Robust Image Steganography against General Scaling Attacks

Qingliang Liu
School of Computer Science and Engineering, Sun Yat-sen University
Guangzhou, China
liuqliang3@mail2.sysu.edu.cn

Jiangqun Ni*
School of Cyber Science and Technology, Sun Yat-sen University
Department of New Networks, Peng Cheng Laboratory
Shenzhen, China
issjqni@mail.sysu.edu.cn

Xianglei Hu
Guangdong Polytechnic Normal University
Guangzhou, China
Zhengzhou Xinda Institute of Advanced Technology
Zhengzhou, China
huxianglei@gpnu.edu.cn

ABSTRACT

Conventional image steganography is assumed to transmit the message, in the most securest way possible for a given payload, over lossless channels, and the associated steganographic schemes are generally vulnerable to active attacks, e.g., JPEG re-compression, and scaling, as seen on social networks. Although considerable progress has been made on robust steganography against JPEG re-compression, there exist few steganographic schemes capable of resisting scaling attacks due to the tricky inverse interpolations involved in algorithm design. To tackle this issue, a framework for robust image steganography resisting scaling with general interpolations either in std form with fixed interpolation block, or pre-filtering-based anti-aliasing implementation with variable block, is proposed in this paper. And the task of robust steganography can be formulated as one of constrained integer programming aiming at perfectly recovering the secret message from the stego image while minimizing the difference between cover and stego images and the embedding distortion between scaled cover and scaled stego images. By introducing a metric - the degree of pixel involvement (dPI) to identify the modifiable pixels in the cover image, the optimization problem above could be effectively solved using the branch and bound algorithm (B&B). Extensive experiments demonstrate that the proposed scheme could not only resist scaling attacks with various interpolation techniques at arbitrary scaling factors (SFs), but also outperform the prior art in terms of security between the cover and stego images by a clear margin. In addition, the application of the proposed method in LinkedIn against the joint attacks of scaling and JPEG re-compression also shows its effectiveness on social network in real-world scenarios.

CCS CONCEPTS

• **Security and privacy → Social aspects of security and privacy**.

*Corresponding Author.

MM '23, October 29-November 3, 2023, Ottawa, ON, Canada

ACM ISBN 979-8-4007-0108-5/23/10…$15.00
https://doi.org/10.1145/3581783.3612267

KEYWORDS

robust steganography, scaling attacks, integer programming, branch and bound algorithm, security

ACM Reference Format:
Qingliang Liu, Jiangqun Ni, and Xianglei Hu. 2023. Robust Image Steganography against General Scaling Attacks. In *Proceedings of the 31st ACM International Conference on Multimedia (MM '23), October 29-November 3, 2023, Ottawa, ON, Canada.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3581783.3612267

1 INTRODUCTION

Image steganography is the science and art of covert communication, in which the sender embeds the secret message into the cover images by slightly modifying the pixel values (in spatial domain) or the quantized DCT coefficients (in JPEG domain). To conceal the very existence of communication, the stego images have to be statistically indistinguishable from the cover images.

Traditionally, image steganography assumes a lossless channel without active attacks. To achieve superior security, the past decade has witnessed the emergence of a large number of content-adaptive image steganographic methods based on the framework of minimal distortion embedding, either heuristically designed or model-driven. According to the domain in which the secret messages are embedded, these methods can be categorized as the ones in spatial domain and JPEG domain, respectively. For image steganography in spatial domain, the messages are embedded into the complex regions with rich texture contents in cover images, such as WOW [10], S-UNIWARD [11], HiLL [14], MiPOD [17], GMRF [20] etc. While for JPEG steganography, the quantized DCT coefficients corresponding to complex texture regions, which are more tolerant to steganalytic attacks, are utilized in priority for data embedding, such as UED [8], UERD [9], J-UNIWARD [11], BET [12], J-MiPOD [3] etc.

Traditional image steganography, however, could by no means survive the lossy channel with active attacks, e.g., scaling, JPEG re-compression, or the combination of the two. To achieve secure covert communication under known active attacks, researchers have made several attempts. For the lossy channel with JPEG re-compression attack, the effective steganographic scheme is achieved by transport channel matching [7, 22, 26], by fitting the inverse procedure of the JPEG compression channel with a pre-trained auto-encoder [15] etc. For the lossy channel with scaling attack, however, there exist few effective steganographic schemes due to the tricky inverse interpolations involved in the algorithm design. Unless otherwise specified, only the down-scaling attacks with a

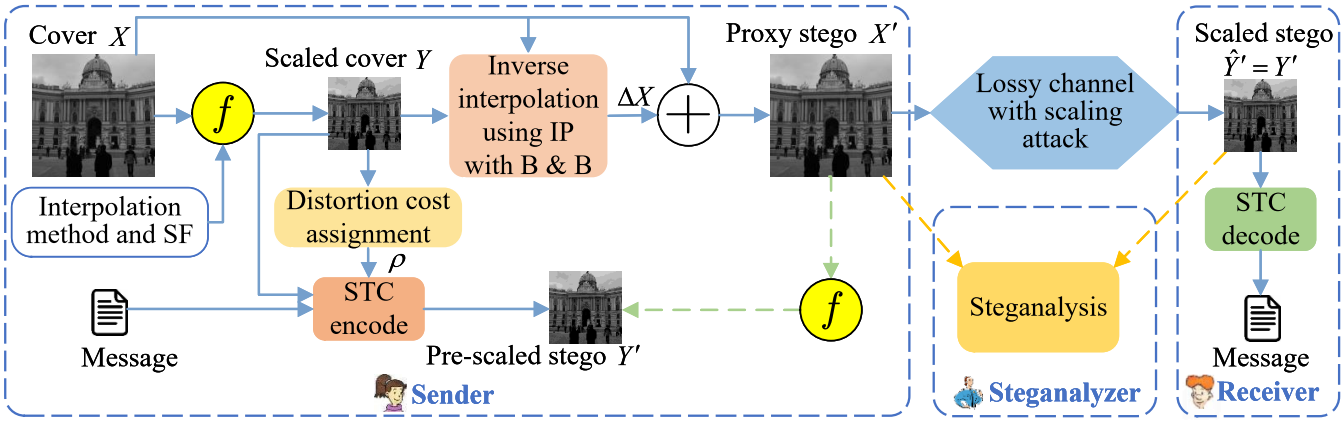


Figure 1: The diagram of the proposed steganographic framework against general scaling attacks.

scaling factor (SF) in $(0, 1]$ are addressed throughout the paper, as they are the ones the social networks run into most frequently. Early anti-scaling steganographic schemes [23, 25] heavily depend on watermarking techniques, albeit being capable of surviving the scaling attacks, they usually exhibit low embedding capacity and negligible security performance. Recently, Luo et al. [24] proposed an image steganographic scheme against nearest neighbor interpolation scaling attack by taking advantage of the invariant pixels to embed messages. Later, they further extend their method to the general scaling attacks with anti-aliasing bilinear and bicubic interpolations by incorporating the inverse interpolation rule from scaled images to original cover images [27]. Although the relatively large embedding capacity and satisfactory statistical undetectability are achieved for scaled stego images, the improvement of the security performance for the generated stego images at the same dimension as the original images (the proxy stego images in this paper) is not evident due to the selection of modifiable pixels in cover images in the inverse interpolation process. In addition, this method [27] is only effective when the SFs for anti-aliasing bilinear and bicubic interpolation scaling are confined to (0,0.5] and (0,0.25], respectively.

