SWIFT: Semantic Watermarking for Image Forgery Thwarting

Gautier Evennou, Vivien Chappelier, Ewa Kijak, Teddy Furon

▶ **To cite this version:**

Gautier Evennou, Vivien Chappelier, Ewa Kijak, Teddy Furon. SWIFT: Semantic Watermarking for Image Forgery Thwarting. WIFS 2024 - 16th IEEE International Workshop on Information Forensics and Security, IEEE, Dec 2024, Roma, Italy. pp.1-6. hal-04728070

HAL Id: hal-04728070
https://hal.science/hal-04728070v1

Submitted on 9 Oct 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution 4.0 International License

SWIFT: Semantic Watermarking for Image Forgery Thwarting

Gautier Evennou[1,2], Vivien Chappelier[2], Ewa Kijak[1], Teddy Furon[1]

[1]*IRISA, Univ. Rennes, Inria, CNRS* [2]*Imatag*

***Abstract*—This paper proposes a novel approach towards image authentication and tampering detection by using watermarking as a communication channel for semantic information. We modify the HiDDeN deep-learning watermarking architecture to embed and extract high-dimensional real vectors representing image captions. Our method improves significantly robustness on both malign and benign edits. We also introduce a local confidence metric correlated with Message Recovery Rate, enhancing the method's practical applicability. This approach bridges the gap between traditional watermarking and passive forensic methods, offering a robust solution for image integrity verification. The code is available at https://github.com/gautierevn/swift_watermarking.**

***Index Terms*—Watermarking, Image authentication, Semantic information**

I. Introduction

Many technical means can verify the authenticity of multimedia content. This ranges from a digital signature stored in the metadata like in the recent C2PA (Coalition for Content Provenance and Authenticity) and IPTC (International Press Telecommunications Council) initiatives, to passive forensics [1], [2] and active fragile watermarking [3], [4]. The main difficulty resides in making a clear cut between benign processing which are common editing in the entertainment industry and malicious transformations which modify on purpose the content. Semi-fragile watermarking faces this challenge: it should be robust to benign processing but fragile to deeper transformations. At the decoding side, its absence reveals that the piece of content has been modified beyond the accepted limit. This limit between benign and malicious editing is not easy to be defined in mathematical terms, although in real life the difference is straightforward: Any modification of the semantics is malicious.

This paper investigates the idea of hiding semantics information within the cover work in an imperceptible and robust manner. The verification amounts to compare the semantics of the content with the decoded information. To the best of our knowledge, embedding its own semantic into the content itself to ensure integrity is an unexplored research path. A first challenge lies in the poor capacity of robust image watermarking. Multi-bit watermarking embeds messages into images but typically only supports up to 64 bits of data transmission. Higher capacity schemes exist but with a much lower robustness. The second challenge is the representation of the semantic of an image whose definition is still a matter of debate. We chose the textual description of the image given by an automatic captioning as the message to be hidden.

The scenario establishes a covert channel between two entities: Alice, the sender who authenticates the original work, and Bob, the recipient tasked with verifying its authenticity. The cover work may undergo modifications by a third party, referred to as Eve, acting as an intermediary. Eve's alterations may be intentional, involving semantic edits, or unintentional, comprising benign changes. The crux of our method lies in Bob's ability to recover the message embedded by Alice. This recovery enables Bob to assess whether Eve's modifications have introduced semantically misleading alterations to the original content. By comparing the recovered message with the received work, Bob makes informed decisions about the nature and extent of any change. The robustness of the communication channel despite potential interferences is key.

To this end, we propose to increase the utility, re-usability and flexibility by disentangling the watermarking layer from the encoding layer. The watermarking layer is responsible for hiding a high-dimensional real-valued unit-norm vector in the cover while optimizing robustness to various transforms and the watermark imperceptibility. The encoding layer is responsible for encoding a message as a signal to be transmitted on this noisy communication channel. The decoding layer then retrieves the message with some confidence level.

Our framework SWIFT, Semantic Watermarking for Image Forgery Thwarting, provides a robust mechanism for authentication and content verification in scenarios where the integrity of digital media may be compromised between creation and reception. This paper introduces three contributions:

- Hide-$\mathbb{R}$: Inspired by HiDDeN [5], we propose an encoder-decoder network architecture jointly trained to embed and extract high-dimensional unit-norm vectors in images.
- Encoding layer: It encodes a variable-length binary message into a vector to be hidden in images.
- Caption Compression: We finetune a large language model for captioning and combine it with an arithmetic codec to compress the payload as in LLMZip [6].

