M-ALL, M-Grad, M-LL, M-HL, M-LH, and M-HH have structure similar to StampOne with the difference that the wavelet sub-bands filters and we also do not apply mask strategy here. These models are only trained with face dataset.

M-AIRM-NoPatches, M-AIRM, M-Stein, and M-Jeffrey share the same architecture as StampOne, with the decoder trained using the AIRM covariance loss function.

The only distinction between these models lies in the covariance loss function applied to the encoder—that is, the loss computed between the original and encoded images. Each variant employs a different formulation: no encoder-side covariance loss (M-AIRM-NoPatches), AIRM (M-AIRM), Stein divergence (M-Stein), and Jeffrey divergence (M-Jeffrey), respectively.

C. FREQUENCY FILTER COMPARISON

Table 1 compares six steganography models using different frequency-pass filters to identify the most effective strategy for frequency processing in the proposed model. M-All, which uses all sub-bands, achieves the best performance and is therefore selected for subsequent experiments. As shown in Figure 3, the filter responses vary depending on the input, which motivated us to initialize the depthwise coefficients adaptively, as described in Section IV-A.

D. COVARIANCE LOSS FUNCTION EVALUATION

Figure 4 illustrates the performance of our base model under different covariance loss functions for the decoder. According to the results, AIRM is the most effective covariance loss for the decoder. In contrast, Table 2, which compares the perceptual metrics of encoded images obtained with different covariance losses for the encoder, indicates that Stein divergence is more suitable for the encoder. This observation highlights the task-specific nature of covariance metrics.

E. IMPACT ON PERCEPTUAL QUALITY

Figures 7 and 8 present examples of encoded images masking and unmasking, respectively. Corresponding quantitative perceptual comparisons on the face and object datasets are reported in Tables 3 and 4, respectively.

For the face dataset without masking as shown in Table 3 (A), RoSteALS achieves the best PSNR (32.81) and SSIM (0.95), while RiemStega obtains the lowest LPIPS (0.02). Although DocSafe M-1 is slightly behind in these metrics, with a PSNR of 31.01, an SSIM of 0.85, and an LPIPS of 0.03, it outperforms all competing methods in ColorHisto (0.05) and FID (16.77), indicating superior color preservation and visual realism. In comparison, RiemStega achieves 0.06 in ColorHisto, and CodeFace reports an FID of 20.77, both of which are inferior to DocSafe M-1. For the face dataset with masking as shown in Table 3(B), performance further improves when spatial masking is applied. Specifically, PSNR increases from 31.01 to 31.17 and SSIM increases from 0.85 to 0.90, while FID and LPIPS remain unchanged at 16.77 and 0.03, respectively, compared with the model without masking.

For the more complex COCO dataset, DocSafe M-1 struggles to converge. To address this limitation, DocSafe M-2 and M-3 were introduced. As shown in Table 4(A), M-1 achieves the best FID score (4.11), while M-2 and M-3 outperform the other methods in terms of ColorHisto (0.05–0.08) and competitive LPIPS values (0.04–0.13). Compared with previous methods, RiemStega and CodeFace also show strong perceptual performance. RiemStega achieves an FID of 8.66 and a ColorHisto score of 0.07, and further obtains the best LPIPS score (0.04). CodeFace achieves the best SSIM score (0.90) and the highest PSNR (25.37). These results indicate that previous methods remain competitive on some perceptual metrics, although our models achieve stronger overall performance across multiple evaluation criteria. As shown in Table 4 (B), the performance of our model improves slightly when masking is applied; however, the gain is less significant than that observed on the face dataset.

Across all datasets, DocSafe models consistently achieve the lowest FID and ColorHisto scores, while remaining competitive on the other evaluation metrics. This highlights their strong generalization ability and effectiveness in preserving the image distribution after message embedding.