In this paper, we propose a flexible framework for robust image steganography against general scaling attacks at arbitrary scaling factors (SFs). For steganographic image covert communication over the lossy channel with known general scaling attacks as shown in Fig. 1, the cover image X is resized to the scaled cover image Y, in which the message is embedded to generate the pre-scaled stego image Y'. The proxy stego image X', which is the same dimension as X and to be uploaded to the scaling channel, i.e., $X' = X + \Delta X$, is sought based on Y' through inverse interpolation with the objective to minimize $\|X - X'\|_1$ for the possibly best security performance and reliably recover the $f(X')$ (robustness), where f is the function of involved interpolation scaling. Note that, for practical applications, the interpolation methods and SFs involved for a specific scaling channel are usually fixed and could be approximately identified through channel test, therefore, **we thus assume a known scaling channel throughout the paper**. The main contributions of our scheme are summarized as follows :

- The task of robust image steganography against scaling attacks is formulated as the problem of constrained integer programming (IP) aiming at minimizing $\|X - X'\|_1$ for the possibly best security performance of proxy stego image X', while reliably recover the message from $f(X')$ (robustness).
- A metric known as the degree of pixel involvement (dPI) is introduced to identify the modifiable pixels ($\Delta x_{i,j} \neq 0$) in cover X, and the above IP problem can then be effectively solved using the branch and bound (B&B) algorithm. In addition, the dPI could also be incorporated to assign the embedding cost for scaled cover Y to generate pre-scaled stego Y' with superior performance.
- The proposed scheme is not only shown to outperform the prior arts in terms of security performance for proxy stego X' with evidence from extensive experiments using both standard bilinear/bicubic interpolation scaling and their anti-aliasing variants at arbitrary SFs, but also applied to the real-world social networks, e.g., LinkedIn, to survive the joint attacks of scaling and JPEG re-compression.

2 PRELIMINARY AND RELATED WORK

2.1 Image Interpolation Scaling

Since the involved scaling attack on social network applications (e.g., Facebook, LinkedIn, etc.) are usually down-scaling with interpolations, this paper only concentrates on the down-scaling channels with the SFs confined to $(0, 1]$, unless otherwise specified.

Image scaling operation consists of geometric transformation and interpolations, e.g., nearest, bilinear, and bicubic either in std forms with fixed block sizes, or their anti-aliasing variants with variable block sizes. Note that anti-aliasing attempts to minimize the appearance of jagged diagonal edges. It works by taking into account how much an ideal edge overlaps adjacent pixels and is widely adopted in social networks. For a given cover image $X = [x_{i,j}]$, it is shrunk with interpolation to generate the scaled cover image $Y = [y_{u,v}]$. In general, backward mapping is adopted to re-sample $y_{u,v}$ in Y based on the closest neighbors $P = [p_{i,j}]$ (interpolation

block) in X to $T^{-1}(u,v)$, where $p_{i,j} \in X$, T^{-1} is the backward geometric mapping. Let $W = [w_{s,t}]$ be the involved interpolation kernel, and the $y_{u,v}$ can be determined by the weighted sum of the pixel values defined in the closest neighbor $P = [x_{i,j}]$. In an implementation, the W can usually be factorized into $W^{(V)}$ and $W^{(H)}$, along the vertical and horizontal directions, i.e., $W = W^{(V)} \cdot (W^{(H)})^{\tau}$, where τ denotes the transpose of a vector or matrix in this work, thus we have,

$$y_{u,v} = round((W^{(V)})^{\tau} \cdot P \cdot W^{(H)}), \tag{1}$$

where $round(\cdot)$ is the rounding function, $W^{(V)}$ and $W^{(H)}$ are the 1-D interpolation kernels in column vector.

For pre-filtering-based anti-aliasing scaling, the high-frequency components of X are attenuated with an anti-aliasing filter W_{anti} to avoid aliasing before sampling at pixel rates. With an interpolation kernel W_s in std form (bilinear and bicubic), the equivalent anti-aliasing interpolation kernel can be re-written as,

$$W = W_{anti} \otimes W_s, \tag{2}$$

where $\otimes$ represents convolution, and the W_{anti} is performed zeros padding to keep $W_{anti} \otimes W_s$ in the same dimension as W_{anti}.

Generally speaking, the interpolation kernel has bell-shaped distribution and its size varies with the SFs. The smaller SFs, the larger kernel size becomes. Take the anti-aliasing bilinear interpolation scaling at $SF = 0.25$ and 0.5 as an example, the corresponding sizes for kernel W or interpolation block P are 8×8 and 4×4, respectively, as compared to the fixed block of 2×2 for interpolation in std form.

2.2 Degree of Pixel Involvement

In the proposed framework of robust image steganography against general scaling attacks, the key idea is to seek the proxy stego image X' according to X, Y', the involved interpolation method and its SF by incorporating the constrained integer programming. To identify the modifiable pixels ($\Delta x_{i,j} \geq 0$ or $\Delta x_{i,j} \leq 0$) in X, a metric known as the degree of pixel involvement (dPI) is introduced. For a pixel $x_{i,j}$ in X, its dPI is defined as the number of its involvement in the interpolation computation of embeddable pixels in Y. To allow pixels in X with relatively larger dPI (e.g., $\text{dPI}(x_{i,j}) \geq 2$) to be modifiable pixels to generate X' would lead to either the task of constrained integer programming for inverse interpolation hard to be solved, or the variations of modifiable pixels in X', in response to the embedding modifications in Y', extremely large as compared to X.

2.3 Related Works

At present, comprehensive research in depth on the robust image steganography capable of resisting general scaling attacks at arbitrary SFs while maintaining security, is not available on record. The most relevant work to our proposed method is the one in [27], where the scheme in [27] is only generalized to the scaling attacks with anti-aliasing bilinear (for SFs in $(0, 0.5]$) and bicubic (for SFs in $(0, 0.25]$) interpolations by exploring the inverse interpolation.