Three major features stem from the combination of these contributions into the SWIFT scheme:

- Reliability: A confidence metric on the decoded caption gives an informed decision-making about authenticity.
- Security: The design guarantees security via a secret key.

- Performance: SWIFT achieves state-of-the-art results across various benign and malicious transforms, demonstrating its robustness in challenging scenarios.

II. Related Work

a) Image forensics: Passive methods detect alterations of a piece of content, possibly malicious ones. They utilize noise residuals or high-frequency features as input to highlight manipulation traces. These methods are limited to providing localized insights into *specific* alterations. For instance, copy-move forgery detection uses Siamese networks [7] while splicing detection leverages two-stream architectures [8], [9]. Inpainting detection methods have focused on traces left by some deep inpainting models [10], [11]. Forensics methods lack the capacity to offer a global perspective due to the absence of contextual information from the original image.

b) Image watermarking: Traditional watermarking schemes embeds invisible marks within multimedia content to assert copyright ownership (robust watermarking) or authenticate content (fragile watermarking). Classic techniques involve manipulating spatial or frequency domains representation of the media [12], [13]. Recently, deep-learning enabled more robustness as first shown with the encoder-decoder HiDDeN architecture [5] and followed with [14], [15]. SSL [16] embeds a binary message in the latent space of a foundation model learned with supervised learning with low perceptibility but high inference cost due to its iterative nature. TrustMark [17] leverages a more classic encoder-decoder architecture and a GAN loss to learn how to embed binary messages. Note that the payload of a watermarking scheme is always fixed in the literature. One of our contributions is to tackle variable-length messages.

Watermarking can be used for authenticity verification, but it usually uses a fragile or semi-fragile signal whose absence reveals tampering [3], [4]. One exception is the idea of embedding a compressed representation of the image in itself with robust watermarking [18]. At the detection stage, the verifier finds back a copy of the original image to be compared with the image. Our work is similar in spirit except that we embed the semantic textual description of the original image.

III. Method

This section presents the design of the encoding and watermarking layers. We break down the encoding layer into two primary components: the message layer and the modulation layer. Fig. 1 depicts our method.

A. The message layer

Alice wants to transmit a message M to Bob so that he can assess the integrity of the cover image. Alice uses an image captioning model like BLIP2 [19] to generate the caption m of the cover. Alice uses arithmetic coding [20] for losslessly compressing m into M to reduce the number of bits. As in LLMZip [6], Alice takes advantage of a LLM to model the distribution of the messages and improve the compression. She uses OPT-125m [21] finetuned on BLIP2 captions from $2,000$ MSCOCO validation set images. This acts as oracle and gives the probability of each caption symbol used by the arithmetic coding [22]. Fig. 2 shows that finetuning OPT on BLIP2 reduce the mean capacity needed from 75 to 45 bits.

B. The watermarking layer

Modern watermarking leverages deep-learning to learn end to end how to embed a message into a cover image. It enables robustness against benign edits by performing augmentations between the watermark embedding and watermark extraction stages [23]. The most famous example is HiDDeN [5] based on two convolutional neural networks (CNN) jointly trained to embed and extract a fixed-length binary message.

Fig. 3 depicts our Hide-$\mathbb{R}$ architecture, resulting from several modifications of HiDDeN. The input data works with unit vectors in high-dimensional real space instead of binary messages. Specifically, we draw random samples X uniformly distributed on the surface of the unit hypersphere in $\mathbb{R}^D$ with $D = 256$:

$$X = \frac{Z}{\|Z\|} \quad \text{with} \quad Z \sim \mathcal{N}(0, I_D). \tag{1}$$

We extend the number of channels in the convolutional layers to $1.5D$ instead of the fixed 64 to account for higher dimension D than the message length $L = 30$ proposed in the original paper. Signal X is concatenated with the cover image I_{co} along the channel dimension before the first convolutional layer. Instead of using a discriminator, we opt for a fixed PSNR budget which both enforce imperceptibility in a flexible way and speed up the learning process. The training minimizes the reconstruction loss $\|X - Y\|$ between X and the reconstructed unit vector Y.