F. PRINT PROOF

To evaluate the print-proof capability of the DocSafe framework on both face and object datasets, we randomly selected 40 images from the VGGFace2 dataset and 40 from the COCO test set. Encoded images from each model were printed at four physical sizes—5 × 5, 4 × 4, 3 × 3, and 2 × 2 cm—using a Brother L3270CDW printer.

Printed images were captured using a Samsung S22 Ultra running the IP Webcam app. A custom Python application decoded the message live. String accuracy (StrAcc) was evaluated based on whether a randomly generated string was correctly decoded over 10 to 20 frames.

Decoding results for the face and object datasets are presented in Tables 5 and 6. Notably, as shown in Table 5(A), DocSafe M-1 achieves 100% bit accuracy even at a print size of 2 × 2 cm on face images, demonstrating strong robustness to resolution loss. DocSafe M-2 also achieves 100% bit accuracy at 3 × 3 cm and 83% bit accuracy at 2 × 2 cm, although its performance is slightly lower than that of M-1, particularly on facial data. These results are substantially better than those of previous steganography methods, such as RiemStega (85% StrAcc at 3 × 3 cm) and StampOne (95% StrAcc at 3 × 3 cm). When a mask is used to embed the message only within the facial region, as shown in Table 5(B), excluding the background, M-1 still achieves 100% StrAcc even at the small print size of 2 × 2 cm. This represents one of the most significant results of our model.

On the more complex object dataset (Table 6), DocSafe M-1 does not maintain the high decoding accuracy observed on facial data. To address this limitation, we introduced two enhanced variants, DocSafe M-2 and M-3, which show significantly improved decoding performance across all tested sizes, particularly at smaller resolutions. In particular, DocSafe M-2 reliably decodes messages from prints as small as 3 × 3 cm with 100% StrAcc and from prints as small as 2×2 cm with 85% StrAcc, demonstrating strong robustness in more heterogeneous and visually challenging domains. Compared with StampOne, our M-2 model improves decoding

TABLE 7. Decoder performance, measured in terms of String accuracy (%), is evaluated under JPEG compression, Gaussian noise, and resolution changes using 2,000 face images randomly selected from the FFHQ dataset.

Methods	JPEG (%)			Gaussian (Std 0 to 1)			Resolution (Pixel)		
	70	60	50	0.08	0.06	0.04	(60 × 60)	(80 × 80)	(100 × 100)
StegaStamp [4]	**100**	**100**	**100**	**100**	**100**	**100**	55	80	91
Code Face [3]	80	84	88	55	75	86	2	11	36
RoSteALS [16]	87	90	94	23	35	53	81	97	98
RiemStega [6]	99	**100**	**100**	98	99	**100**	55	**99**	**100**
StampOne	**100**	**100**	**100**	98	**100**	**100**	84	98	**100**
DocSafe (M-1)	80	85	**100**	98	**100**	**100**	**85**	**100**	**100**
DocSafe (M-2)	82	87	**100**	98	98	**100**	80	98	**100**
DocSafe (M-3)	82	87	**100**	90	97	**100**	80	98	**100**

TABLE 8. Decoder performance, measured by String Accuracy (%), is evaluated under contrast and brightness variations using 2,000 face images randomly selected from the FFHQ dataset.

Methods	Contrast (0 to 1)			Brightness (-1 to 1)	
	0.05	0.1	0.15	-1	1
StegaStamp [4]	2.0	39	77	**100**	**100**
Code Face [3]	0	1	30	90	90
RoSteALS [16]	20	67	85	91	95
RiemStega [6]	**100**	96	63	**100**	**100**
StampOne	**100**	**100**	**100**	**100**	**100**
DocSafe (M-1)	**100**	**100**	**100**	**100**	**100**
DocSafe (M-2)	**100**	**100**	**100**	**100**	**100**
DocSafe (M-3)	**100**	**100**	**100**	**100**	**100**

accuracy by 5% at 3×3 cm and by 23% at 2×2 cm on printed encoded images. When a mask is used to embed the message only within the object region, as shown in Table 6(B), M-2 achieves 100% StrAcc at 3×3 cm and 85% StrAcc at 2×2 cm. Under the same setting, M-3 achieves 72% StrAcc at 3×3 cm and 45% StrAcc at 2×2 cm.