To more clearly illustrate Zhu's method [27] as compared to our approach, we take the anti-aliasing bilinear interpolation at $SF = 0.5$ as an example. In Zhu's method, each pixel in Y' is assigned to be the embeddable one, consequently, the dPI value for most of the pixels in X is 4 except for a few pixels on the boundary of the image. Let's denote the 4 adjacent embeddable pixels in Y as $y_{u,v}, \cdots, y_{u+1,v+1}$ and their corresponding closest neighbors or interpolation blocks of 4×4 in X as $P_{u,v}, \cdots, P_{u+1,v+1}$, as shown in Fig. 2(a). Zhu's method tries to find the intersection block of 2×2 among $P_{u,v}, \cdots, P_{u+1,v+1}$ in X, i.e., $XP_{u,v}$ bounded by the black round block as shown in Fig. 2(a), which we call supporting block in X for $y_{u,v}$ in the process of inverse interpolation, i.e., only the pixels in XP can be modified to meet the embedding modifications of $y_{u,v}$, while the rest of the pixels are the same as the corresponding pixels in X. Note that all the 4 pixels in $XP_{u,v}$ involve in the interpolation computation of $y_{u,v}, y_{u,v+1}, y_{u+1,v}$, and $y_{u+1,v+1}$ at the same time, which is always achievable for anti-aliasing bilinear interpolation for $SF \leq 0.5$. Following the notion in (1), let $W_{u,v} = W^{(V)}_{u,v}(W^{(H)}_{u,v})^{\tau}, \cdots, W_{u+1,v+1} = W^{(V)}_{u+1,v+1}(W^{(H)}_{u+1,v+1})^{\tau}$ to be the partial weighs for $XP_{u,v}$ corresponding to $y_{u,v}, \cdots, y_{u+1,v+1}$, respectively, the inverse interpolation equation set for $\Delta XP_{u,v}$ is built by assigning the $XP_{u,v}$ is only contributed to $y_{u,v} \pm 1$ for the possible embedding modifications at $y_{u,v}, \cdots, y_{u+1,v+1}$, i.e.,

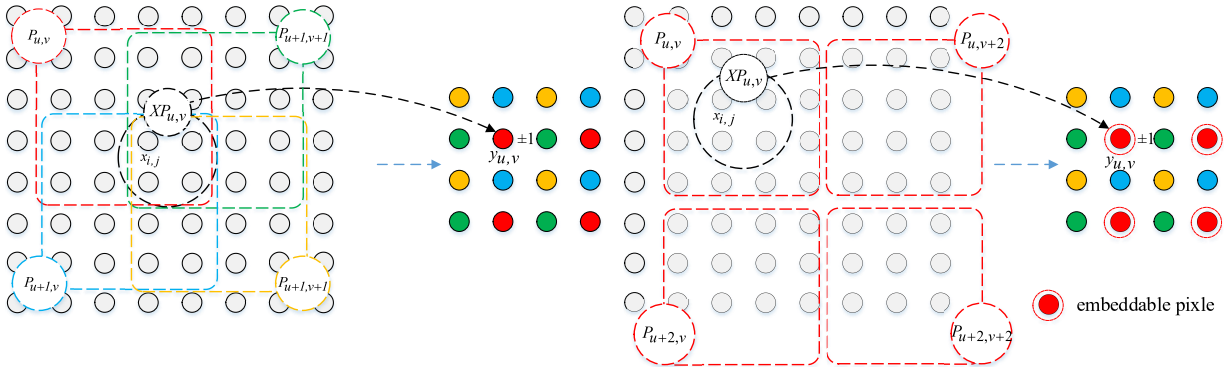
$$\begin{cases} round((W^{(V)}_{u,v})^{\tau} \cdot \Delta XP_{u,v} \cdot W^{(H)}_{u,v}) = \pm 1 \\ round((W^{(V)}_{u,v+1})^{\tau} \cdot \Delta XP_{u,v} \cdot W^{(H)}_{u,v+1}) = 0 \\ round((W^{(V)}_{u+1,v})^{\tau} \cdot \Delta XP_{u,v} \cdot W^{(H)}_{u+1,v}) = 0 \\ round((W^{(V)}_{u+1,v+1})^{\tau} \cdot \Delta XP_{u,v} \cdot W^{(H)}_{u+1,v+1}) = 0 \end{cases}, \tag{3}$$

where $\Delta XP_{u,v}$ is the variations compared to $XP_{u,v}$ in response to the embedding at $y_{u,v}$ in the inverse interpolation. The problems with the method in [27] are two-fold: (1) for given $y_{u,v}$ in Y, the determined $XP_{u,v}$ is inevitably located in the boundary of $P_{u,v}$ with smaller weights, leading to relatively larger modification $\Delta XP_{u,v}$ due to the embedding modifications at $y_{u,v}$; (2) for scaling attacks with anti-aliasing bilinear interpolation at $SF > 0.5$ (or bicubic at $SF > 0.25$), the dPI values for most of the pixels in X are greater than 4, and the established constraint equation sets tend to be over-determined accordingly, which are usually unsolvable. These issues, however, could be effectively solved with our proposed method, which will be elaborated later in Section 3.1.

3 THE FRAMEWORK OF ROBUST IMAGE STEGANOGRAPHY AGAINST GENERAL INTERPOLATION SCALING ATTACKS

3.1 Image Inverse Interpolation As Constrained Integer Programming

The proposed framework for robust image steganography against known general scaling attacks is shown in Fig. 1, where the proxy stego image $X' = X + \Delta X$ with the same dimension as cover X is generated from pre-scaled stego Y' through inverse interpolation using IP with B&B. And the task of robust image steganography can then be formulated as one of constrained IP with the objective to obtain X', which could not only survive the general interpolation scaling attacks but also be statistically indistinguishable from X. Accordingly, the security of scaled stego image $\hat{Y}'$ (or equivalently the pre-scaled stego image Y') compared to the scaled cover Y could also be achieved.



(a) Zhu's Scheme (b) Our Scheme

Figure 2: The diagram of robust image steganography schemes for anti-aliasing bilinear interpolation with $SF = 0.5$.