This framework gives a zero-bit watermarking system. Assume X_0 is drawn according to (1) from a pseudo-random generator seeded by the secret key K associated with a fixed index $M = 0$. A watermarking signal is deemed present if the cosine similarity $C_0 = Y^\top X_0$ is above a threshold c. Under the hypothesis $\mathcal{H}_0$, the cover is not watermarked or watermarked with another secret key K'. Then, the p-value is defined as the probability of having higher cosine similarity C_0 than the threshold c and given by:

$$\rho_0(c) = P(C_0 \geq c) = \begin{cases} \frac{1}{2} I_{1-c^2}\left(\frac{D-1}{2}, \frac{1}{2}\right), & \text{if } c > 0 \\ \frac{1}{2} I_{c^2}\left(\frac{1}{2}, \frac{D-1}{2}\right), & \text{otherwise,} \end{cases} \tag{2}$$

where I is the regularized incomplete beta function. This illustrates how a confidence value is available to Bob at the watermark extractor.

C. The modulation layer

1) Multi-bit watermarking with confidence: One way to use zero-bit watermarking to send a N-bit message M is to share 2^N different secret keys, each associated with one possible message. Then Alice selects the key K corresponding to the message to send, and Bob runs the watermark extractor with all the 2^N possible keys. Bob selects the decoded message as:

$$\hat{M} = \underset{m \in [\![0..2^N-1]\!]}{\operatorname{argmax}} (C_m) \tag{3}$$

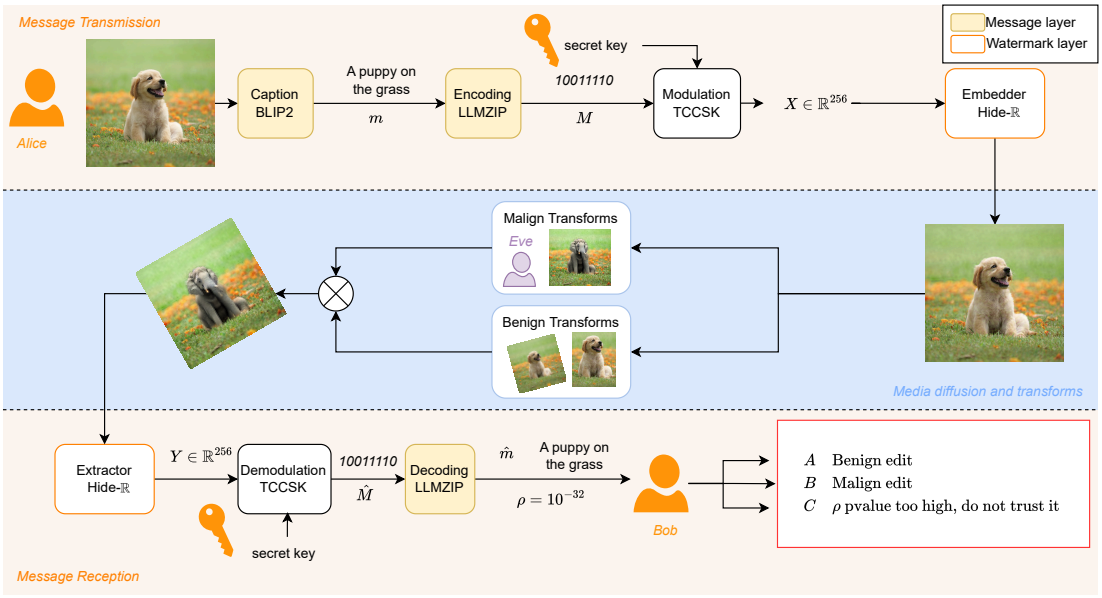

Fig. 1. Overview of SWIFT. To ensure the integrity of an image, Alice first leverages the message layer to tailor a semantic representation of the image, in our case a caption compressed in a lossless fashion. The resulting bit stream is fed to the TCCSK modulation layer thus enabling security and confidence (see Sect. IV-C), and then to the watermarking layer based on our Hide-$\mathbb{R}$ encoder-decoder neural network. At reception, Bob executes the inverse process with a secret key and obtains both the caption and the p-value ρ. If ρ is low enough, Bob can entrust the decoded caption and compare the image he received with the caption, enabling comparison between a proxy of the original image and the received one.