CodeFace, originally developed for facial images, does not generalize well to more diverse datasets and was therefore excluded from the object-based evaluations. Likewise, prior robust methods, including StegaStamp, RiemStega, and StampOne, do not support partial image embedding. These methods require the message to be distributed over the entire image, and their decoding fails entirely when spatial masking is applied. While cropping the region of interest, encoding it separately, and reinserting it into the original image may be considered a possible workaround, this solution lacks the integrated consistency provided by our masking strategy. In addition, other steganography methods that employ masking for region-specific embedding, such as WAM [22], MaskWM [48], and PWFN [49], do not demonstrate printability capability and are therefore not directly comparable. By contrast, DocSafe enables seamless mask-based embedding while preserving both spatial and frequency consistency throughout the image.

Finally, although RoSteALS performs well under digital distortions, it fails to decode any messages from printed images, highlighting its lack of print-proof capability.

G. DIGITAL ROBUSTNESS

To evaluate the performance of the decoder under different digital noise conditions, we conducted experiments using various distortion types, including JPEG compression, Gaussian noise, resolution reduction, and variations in contrast and brightness. Decoder effectiveness was measured by calculating the percentage of correctly decoded messages from the encoded images. The results are summarized in Tables 7 and 8, based on experiments conducted on the face dataset.

DocSafe, particularly M-1, demonstrated the strongest overall performance under Gaussian noise, resolution changes, and contrast/brightness variations. Under Gaussian noise with a standard deviation of 0.06, DocSafe, StampOne, and StegaStamp all achieved 100% decoding accuracy, while RiemStega achieved 99%. When the standard deviation increased to 0.08, StegaStamp maintained 100% accuracy, whereas DocSafe M-1 and M-2 followed closely with 98%. For the lower resolution of 60×60 pixels, M-1 achieved the best result at 85%, followed by StampOne at 84%, and M-2 and M-3 at 80%. At 80×80 pixels, M-1 reached 100%, outperforming RiemStega at 99% and M-2/M-3 at 98%. Under contrast and brightness variations, all three DocSafe variants and StampOne achieved perfect decoding accuracy of 100%.

However, under JPEG compression, StampOne and RiemStega demonstrated greater robustness. At 70% JPEG quality, these methods achieved decoding accuracies in the range of 90%–100%, whereas the DocSafe models achieved 80%–82%. At 60% JPEG quality, our models performed more competitively, reaching 85%–87%, which is closer to the performance of StampOne and RiemStega. Under 90% JPEG quality, our models were able to decode the message completely.

VI. CONCLUSION

DocSafe achieves two key advancements. First, it enables spatially selective message embedding using flexible, non-rectangular masks, making it well-suited for applications like secure ID stamps and brand protection. Second, it provides insights into the role of frequency components and

covariance alignment in enhancing robustness. By integrating frequency-aware processing and covariance-based loss, DocSafe improves retrieval accuracy and generalization under diverse distortions.

Two key directions emerge for future research: First, our analysis in Section III-A suggests that the optimal contribution of each frequency sub-band may vary by input type (e.g., original, message, encoded), yet we used fixed filtering across all inputs. Exploring adaptive, input-dependent frequency selection could enhance performance. Second, while our evaluation of covariance-based losses yielded useful insights, these findings may not generalize to other architectures, highlighting the need for architecture-specific optimization in future studies.