For each embeddable pixel $y_{u,v}$ in Y', let $P_{u,v}$ of size $p \times p$ and $XP_{u,v}$ be its interpolation and supporting blocks in X, the inverse interpolation is utilized by taking advantage of the constraint between $y_{u,v}$ and its supporting block $XP_{u,v}$ to obtain the variation $\Delta XP_{u,v}$ due to the embedding modifications at $y_{u,v}$. Unlike Zhu's method [27], where the involved supporting blocks are composed of 2×2 with $dPI = 4$, the proposed method assigns a unique supporting block $XP_{u,v}$ of 2×2 adjacent pixels centered on its interpolation block $P_{u,v}$ in X for each embeddable pixel $y_{u,v}$ in Y' as shown in Fig. 2(b). Note that the $XP_{u,v}$ in our method includes sufficient pixels with dPI = 1 to be modified in inverse interpolation corresponding to the embedding modifications at $y_{u,v}$. This allocation of supporting blocks could ensure the task of inverse interpolation is always solvable and maintain the less variations for ΔX in inverse interpolation to improve the security performance of X'. To this end, Y is split into the sub-images YEP and YIP, which are composed of embeddable pixels and idle pixels, respectively. For a given interpolation method with SF and interpolation block P of $p \times p$, the YEP for data embedding is generated by sampling Y with interval s along the horizontal and vertical directions. Note that the sampling interval s for different interpolation methods with various SFs is determined based on the rule of payload first, where the sampling interval $s = 1$ is adopted for std bilinear interpolation with SF in $(0, 0.5]$ and std bicubic interpolation with SF in $(0, 0.25]$, i.e, the generated YEP is Y itself, while $s = \lceil p/2 \rceil$ for anti-aliasing bilinear and bicubic interpolations. We observed that the sampling interval s is decreased with the increase of SF under the anti-aliasing scaling, sufficient pixels (embeddable pixels) in Y are usually preserved for data embedding at relatively large SF, which is quite affordable for practical applications.

Recall that, in our proposed framework, for each embeddable pixel $y_{u,v}$ in Y, only the pixels with dPI = 1 in its supporting block $XP_{u,v}$ rather than its interpolation block $P_{u,v}$ in X are considered to be modified in inverse interpolation. In general, allowing more pixels in $XP_{u,v}$ to be modified would lead to more subtle changes of $\Delta XP_{u,v}$, thus more statistically undetectable for X'. For practical applications, the determination of modifiable pixels in $XP_{u,v}$ are also subjected to other two issues besides their dPI values, i.e., the variation bound and over/underflow in reverse interpolation. In specific, let $M_{u,v} = [m_{i,j}]_{2\times 2}$ be the mask matrix to identify the modifiable pixels in $XP_{u,v}$, i.e, $m_{i,j} = 1$ or 0 when the pixel $x_{i,j}$ is modifiable or not. We then have the constraint equation for ΔXP, i.e, the variation of supporting block $XP_{u,v}$ due to the embedding modifications $\Delta y_{u,v}$ at $y_{u,v}$,

$$round((W_{u,v}^{(V)})^{\tau} \cdot (M_{u,v} \odot \Delta XP_{u,v}) \cdot W_{u,v}^{(H)}) = \Delta y_{u,v}, \tag{4}$$

where $W_{u,v}^{(V)}$ and $W_{u,v}^{(H)}$ are the partial interpolation weights for $XP_{u,v}$, $\odot$ is the operator for element-wise multiplication and $\Delta y_{u,v} \in \{0, 1, -1\}$. For $\forall y_{u,v} \in YEP$, let $XP_{u,v}^{s}$ be the possibly largest sub-set of $XP_{u,v}$, which consists of the modifiable pixels $x_{i,j}$ with associated weight $w_{i,j}$, and $\omega_{u,v} = \sum_{i,j} w_{i,j}$ for $x_{i,j} \in XP_{u,v}^{s}$. For $\Delta y_{u,v} = 0$, a trivial solution for (4) is readily available, i.e., $\Delta XP_{u,v} = 0$, while for $\Delta y_{u,v} = \pm 1$, the variation bound for $\forall y_{u,v} \in YEP$ can be determined as,

$$\Delta_{u,v} = \lceil 1/\omega_{u,v} \rceil. \tag{5}$$

The bound $\Delta_{u,v}$ is closely relevant to the security performance of X' and should be carefully determined. Although the $\Delta_{u,v}$ is $XP_{u,v}$ dependent, according to our experimental results, in most cases, $\omega_{u,v} \geq 0.5$, and we could usually take $\Delta_{u,v} \leq 2$ (the possible maximum variations for $x_{i,j}$ are ± 2) in our implementation for security concerns. With given $\Delta_{u,v}$ and 8 bit grey scale image, we further let $x_{i,j} \pm \Delta_{u,v} \in \mathbb{N}_{256}$ to prevent the generated X' from over/underflowing. Therefore, the sub-set $XP_{u,v}^{s}$ for $XP_{u,v}$ can be explicitly defined as,

$$XP_{u,v}^{s} = \{x_{i,j} | x_{i,j} \in XP_{u,v}, \mathrm{dPI}(x_{i,j}) = 1, x_{i,j} \pm \Delta_{u,v} \in \mathbb{N}_{256} \text{ and } \lceil 1/\sum_{i,j} w_{i,j} \rceil \leq \Delta_{u,v}\}. \tag{6}$$

And the $M_{u,v}$ for $XP_{u,v}$ can be determined accordingly. In specific, for $\forall x_{i,j} \in XP_{u,v}$, if $x_{i,j} \in XP_{u,v}^{s}$ then $m_{i,j} = 1$, otherwise $m_{i,j} = 0$.

We then proceed to the generation of $X' = X + \Delta X$ based on X, the involved interpolation scaling method with known SF and the given payload α of the message msg relative to Y. In the proposed method, to minimize the embedding distortion cost between Y and Y', msg is embedded into YEP of Y to obtain $YEP' = Emb(YEP, \rho, msg) = [y_{u,v} + \Delta y_{u,v}]$ by incorporating some existing SOTA steganographic schemes, with customized design for scaled images. On the other hand, for $\forall y_{u,v} \in YEP$, there exists a unique

correspondence between $y_{u,v}$ and its supporting block $XP_{u,v}$ in X. Denote $X^{(1)} = \bigcup_{u,v} XP_{u,v}$, X can then be decomposed into $X^{(1)}$ and its complement $X^{(2)}$. The task of robust image steganography against scaling attacks amounts to determine the possible optimal variations $\Delta X = \Delta X^{(1)} \cup \Delta X^{(2)}$ due to data embedding, which can be formulated as the following constrained IP:

$$\min_{\Delta X} \ \|X - X'\|_1,$$
$$\text{s.t.} \begin{cases} round((W_{u,v}^{(V)})^\tau \cdot (M_{u,v} \odot \Delta XP_{u,v}) \cdot W_{u,v}^{(H)}) = \Delta y_{u,v}, \\ \qquad \Delta y_{u,v} \in \{0, \pm 1\}, \forall y_{u,v} \in YEP \\ YEP' = [y_{u,v} + \Delta y_{u,v}] = Emb(YEP, \rho, msg) \\ X' = X + \Delta X \\ X^{(1)} = \bigcup_{u,v} XP_{u,v} \\ X = X^{(1)} \cup X^{(2)} \\ Y = YEP \cup YIP \end{cases}, \quad (7)$$