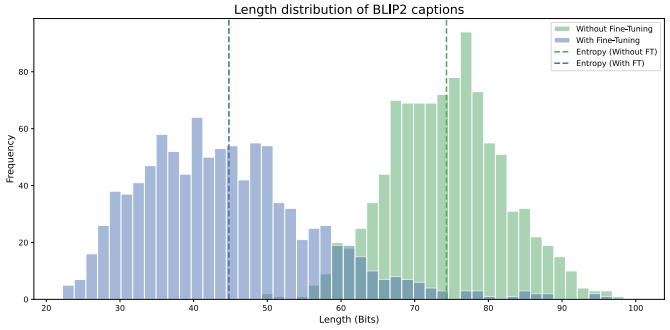

Fig. 2. Length distribution of BLIP2 captions encoded by OPT-125m version. We show that the finetuned version leads to entropy reduction and thus is more efficient to encode captions.

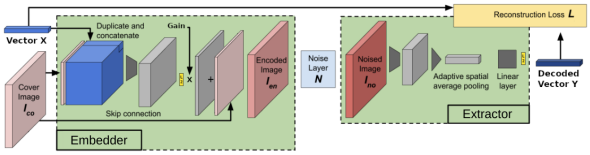

Fig. 3. Hide-$\mathbb{R}$ architecture. We use a L2 norm to control the watermark power to enforce a target PSNR on both the watermark signal in the embedder and on the decoded vector in the extractor.

The p-value under $\mathcal{H}_0$ of decoding this specific message $\hat{M}$ by chance in a non-watermarked image is given by:

$$\rho_1(c) = 1 - (1 - \rho_0(c))^{2^N} \lesssim 2^N \rho_0(c) \text{ if } 2^N \rho_0(c) \ll 1 \quad (4)$$

Equation (4) reveals a trade-off between minimizing $\rho_1(c)$ and increasing the length N of the message: If $N > -\log_2(\rho_0(c))$, it is likely that the decoded message cannot be trusted by Bob. This gives an approximation of the maximum quantity of information that may be transmitted.

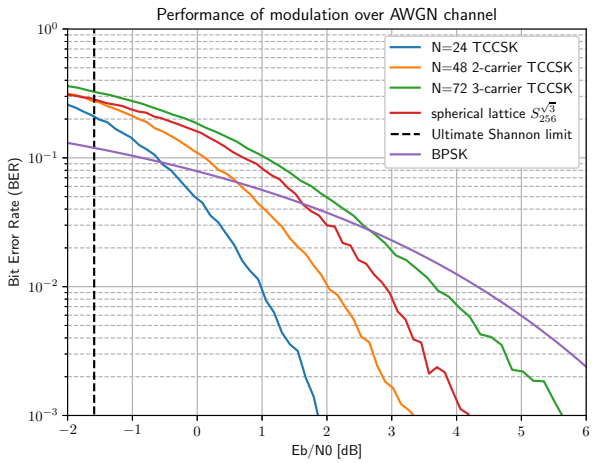

Fig. 4. Performance of different encoding schemes under additive white Gaussian noise. $S_{256}^{\sqrt{3}}$ is the spherical lattice from [24], capable of encoding $22108160 \simeq 2^{24.4}$ messages. TCCSK is most suited for our use case as it can cope with variable length messages.

2) Modulating variable-rate signals: In the previous section, Bob needs to test 2^N keys which becomes intractable as N grows. An alternative is to generate multiple carriers from the same pseudo-random generator seeded by secret key K,

and to combine them to produce $X \in \mathbb{R}^D$.

This paper uses truncated cyclic shift keying (TCCSK) modulation [25] for its improved performance in the additive white Gaussian noise (AWGN) channel compared to Binary Phase Shift Keying (BPSK) (see Fig. 4). The message M is split into T equal blocks of length L, padding the last block with 0 if necessary. Then the j-th block is encoded from a carrier $Z_j \in \mathbb{R}^{2^L} \sim \mathcal{N}(0, I_{2^L}), j \in [\![0..T-1]\!]$ by cyclically shifting it by a number equal to the value represented by the L-bit sub-message $M_j := M_{[\![jL..(j+1)L]\!]}$. The carriers are truncated to dimension D, normalized, summed, and normalized again for transmission:

$$X = \frac{Z}{\|Z\|} \quad \text{with} \quad Z = \sum_{j=0}^{T-1} \frac{Z_j^{M_j}}{\|Z_j^{M_j}\|}, \tag{5}$$

where Z^a denotes a cyclic shift of Z by a followed by truncation to the first D dimensions.