ACKNOWLEDGMENT

This research was supported by FCT - Fundação para a Ciência e a Tenologia, I.P., under the projects UIDB/00048/2020[1] and the projects CertifyMeAP (2024.07457.IACDC)[2] and BlockDFake (2024.07681.IACDC)[3] by the joint call of FCT - Fundação para a Ciência e Tecnologia, I.P. and the Plano de Recuperação e Resiliência - PRR from Estrutura de Missão Recuperar Portugal (EMRP). The authors acknowledge the support of Instituto de Sistemas e Robótica - Pólo de Coimbra (ISR) and the University of Coimbra for providing computational resources and infrastructure. Parts of the text, including its grammar and wording, have been improved using artificial intelligence tools.

REFERENCES

[1] F. Shadmand, I. Medvedev, L. Schirmer, J. Marcos, and N. Gonçalves, "StampOne: Addressing frequency balance in printer-proof steganography," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2024, pp. 4367–4376.

[2] T. Bui, S. Agarwal, and J. Collomosse, "Trustmark: Universal watermarking for arbitrary resolution images," 2023, *arXiv:2311.18297*.

[3] F. Shadmand, I. Medvedev, and N. Gonçalves, "CodeFace: A deep learning printer-proof steganography for face portraits," *IEEE Access*, vol. 9, pp. 167282–167291, 2021.

[4] M. Tancik, B. Mildenhall, and R. Ng, "StegaStamp: Invisible hyperlinks in physical photographs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 2114–2123.

[5] J. Chen, X. Liu, S. Liang, X. Jia, and Y. Xun, "Universal watermark vaccine: Universal adversarial perturbations for watermark protection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2023, pp. 2321–2328.

[6] A. Cruz, G. Schardong, L. Schirmer, J. Marcos, F. Shadmand, and N. Gonçalves, "RiemStega: Covariance-based loss for print-proof transmission of data in images," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Feb. 2025, pp. 7572–7581.

[7] R. Durall, M. Keuper, and J. Keuper, "Watch your up-convolution: CNN based generative deep neural networks are failing to reproduce spectral distributions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 7887–7896.

[8] M. Barni, F. Bartolini, and A. Piva, "Improved wavelet-based watermarking through pixel-wise masking," *IEEE Trans. Image Process.*, vol. 10, no. 5, pp. 783–791, May 2001.

[9] P. Morerio and V. Murino, "Correlation alignment by Riemannian metric for domain adaptation," 2017, *arXiv:1705.08180*.

[10] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.

[11] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and S. Nong, "BiSeNet: Bilateral segmentation network for real-time semantic segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 325–341.

[12] T. Pevný, T. Filler, and P. Bas, "Using high-dimensional image models to perform highly undetectable steganography," in *Proc. Int. Workshop Inf. Hiding*, 2010, pp. 161–177.

[13] V. Holub, J. Fridrich, and T. Denemark, "Universal distortion function for steganography in an arbitrary domain," *EURASIP J. Inf. Secur.*, vol. 2014, no. 1, pp. 1–13, Dec. 2014.

[14] V. Holub and J. Fridrich, "Designing steganographic distortion using directional filters," in *Proc. IEEE Int. Workshop Inf. Forensics Security (WIFS)*, Dec. 2012, pp. 234–239.

[15] M. Urvoy, D. Goudia, and F. Autrusseau, "Perceptual DFT watermarking with improved detection and robustness to geometrical distortions," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 7, pp. 1108–1119, Jul. 2014.

[16] T. Bui, S. Agarwal, N. Yu, and J. Collomosse, "RoSteALS: Robust steganography using autoencoder latent space," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2023, pp. 933–942.

[17] S. Indah Desti and A. A. Hamdan, "Forgery detection and authentication of medical images using adaptive bit substitution watermarking," *Int. J. Adv. Comput. Informat.*, vol. 2, no. 2, pp. 139–151, Feb. 2026.

[18] J. Zhu, R. Kaplan, J. Johnson, and L. Fei-Fei, "HiDDeN: Hiding data with deep networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 682–697.

[19] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 586–595.

[20] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 815–823.