where ρ is the specifically designed embedding cost function for Y, which will be discussed later in Section 3.2. Consequently, we have $\Delta X = \Delta X^{(1)} \cup \Delta X^{(2)}$, among which, the solution for $\Delta X^{(2)}$ is trivial and readily available, i.e., $\Delta X^{(2)} = 0$, while the one for $\Delta X^{(1)}$ is non-trivial and could be sought by incorporating the B&B. The B&B recursively splits the admissible solution space into a series of sub-solution spaces, i.e., branching, and searches the solution therein. To improve the efficiency of solution searching, the B&B then keeps track of bounds on the minimum that is trying to find, and uses the bounds to prune the search space to find the optimal solution, i.e., bounding. To apply B&B to solve the constrained IP in (7), for $\forall y_{u,v} \in YEP$, the searching for solution $\Delta XP_{u,v}$ could be classified into two scenarios according to if there exists a nonempty set $XP_{u,v}^s$ defined in (6) for $\Delta y_{u,v} \pm 1$, and can be obtained by the pseudo-code in Algorithm 1.

Note that the proposed method could be easily generalized to the scaling channel with nearest neighbor interpolation, which is nothing more than a special case of the scaling channels under consideration. Under the circumstance, YEP is Y itself, and for $\forall y_{u,v} \in YEP$, its $P_{u,v}$ comprises of 2×2 pixels, in which the pixel nearest to $T^{-1}(u, v)$ is used to constitute $XP_{u,v}$ of size 1×1 with weight = 1 and dPI = 1, where the $T^{-1}(u, v)$ is the backward mapping of coordinate (u, v) in X.

3.2 The Construction of the Embedding Distortion Function for Scaled Images

In the proposed framework of robust image steganography against general scaling attacks, the messages are embedded into YEP of Y, which is then used to generate proxy stego X' by IP-based inverse interpolation with B&B algorithm. For $\forall y_{u,v} \in YEP$, there exists a unique $XP_{u,v}$ composed of sufficient pixels $x_{i,j}$ with $\mathrm{dPI}(x_{i,j}) = 1$ in X, and a mask matrix $M_{u,v} = [m_{i,j}]$ is adopted to identify which pixels in $XP_{u,v}$ are modifiable in the inverse interpolation. In general, the embedding distortion function of some existing SOTA steganographic schemes could be used to evaluate the cost for embeddable pixel $y_{u,v}$ based on YEP. Considering the fact of robust steganography against scaling attacks that maintaining the statistical indistinguishability of X' from X depends on the statistical indistinguishability of Y' from Y because the X' is generated by Y' through inverse interpolation. We thus propose an effective way to design embedding costs for Y based on the existing SOTA steganographic methods.

Let ψ be the adopted distortion function for an existing SOTA steganographic method, such as S-UNIWARD and HiLL. We have the customized design of embedding cost for YEP by taking into account the statistics of cover image X, which is defined as:

Algorithm 1: The IP-based inverse interpolation with B&B to obtain the proxy stego image X'.

Input: X, Y, and Y'
Output: X'

Determine the sampling interval s according to the known interpolation scaling method and SF, and obtain the YEP;
For $\forall y_{u,v} \in YEP$, calculate the embedding modifications $\Delta y_{u,v}$ and identify $XP_{u,v}$ in X, let $X^{(1)} = \bigcup_{u,v} XP_{u,v}$;
Decompose the X into $X^{(1)}$ and its complement $X^{(2)}$, and initialize $\Delta X^{(1)}$ and $\Delta X^{(2)}$ with 0;
for each $y_{u,v} \in YEP$ **do**
    Determine the largest sub-set $XP_{u,v}^s$ in $XP_{u,v}$, its set size $N_{u,v}$ and the mask $M_{u,v}$ for $XP_{u,v}$ according to (6);
    if $XP_{u,v}^s = \varnothing$ **then**
        $\Delta x_{i,j} = 0$ for $\forall x_{i,j} \in XP_{u,v}$
    end
    if $XP_{u,v}^s \neq \varnothing \& \Delta y_{u,v} = 0$ **then**
        $\Delta x_{i,j} = 0$ for $\forall x_{i,j} \in XP_{u,v}$
    end
    if $XP_{u,v}^s \neq \varnothing \& |\Delta y_{u,v}| = 1$ **then**
        Decompose the $\Delta XP_{u,v}$ into $\Delta XP_{u,v}^s$ and its complement $\Delta XP_{u,v}^c$, and assign $\Delta XP_{u,v}^s = 0$. The $\Delta x_{i,j}$s in $\Delta XP_{u,v}^s$ are initialized as 0, and sorted in descending order in terms of their weights $w_{i,j}$s. The coordinates of the sorted pixels in $\Delta XP_{u,v}^s$ are arranged as an array $Z[n]$ of length $N_{u,v}$;
        $n = 0$;
        while $|round((W_{u,v}^{(V)})^\tau \cdot (M_{u,v} \odot XP_{u,v}) \cdot W_{u,v}^{(H)})| \neq 1$ **do**
            if $\Delta y_{u,v} = 1$ **then**
                $\Delta x_{Z[n]} = \Delta x_{Z[n]} + 1$;
            end
            if $\Delta y_{u,v} = -1$ **then**
                $\Delta x_{Z[n]} = \Delta x_{Z[n]} - 1$;
            end
            $n = n + 1$;
            if $n \geq N_{u,v}$ **then**
                $n = 0$;
            end
        end
    end
end
return $X' = X + \Delta X$

Table 1: The feasibility comparison of the proposed method with other competing methods for various scaling channels.

Method \ Channel	Nearest $SF \in (0,1)$	Std Bilinear $SF \in (0,1)$	Std bicubic $SF \in (0,1)$	Anti-aliasing bilinear $SF \in (0,0.5]$	Anti-aliasing bilinear $SF \in (0.5,1)$	Anti-aliasing bilinear $SF \in (0,0.25]$	Anti-aliasing bilinear $SF \in (0.25,1)$
[24]	✓	✗	✗	✗	✗	✗	✗
[27]	✗	✗	✗	✓	✗	✗	✗
Ours	✓	✓	✓	✓	✓	✓	✓

$$\begin{cases} \rho_{u,v}^{(\pm 1)} = \psi_{u,v}^{\pm 1},\ y_{u,v} \in YEP \text{ and } XP_{u,v}^{s} \neq \varnothing \\ \rho_{u,v}^{(\pm 1)} = +\infty,\ y_{u,v} \in YEP \text{ and } XP_{u,v}^{s} = \varnothing \\ \rho_{u,v}^{(+0)} = 0,\ y_{u,v} \in YEP \end{cases}, \tag{8}$$

where $XP_{u,v}^{s} = \varnothing$ indicates that the embedding modifications (± 1) at $y_{u,v}$ in YEP would inevitably lead to the under/overflowing for $\forall x_{i,j} \in XP_{u,v}$ and $\mathrm{dPI}(x_{i,j}) = 1$, and $y_{u,v}$ is identified as a wet pixel with the cost of infinity.