Decoding is achieved by recovering each sub-message from:

$$\hat{M}_j = \underset{k \in [\![0..2^L-1]\!]}{\operatorname{argmax}} \; (Y^\top \frac{Z_j^k}{\|Z_j^k\|}). \tag{6}$$

The p-value ρ is then obtained from Fisher's combined probability test [26] on evaluations of (4) for each submessage:

$$\rho = 1 - \gamma(T, -\sum_{j=0}^{T-1} \log(\rho_1(C_j))), \text{ with } C_j = Y^\top \frac{Z_j^{\hat{M}_j}}{\|Z_j^{\hat{M}_j}\|}, \tag{7}$$

where γ is the lower incomplete gamma function.

Although we want to set L to N, testing all 2^L cyclic shifts becomes untractable as L grows. Proceeding by blocks tackles the variable-length of the message. Yet, some capacity is lost due to padding unless the message length N is a multiple of L.

Finally, to illustrate the flexibility of the approach, we also compare TCCSK with lattice-based modulation on the spherical lattice used in [24]. Although this method is very fast and better than BPSK for the 24-bit case, its efficiency is lower than TCCSK with a single carrier (see Fig. 4).

D. Security

Our approach adheres to Kerckhoffs's principle, relying solely on the shared secret key K between Alice and Bob for security. We assume Eve has full knowledge of the system, except for K. In content authentication, the attacker's primary goal is to forge watermarked content without K (spoofing attack), rather than removing existing watermarks. Each system use reveals at most D out of 2^L carrier values, reducing the urgency for key rotation if usage is limited. As a symmetric system, both Alice and Bob can produce forgeries using K. Thus, mutual trust between them is assumed, with Eve being the only untrusted party in our threat model.

IV. Experiments & Results

This section introduces evaluation of SWIFT and recent watermarking methods for the task of message recovery.

A. Evaluation

Metrics. The benchmark compares watermarking methods by their Message Recovery Rate (MRR), which is defined as the rate of messages being perfectly transmitted without any modifications, over a test set of watermarked images $\mathcal{I}_t$. Let m_i be the original caption and $\hat{m}_i$ the corresponding recovered caption for an image $I_i \in \mathcal{I}_t$. The MRR is defined as follows:

$$MRR = \frac{1}{|\mathcal{I}_t|} \sum_{i=1}^{|\mathcal{I}_t|} \delta(m_i, \hat{m}_i) \tag{8}$$

We chose this metric to ensure practicability and accurately assess the robustness of a system. A watermarking system designed for our task should strive to reach 100% MRR, especially when no confidence metric is available at the decoding step, which is the case of all systems but SWIFT.

Test set. Our test set $\mathcal{I}_t$ is composed of 20,220 images. We use the Emu Edit test set [27] which comprises 2,022 images from MSCOCO [28] and editing instructions for Image Editing models from 8 classes (*local, add, remove, global, text, background, style, color*). For each image, we perform 6 benign and 1 malign transformations with 4 variations in classifier-free guidance. Benign ones are chosen to be realistic in a web setting, or quite important distortion-wise but without semantic alterations: crop 40% of image surface, random noise, grayscale conversion, resize to 128×128, jpeg compression with quality coefficient at 50. Malign ones are images edited by a diffusion model according to Emu Edit instructions, supposed to change the meaning of the cover.

B. Comparison with state of the art

Table I shows the results of state-of-the-art methods SSL and TrustMark against SWIFT: we observe superior resilience to malign transforms while maintaining state-of-the-art performance on benign modifications with significant improvement on resize and grayscale transforms due to our training. We provide another version of SWIFT to watermark at 42db which performs better than TrustMark(Base) on almost all settings.

C. Confidence metric

After the TCCSK modulation, a message M is encoded into a vector X on the 256-d hypersphere. Given an encoded message X, due to transforms during transmission, Y is the noisy extracted vector. At inference, Y is decoded by the TCCSK demodulation into $\hat{M}$. X is unknown, but let $\hat{X}$ be the perfect encoding of $\hat{M}$ and C the cosine similarity computed between $\hat{X}$ and Y (see Fig. 5). We define our pratical confidence metric as ρ (7).