[21] D. Hendrycks and T. Dietterich, "Benchmarking neural network robustness to common corruptions and perturbations," 2019, *arXiv:1903.12261*.

[22] T. Sander, P. Fernandez, A. Durmus, T. Furon, and M. Douze, "Watermark anything with localized messages," 2024, *arXiv:2411.07231*.

[23] K. Dzhanashia and O. Evsutin, "Robust image watermarking for diverse channels with template-forming neural network," *Appl. Soft Comput.*, vol. 186, Jan. 2026, Art. no. 114125.

[24] S. B. Lakasek and H. Elhadi, "A robust image watermarking technique using NSST-HD-SVD," *Int. J. Adv. Comput. Informat.*, vol. 2, no. 2, pp. 115–126, Feb. 2026.

[25] J. Quionero-Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence, *Dataset Shift in Machine Learning*. Cambridge, MA, USA: MIT Press, 2009.

[26] G. Wilson and D. J. Cook, "A survey of unsupervised deep domain adaptation," *ACM Trans. Intell. Syst. Technol.*, vol. 11, no. 5, pp. 1–46, Oct. 2020.

[27] Y. Zhang, T. Liu, M. Long, and M. I. Jordan, "Bridging theory and algorithm for domain adaptation," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 7404–7413.

[28] J. Liang, D. Hu, and J. Feng, "Do we really need to access the source data? Source hypothesis transfer for unsupervised domain adaptation," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 6028–6039.

[29] X. Chen, S. Wang, M. Long, and J. Wang, "Transferability vs. Discriminability: Batch spectral penalization for adversarial domain adaptation," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 1081–1090.

[30] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16×16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.

[31] G. D. Forney, "On decoding BCH codes," *IEEE Trans. Inf. Theory*, vol. IT-11, no. 4, pp. 549–557, Apr. 1965.

[32] O. Oktay, J. Schlemper, L. Le Folgoc, M. Lee, M. Heinrich, K. Misawa, K. Mori, S. McDonagh, N. Y. Hammerla, B. Kainz, B. Glocker, and D. Rueckert, "Attention U-Net: Learning where to look for the pancreas," 2018, *arXiv:1804.03999*.

[1] https://doi.org/10.54499/UID/00048/2025
[2] https://doi.org/10.54499/2024.07457.IACDC
[3] https://doi.org/10.54499/2024.07681.IACDC

[33] M. Afifi, B. Price, S. Cohen, and M. S. Brown, "When color constancy goes wrong: Correcting improperly white-balanced images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1535–1544.

[34] J. T. Barron, "Convolutional color constancy," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 379–387.

[35] M. Afifi, M. A. Brubaker, and M. S. Brown, "HistoGAN: Controlling colors of GAN-generated and real images via color histograms," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 7937–7946.

[36] E. Wengrowski and K. Dana, "Light field messaging with deep photographic steganography," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1515–1524.

[37] T. Cunha, L. Schirmer, J. Marcos, and N. Gonçalves, "Noise simulation for the improvement of training deep neural network for printer-proof steganography," in *Proc. 13th Int. Conf. Pattern Recognit. Appl. Methods*, 2024, pp. 179–186.

[38] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4396–4405.

[39] P. J. Grother, P. J. Grother, M. Ngan, and K. Hanaoka, "Face recognition vendor test (FRVT)," Dept. U.S. Dept. of Commerce, Nat. Inst. of Standards and Technol., Gaithersburg, MD, USA, Tech. Rep. NISTIR 8009, 2014.

[40] (2021). *The Database of Faces (AT&T)*. [Online]. Available: https://git-disl.github.io/GTDLBench/datasets/attfacedataset

[41] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, "VGGFace2: A dataset for recognising faces across pose and age," in *Proc. 13th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, May 2018, pp. 67–74.