4 EXPERIMENT RESULTS AND ANALYSIS

4.1 Experimental Settings

In this paper, the involved experiments are carried out on BOSSbase 1.01 [1], which comprises 10,000 $512 \times 512 \times 8$-bit gray-scale images with diverse texture characteristics and is widely adopted in image steganography. For a given interpolation scaling channel with a specific SF, the pre-scaled cover images Ys and its embeddable sub-images YEPs are generated. The SOTA content-adaptive image steganographic scheme in the spatial domain, namely, S-UNIWARD [10], is explored to design the customized distortion functions for scaled images within the framework of minimal distortion embedding. For the given scaling channel, embedding distortion function, and payload α, the messages are embedded into YEPs using ternary STC [5] with the parity-check matrix of $h = 10$ to obtain the scaled stego image Y', which are then used to generate the proxy stego image X's through IP-based inverse interpolation. It is worth noting that, throughout the paper, **the relative payload α for proxy stego X' and scaled stego Y' is identical, which is evaluated in terms of the size of the pre-scaled cover Y rather than the X or YEP in** Y, where the achievable relative payload $\alpha \leq (\log_2 3)/s^2$ where s is the sampling interval to obtain YEP. To evaluate the security performance of the involved robust image steganographic methods for both proxy stego X' vs. cover X and scaled stego Y' vs. scaled cover Y, the SOTA steganalyzer using SRM-34,671D [6] feature set with the Fisher linear discriminant ensemble classifier [13] is adopted, where the Fisher linear discriminants are used as base learners. In our implementation, half of X and X' pairs are randomly selected as the training set, and the remaining half is used as a test set to evaluate, which is the same for Y and $\hat{Y}'$ pairs. The security performance is quantified as the minimal total probability of error under equal priors achieved on the test set by ten times of random testing, denoted as $\overline{P}_E$. The robust performance is quantified as the mean bit error rate for message extraction from the tested scaled stego images, denoted as BER.

Table 2: The comparison of the proposed method with the competing method in terms of BER for various scaling channels SFs at the payload of 0.01 bpp.

Scheme	Anti-aliasing bilinear with various SF				Anti-aliasing bicubic with various SF			
	0.25	0.3	0.5	0.7	0.2	0.25	0.5	0.7
[27]	0	0	0	–	0.0001	0.0021	–	–
Ours	0	0	0	0	0	0	0	0

4.2 The Comparison of Robustness and Feasibility to Various Scaling Channels

The key objective of robust image steganography against scaling attacks is to survive the scaling channels. From the perspective of communication, the feasibility of a robust steganographic scheme could be evaluated in terms of robustness to reliably recover the embedded data from the noisy channels with and without channel coding. To this end, we calculate the BER of our proposed scheme for various scaling channels at various achievable payloads with different embedding distortion functions, e.g., S-UNIWARD. As expected, the proposed scheme could always perfectly recover the hidden messages from the received scaled stego images (i.e., $BER = 0$) with achievable payloads for various scaling channels at arbitrary SFs. This is because the task of inverse interpolation is well formulated as the problem of Integer Programming by taking into account the constraints between the proxy stegos and pre-scaled stegos due to embedding modifications. And the IP-based inverse interpolation could be effectively solved with the B & B algorithm. In other words, the proposed method could well survive the involved scaling attacks with arbitrary SFs, even without channel coding, as shown in Table 1, where "✓" and "✗" stand for the associated method is and is not applicable to the corresponding scaling channel, respectively. It is also observed from Table 1 that other competing methods could be only applicable to some specific scaling channels with relatively small SFs. To be specific, the method in [24] is specifically designed for the scaling channel with nearest neighbor interpolation, and the one in [27] is for the scaling attacks with anti-aliasing bilinear (for SFs in $(0, 0.5]$) and bicubic (for SFs in $(0, 0.25]$) interpolation. We further compare the robustness performance of our proposed method with the one in [27] in terms of BER for different scaling channels with various SFs at payload 0.01 bpp, as shown in Table 2. It is ready to see that, the proposed method could always achieve $BER = 0$ for various scaling channels at arbitrary SFs, whereas the method in [27] could only perfectly recover the messages for the anti-aliasing bilinear scaling channel with $SF \in (0, 05]$.

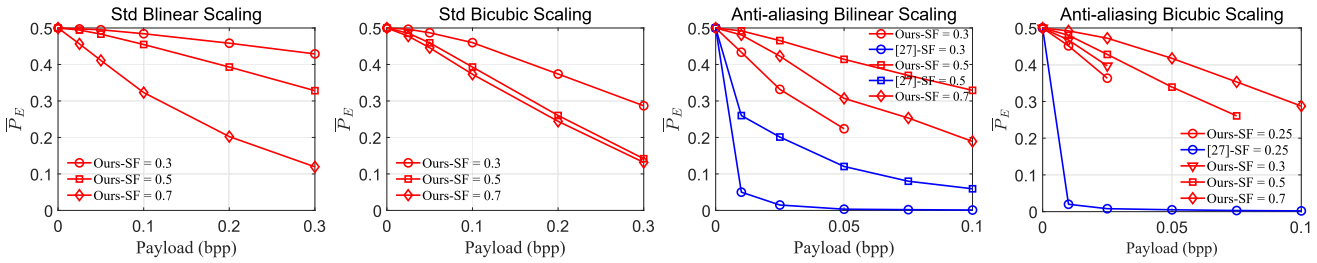


Figure 3: The comparison of security performance in terms of $\overline{P}_E$ between the proxy stego and cover images for various scaling channels, SFs, and payloads.