We assume that Y close to the perfect representation $\hat{X}$ of the decoded message $\hat{M}$ means that Y is also close to the (unknown) originally embedded vector X. Thus, a low ρ would entail a low number of errors at the decoding step, as confirmed by the Pearson correlation coefficient of -0.89 computed between $8,000$ ρ values and corresponding MRR. Tab. II shows the MRR on watermarked images under several scenarios: watermarked images, benign attacks and malign

TABLE I

MESSAGE RECOVERY RATE BY TRANSFORM TYPE. ALL THE METHODS SHARE THE SAME MESSAGES FROM THE LLMZIP AS INPUTS. WE PROVIDE RESULTS FOR SWIFT AT 2 DIFFERENT TARGET PSNR FOR FAIR COMPARISON WITH EXISTING METHODS.

Transform	Message Recovery Rate(%) ↑				
	SWIFT	SWIFT	SSL	TrustMark(Q)	TrustMark(Base)
PSNR(db)	40	42	40	42.5	41.6
global	**63.7**	45.1	0.45	0.0	39.0
text	**65.3**	45.9	0.38	0.0	40.0
style	**64.8**	45.3	1.32	0.0	38.3
local	**65.9**	48.2	0.39	0.0	40.2
background	**66.5**	48.3	0.75	0.0	43.9
color	**65.3**	46.9	1.53	0.0	44.6
remove	**67.2**	50.0	0.0	0.0	48.0
add	**69.5**	49.2	0.0	0.0	43.7
crop 40%	82.4	76.7	50.1	86.4	**91.9**
noisy	**75.5**	65.4	0.0	11.8	17.2
resize 128	**95.1**	91.3	0.00	0.0	0.0
grayscale	**95.8**	93.8	2.82	78.0	91.2
jpeg@50	**96.2**	93.5	1.5	82.1	93.2
original	**96.3**	95.1	92.6	94.6	95.7

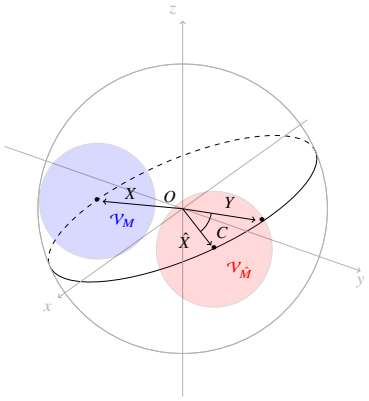

Fig. 5. Representation of the message space after modulation. The circles $\mathcal{V}_M, \mathcal{V}_{\hat{M}}$ illustrate Voronoï cells mapping to different binary messages on the surface of the hyper-sphere. X is associated to the message to be hidden M. The extraction retrieves Y, which is decoded into $\hat{M}$, while $\hat{X}$ results from the modulation of $\hat{M}$, accounting for the perfect representation of $\hat{M}$. The similarity between Y and $\hat{X}$ is given by C. This provides a confidence score for the decoding as explained in Sect. IV-C. The given example illustrates a failure case with a wrong decoded message and a low confidence reflected by a high value of ρ.

attacks, and three ρ-thresholds. Only messages with $\rho < \rho$-threshold should be trusted. An adequate ρ-threshold ensures the perfect extraction, meaning 100% MRR, of the watermarked caption. Note that ρ also refers to the error probability on non-watermarked images: the higher it is, the more a non-watermarked image could be flagged as watermarked.

A confidence metric could also be used in a multi-bit setting with fixed-length code of 64 bits. Padding a n-bit message with $64 - n$ equiprobable random bits drawn from a synchronous source shared by Alice and Bob (e.g. via a secret key K) allows to carry confidence information, at the expense of capacity. Indeed, Bob can check the padding bits and discard any message not matching the expected sequence. In this case, assuming the multi-bit system outputs random codes uniformly under $\mathcal{H}_0$, there is still a 2^{-64+n} probability of ending up with the expected sequence by chance, giving $\rho \geq 2^{-64} \simeq 5e-20$. Our method benefits from comparable confidence, with $\rho = 1.4e-16$ to be protected against all attacks, along with a much greater capacity.

TABLE II

MESSAGE RECOVERY RATE (MRR) WITH SWIFT (40DB) ON EMU EDIT FOR DIFFERENT ρ-THRESHOLDS. ONLY MESSAGES WITH $\rho < \rho$-THRESHOLD ARE DECODED. THEIR PERCENTAGE IS GIVEN BY THE CUMULATIVE DISTRIBUTION FUNCTION (CDF).