[42] O. Jesorsky, K. Kirchberg, and R. Frischholz, "Robust face detection using the Hausdorff distance," in *Proc. Audio-Video-Based Biometric Person Authentication, 3rd Int. Conf.*, AVBPA 2001 Halmstad, Sweden, 2001, pp. 90–95.

[43] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang, "DeepFashion: Powering robust clothes recognition and retrieval with rich annotations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1096–1104.

[44] L. Yang, B. Kang, Z. Huang, Z. Zhao, X. Xu, J. Feng, and H. Zhao, "Depth anything v2," 2024, *arXiv:2406.09414*.

[45] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[46] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd ed., London, U.K.: Pearson, 2008.

[47] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs trained by a two time-scale update rule converge to a local Nash equilibrium," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 30, 2017, pp. 6626–6637.

[48] R. Hu, J. Zhang, S. Zhao, N. Lukas, J. Li, Q. Guo, H. Qiu, and T. Zhang, "Mask image watermarking," in *Proc. Adv. Neural Inf. Process. Syst.*, 2025.

[49] S. Xie, C. Zhao, N. Sun, W. Li, and H. Ling, "Picking watermarks from noise (PWFN): An improved robust watermarking model against intensive distortions," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2024, pp. 1–6.

FARHAD SHADMAND received the master's degree in advanced statistical physics in 2018. He is currently pursuing the Ph.D. degree. He is a Researcher with the University of Coimbra, with a research focus on deep learning-based face image processing and steganography. Since 2019, he has been actively involved in computer vision and machine learning. For more information, please visit his LinkedIn profile: www.linkedin.com/in/farhadsh1992.

IURII MEDVEDEV received the Ph.D. degree in electrical and computer engineering from the Faculty of Sciences and Technology, University of Coimbra. He is currently a Researcher with the Visual Information Security Team, Institute of Systems and Robotics, University of Coimbra. His research interests include deep learning–based computer vision, with a focus on face recognition, biometric security, and presentation attack detection.

LUIZ SCHIRMER is currently a Researcher and a Professor with Universidade Federal de Santa Maria (UFSM). Additionally, he was a Postdoctoral Researcher with the Vision and Computer Graphics Laboratory (Visgraf), IMPA, from 2021 to 2023. Previously, he was a Postdoctoral Researcher with the Institute of Systems and Robotics, University of Coimbra, where he was a Member of the Visual Information Security Team (VisTeam). His research interests include multilinear (tensor) algebra and deep learning for computer vision. He is also very interested in image processing, computer graphics, and machine learning.

NUNO GONÇALVES (Member, IEEE) received the M.Sc. and Ph.D. degrees from the University of Coimbra, in 2002 and 2008, respectively. He is a Researcher with the Institute of Systems and Robotics, University of Coimbra, and a Tenured Assistant Professor with the Department of Electrical and Computers Engineering, University of Coimbra. Since 2018, he has been the Innovation Manager with INCM. He has scientific publications in the following topics: visual coding (machine-readable codes), steganography, object recognition, facial recognition and diagnosis, biometrics, documents security, augmented and virtual reality, reflections for image rendering, light field cameras, omnidirectional vision, non-central cameras, calibration, optics, camera models, motion estimation, pose estimation, web information systems, sports vision, and legged robotics. He was the principal investigator of a closed project, funded by Portuguese Science and Technology Foundation, in non-central camera models for computer graphics and computer-aided surgery. He is currently a Scientific Coordinator of several projects with the industry (funded by the INCM—Portuguese Mint and Official Printing Office) in the area of security elements involving biometrics, machine readable codes (with some applications in virtual and augmented reality), security unique marks in printing labels, security unique marks in assay contrasts in artifacts of precious metals, and processing of human faces in 3-D by using several types of cameras. He is inventor of six patents. His main research interests include computer vision, biometrics, machine-readable codes, security printing, computer graphics and machine learning, with special emphasis to biometrics and steganography for ID and travel documents.