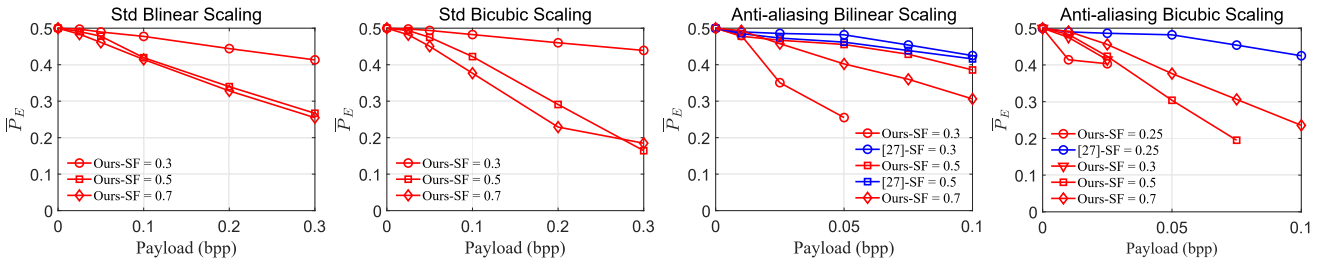


Figure 4: The comparison of security performance in terms of $\overline{P}_E$ between the scaled stego and cover images for various scaling channels, SFs, and payloads.

4.3 Comparison with Prior Arts in Terms of Security Performance

We then proceed to compare the security performance of our method with the competing method [27] for different scaling channels with various SFs and payloads, e.g., the std bilinear and bicubic scaling channels and their anti-aliasing variants, as shown in Fig. 3 and Fig. 4. In the interest of fair comparison, the same embedding distortion function borrowed from S-UNIWARD and steganalyzer using SRM-34,671D feature set, are adopted for embedding and steganalysis, respectively. Unlike [27], **where only the security performance for scaled stego images are evaluated**, we also compare the security performance for the proxy stego images with identical dimensions to cover images, this is because generating the secure and valid proxy stego image (e.g., no overflow) from pre-scaled stego image is much more challenging and prerequisite for practical applications. For the scaling channels with std bilinear and bicubic scaling, the competing method [27] is not applicable, the proposed method, however, could still achieve superior security performance not only for proxy stego images but also scaled stego images for various payloads and SFs, as shown in Fig. 3 and Fig. 4. As for the scaling channel with both anti-aliasing bilinear and bicubic interpolations, the proposed method consistently outperforms the one in [27] by a clear margin for proxy stego images at various SFs and payloads as shown in Fig. 3. This is because, with the method in [27], for $\forall y_{u,v}$ in scaled image Y, the corresponding supporting block $XP_{u,v}$ is usually located on the boundary of its interpolation block $P_{u,v}$ and assigned with smaller weights, thus leading to much larger variations $\Delta XP_{u,v}$ in proxy stego image X', **which may even result in overflow of generated proxy stego image in some circumstances according to our observation**, due to the embedding change $\Delta y_{u,v} = \pm 1$ through inverse interpolation. It is observed that the security performance of our method is inferior to Zhu's method [27] as shown in Fig. 4. Considering the fact that some of the generated proxy stegos from scaled stego images with Zhu's method are unstable, our proposed method exhibits much better overall performance in that it could generate proxy stego images with superior security performance for various scaling channels at arbitrary SFs, which is much more desirable in practical steganographic applications.

4.4 Evaluation in Social Network – LinkedIn

LinkedIn [4] is the world's largest social network to connect the world's professionals with more than 930 million members in more than 200 countries worldwide.

The application of the proposed method in LinkedIn is challenging because it would experience the joint attack of scaling interpolation f and JPEG compression g. Therefore the lossy channel should first be identified from a candidate joint attacking model set $G \circ F$ through the model test, where F = {nearest, std bilinear, std bicubic, anti-aliasing bilinear, anti-aliasing bicubic, etc.} and G = {libjpeg [19], mozjpeg [16], etc.}. As shown in Fig. 5, let X be the given spatial image to be uploaded, it may go through $Y = f(X)$ for interpolation scaling and $JY = g(Y)$ for JPEG compression. The adopted criterion for channel identification is as follows: for downloaded image JY corresponding to X, if $JY \approx g^*(f^*(X))$ for a set of tested spatial image set Xs, where $f^* \in F$ and $g^* \in G$, the channel model for joint scaling and JPEG compression could be determined as $g^* \circ f^*$, note that the identification of f and g also includes the determination of their associated parameters, e.g., the QFs for JPEG compression g. According to the result of our channel

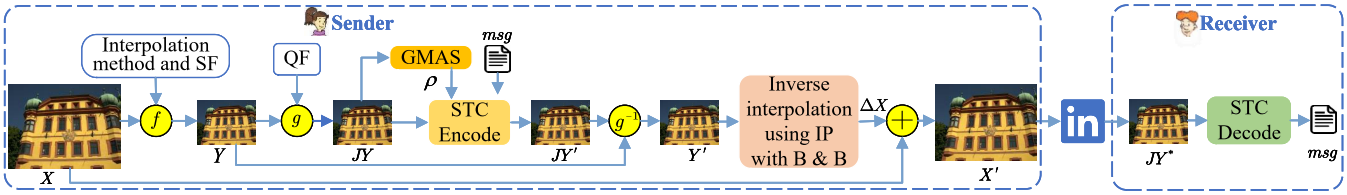


Figure 5: The application of the proposed method in the social network LinkedIn against joint attacks of scaling and JPEG compression

Table 3: The security performance in terms of $\overline{P}_E(X\&X')$ and $\overline{P}_E(JY\&JY^*)$, and robustness performance in terms of *BER* for the lossy LinkedIn channel at various payloads.

Performance	Relative payload α (bpnzAC)			
	0.01	0.02	0.05	0.1
$\overline{P}_E\ (X\&X')$	0.4975	0.4951	0.4843	0.4583
$\overline{P}_E\ (JY\&JY^*)$	0.3202	0.2926	0.1635	0.085
BER	0.0314	0.0316	0.0379	0.0513

test for LinkedIn, the std bilinear interpolation and libjpeg [19] with QF = 80 could be well determined as the involved components f and g respectively for the joint attacking model, and if the resolution of the uploaded image X is less than $800 \times 600 \times 3$, the operation of f is bypassed.

With the identified model $g^* \circ f^*$ for LinkedIn, we then proceed to use a robust steganographic scheme on the LinkedIn channel by integrating the proposed scheme against scaling attack with widely adopted GMAS [21] to resist the JPEG compression. In specific, for a given spatial image X (e.g., PNG image of $1600 \times 1200 \times 3$), we first perform the identified f and g on X to generate pre-scaled cover image Y and pre-scaled JPEG cover image JY. Then, the GMAS based on the embedding cost function from J-UNIWARD [11] is adopted to embed secret message *msg* into JY with STC [5] to obtain the pre-scaled JPEG stego JY', which is expected to minimize the embedding distortion between JY and JY' while resisting the attack of JPEG compression. Then, JY' is converted to a pre-scaled stego image Y' in the spatial domain by g^{-1} based on Y, which is used to generate the proxy stego image $X' = X + \Delta X$ with the proposed inverse interpolation scheme, i.e., IP with B&B. In addition, to ensure an acceptable security performance between X and X', ΔX is truncated by λ, i.e., $\forall \Delta x_{u,v} \in \Delta X$, if $\Delta x_{u,v} > \lambda$ ($\lambda = 8$ in our implementation), then $\Delta x_{u,v} = \lambda$.