ρ-threshold	Scenario	MRR	CDF	Confidence
1.0	Watermarked	96.3	100	None
	Benign Attacks	89.0	100	
	Malign Attacks	66.1	100	
4.2e-13	Watermarked	100	82.6	Low
	Benign Attacks	96.2	60.1	
	Malign Attacks	94.1	31.1	
2.3e-15	Watermarked	100	72	Medium
	Benign Attacks	100	48.3	
	Malign Attacks	99.3	20.2	
1.4e-16	Watermarked	100	66.2	High
	Benign Attacks	100	42.3	
	Malign Attacks	100	18.1	

D. Qualitative results

In Fig. 6 we present samples of watermarked images from Emu Edit. In the first column we see an original image and its watermarked version : the watermark is visible in the bottom left on the plate. The second column depicts our resilience against heavy jpeg compression and the two last ones demonstrates SWIFT behaviour against malign edits. The background edit leaves the foreground intact and thus does not destroy the watermark. In the last column, the local edit remove watermarked regions and lead to a low confidence score, suggesting the image is altered and cannot be trusted.

E. Limitations

We believe this work introduces to a new way of asserting integrity of images. By disentangling watermarking and encoding layers, we expose two research directions : better

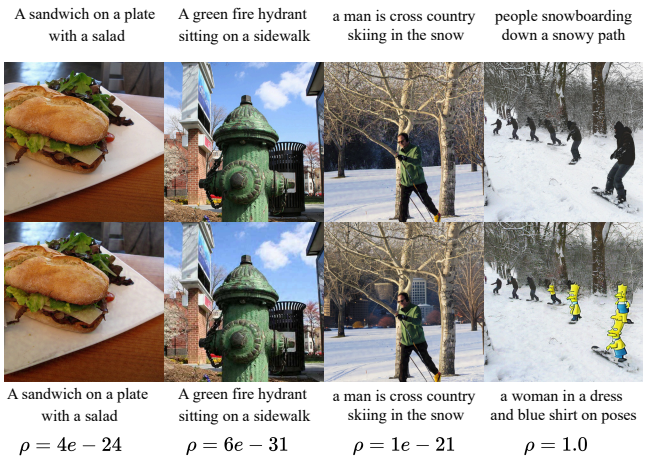

Fig. 6. Examples of recovered messages under different transforms. From top to bottom: message, cover image, watermarked image with or without transforms, retrieved message and confidence metric (the lower, the better). From left to right: no transform, jpeg compression with a quality coefficient at 50, background edit, local edit.

modulation and better representation of the message to transmit. Further research on carrier modulation techniques could potentially enhance performance by reducing inter-carrier interference. On the latter, our choice of a text description may be considered simplistic compared to a specifically learnt representation. Moreover, we limit the granularity of captions to reduce the length of the message to encode. This hampers fine-grained comparison but we believe it will be further optimized. We leave to future work the task of designing a system taking advantage of our pipeline output: Alice could be considered as the source of the original media while Bob would be a moderation system on a social media platform.

V. Conclusion

In this work, we present a novel way to assert the integrity of an image by the relevant use of watermarking as a covert communication channel. Moreover we provide a definition of the image semantics, through its caption, to the recipient of the message. By using an LLM combined with an arithmetic encoder to compress the caption, the limited capacity of Hide-$\mathbb{R}$ to convey information is optimized. Finally, our local confidence metric improves the applicability of our method as any trusted operator may check if the received image is consistent with the descriptive decoded caption of the original content in a trustworthy manner.

Acknowledgments

Experiments were carried out using the Grid'5000 testbed, hosted by Inria and supported by CNRS, RENATER, and several Universities and other organizations (see https://www.grid5000.fr). Work supported by French ANR-20-CHIA-0011-01.

References

[1] M. Chen, J. Fridrich, M. Goljan, and J. Lukás, "Determining image origin and integrity using sensor noise," *IEEE Transactions on information forensics and security*, vol. 3, no. 1, pp. 74–90, 2008.

[2] H. Farid, *Photo forensics*. MIT press, 2016.

[3] M. Utku Celik, G. Sharma, E. Saber, and A. Murat Tekalp, "Hierarchical watermarking for secure image authentication with localization," *IEEE Transactions on Image Processing*, vol. 11, no. 6, pp. 585–595, 2002.

[4] J. Fridrich, M. Goljan, and A. Baldoza, "New fragile authentication watermark for images," in *ICIP*, vol. 1, pp. 446–449, 2000.

[5] J. Zhu, R. Kaplan, J. Johnson, and L. Fei-Fei, "Hidden: Hiding data with deep networks," in *ECCV*, 2018.

[6] C. S. K. Valmeekam, K. Narayanan, D. Kalathil, J.-F. Chamberland, and S. Shakkottai, "Llmzip: Lossless text compression using large language models," *arXiv:2306.04050*, 2023.

[7] Y. Wu, W. Abd-Almageed, and P. Natarajan, "Busternet: Detecting copy-move image forgery with source/target localization," in *ECCV*, 2018.

[8] P. Zhou, X. Han, V. I. Morariu, and L. S. Davis, "Learning rich features for image manipulation detection," in *CVPR*, 2018.

[9] Y. Wu, W. AbdAlmageed, and P. Natarajan, "Mantra-net: Manipulation tracing network for detection and localization of image forgeries with anomalous features," in *CVPR*, 2019.

[10] X. Zhu, Y. Qian, X. Zhao, B. Sun, and Y. Sun, "A deep learning approach to patch-based image inpainting forensics," *Signal Processing: Image Communication*, 2018.

[11] H. Li and J. Huang, "Localization of deep inpainting using high-pass fully convolutional network," in *ICCV*, 2019.

[12] S. Katzenbeisser and F. Petitcolas, *Information Hiding Techniques for Steganography and Digital Watermarking*. Artech House computer security series, Artech House, 1999.

[13] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, *Digital watermarking and steganography*. Morgan kaufmann, 2007.

[14] V. Vukotić, V. Chappelier, and T. Furon, "Are deep neural networks good for blind image watermarking?," in *2018 IEEE International Workshop on Information Forensics and Security (WIFS)*, pp. 1–7, IEEE, 2018.

[15] K. A. Zhang, L. Xu, A. Cuesta-Infante, and K. Veeramachaneni, "Robust invisible video watermarking with attention," *arXiv*, vol. abs/1909.01285, 2019.

[16] P. Fernandez, A. Sablayrolles, T. Furon, H. Jégou, and M. Douze, "Watermarking images in self-supervised latent spaces," in *ICASSP*, 2022.

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

[18] H. He, F. Chen, and Y. Huo, "Self-embedding fragile watermarking scheme combined average with vq encoding," in *The International Conference on Digital Forensics and Watermarking*, pp. 120–134, 2012.

[19] J. Li, D. Li, S. Savarese, and S. C. H. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," in *ICML*, 2023.

[20] R. Pasco, "Source coding algorithms for fast data compression (ph.d. thesis abstr.)," *IEEE Transactions on Information Theory*, 1977.

[21] S. Zhang, S. Roller, N. Goyal, M. Artetxe, M. Chen, S. Chen, C. Dewan, M. T. Diab, X. Li, X. V. Lin, T. Mihaylov, M. Ott, S. Shleifer, K. Shuster, D. Simig, P. S. Koura, A. Sridhar, T. Wang, and L. Zettlemoyer, "Opt: Open pre-trained transformer language models," *arXiv*, vol. abs/2205.01068, 2022.

[22] Y. Huang, J. Zhang, Z. Shan, and J. He, "Compression represents intelligence linearly," *arXiv*, vol. abs/2404.09937, 2024.

[23] S. Yang, W.-T. Xiao, M. Zhang, S. Guo, J. Zhao, and S. Furao, "Image data augmentation for deep learning: A survey," *arXiv*, vol. abs/2204.08610, 2023.

[24] C. S. Alexandre Sablayrolles, Matthijs Douze and H. Jégou, "Spreading vectors for similarity search," in *ICLR*, 2019.

[25] G. Dillard, M. Reuter, J. Zeiddler, and B. Zeidler, "Cyclic code shift keying: a low probability of intercept communication technique," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 39, no. 3, 2003.

[26] R. A. Fisher, *Statistical Methods for Research Workers*, pp. 66–70. New York, NY: Springer New York, 1992.

[27] S. Sheynin, A. Polyak, U. Singer, Y. Kirstain, A. Zohar, O. Ashual, D. Parikh, and Y. Taigman, "Emu edit: Precise image editing via recognition and generation tasks," *arXiv*, vol. abs/2311.10089, 2023.

[28] X. Chen, H. Fang, T.-Y. Lin, R. Vedantam, S. Gupta, P. Dollar, and C. L. Zitnick, "Microsoft coco captions: Data collection and evaluation server," *arXiv:1504.00325*, 2015.