Experiments are carried out to verify the effectiveness of our scheme on LinkedIn with an image data set, which consists of 2,000 color PNG images of $1600 \times 1200 \times$ 24-bit from ALASKA [2]. For the given payload α in bpnzAC (the ratio relative to the non-zero AC coefficients in JY), the proxy stego images can be generated with the proposed scheme for uploading to LinkedIn. The downloaded images are then used to evaluate the security and robustness performance in terms of $\overline{P}_E$ and *BER*. Note that, to improve the robustness performance, the Reed-Solomon Code RS(31,15) [21] is incorporated in our implementation. The SOTA steganalyzer [6] and GFR [18] are adopted to evaluate the security performance for spatial images (X vs. X') and JPEG images (JY vs. JY^*) respectively, as shown in Table 3. According to the experimental results in Table 3, the proposed scheme exhibits acceptable security performance, especially for low payload, while achieving relatively low *BER* for message decoding, which is quite applicable in LinkedIn. Note that, the *BER* could be further decreased to improve the robustness performance by adjusting the λ, decreasing the h of STC along with the enhancement of RS code.

5 CONCLUTION

The traditional image steganographic schemes are generally fragile to the lossy channels with interpolation scaling. In line with this, a framework for robust image steganography against general scaling attacks with either std bilinear and bicubic interpolations or their anti-aliasing variants is proposed in this paper. Following the principle of backward mapping for image interpolation scaling, the scaled image is decomposed into the embeddable sub-image (*YEP*) and idle sub-image (*YIP*) to establish a unique correspondence between each embeddable pixel in *YEP* and the supporting block centered on its interpolation block in the cover image. And the task of robust image steganography can then be formulated as one of constrained integer programming to obtain the proxy stego image with the identical dimension to the cover image, aiming at perfectly recovering the embedded data from the scaled stego image (robustness) while minimizing the L_1 norm between cover and proxy stego images (security). The dPI metric is adopted to identify the modifiable pixels in the cover image due to the embedding modifications in the scaled stego image, which is incorporated to effectively solve the constrained optimization problem with the branch and bound algorithm. Extensive experimental results demonstrate that the proposed scheme could not only survive the scaling attacks with various interpolation methods at arbitrary scaling factors (SFs), but also outperforms the prior arts in terms of security between the proxy stego and cover images by a clear margin. Importantly, the application of the proposed method in LinkedIn also shows its effectiveness on the social network in real-world scenarios.

ACKNOWLEDGMENTS

This work is supported in part by National Natural Science Foundation of China under Grants U1936212 and U22A2030, in part by the Key-Area Research and Development Program of Guangdong Province under Grant 2020B0101360001, in part by the Peng Cheng Laboratory Project under Grant PCL2021A02, in part by Open Foundation of Henan Key Laboratory of Cyberspace Situation Awareness HNTS2022021.

REFERENCES

[1] Patrick Bas, Tomáš Filler, and Tomáš Pevný. 2011. Break Our Steganographic System: the Ins and Outs of Organizing BOSS. In *Proceedings of 2011 Information Hiding - 13th International Conference(IH)*. Springer, 59–70.

[2] Rémi Cogranne, Quentin Giboulot, and Patrick Bas. 2020. ALASKA#2: Challenging Academic Research on Steganalysis with Realistic Images. In *Proceedings of 2020 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, 1–5.

[3] Rémi Cogranne, Quentin Giboulot, and Patrick Bas. 2022. Efficient Steganography in JPEG Images by Minimizing Performance of Optimal Detector. *IEEE Transactions on Information Forensics and Security* 17 (2022), 1328 – 1343.

[4] LinkedIn Corporation. 2023. LinkedIn. https://www.linkedin.com/feed.

[5] Tomáš Filler, Jan Judas, and Jessica Fridrich. 2011. Minimizing Additive Distortion in Steganography using Syndrome-trellis Codes. *IEEE Transactions on Information Forensics and Security* 6, 3 (2011), 920–935.

[6] Jessica Fridrich and Jan Kodovsky. 2012. Rich Models for Steganalysis of Digital Images. *IEEE Transactions on Information Forensics and Security* 7, 3 (2012), 868–882.

[7] Haocheng Fu, Xianfeng Zhao, and Xiaolei He. 2021. Improving Anticompression Robustness of JPEG Adaptive Steganography Based on Robustness Measurement and DCT Block Selection. *Security and Communication Networks* 2021 (2021), 1–15.

[8] Linjie Guo, Jiangqun Ni, and Yun-Qing Shi. 2014. Uniform Embedding for Efficient JPEG Steganography. *IEEE Transactions on Information Forensics and Security* 9, 5 (2014), 814 – 825.

[9] Linjie Guo, Jiangqun Ni, Wenkang Su, Chengpei Tang, and Yun-Qing Shi. 2015. Using Statistical Image Model for JPEG Steganography: Uniform Embedding Revisited. *IEEE Transactions on Information Forensics and Security* 10, 12 (2015), 2669–2680.

[10] Vojtěch Holub and Jessica Fridrich. 2012. Designing Steganographic Distortion Using Directional Filters. In *Proceedings of 2012 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, 234–239.

[11] Vojtěch Holub, Jessica Fridrich, and Tomáš Denemark. 2014. Universal Distortion Function for Steganography in An Arbitrary Domain. *EURASIP Journal on Information Security* 2014, 1 (2014), 1.

[12] Xianglei Hu, Jiangqun Ni, and Yun-Qing Shi. 2018. Efficient JPEG Steganography Using Domain Transformation of Embedding Entropy. *IEEE Signal Processing Letters* 25, 6 (2018), 773–777.

[13] Jan Kodovsky, Jessica Fridrich, and Vojtěch Holub. 2012. Ensemble Classifiers for Steganalysis of Digital Media. *IEEE Transactions on Information Forensics and Security* 7, 2 (2012), 432 – 444.

[14] Bin Li, Ming Wang, Jiwu Huang, and Xiaolong Li. 2014. A New Cost Function for Spatial Image Steganography. In *Proceedings of 2014 IEEE International Conference on Image Processing (ICIP)*. IEEE, 4206–4210.

[15] Wei Lu, Junhong Zhang, Xianfeng Zhao, Weiming Zhang, and Jiwu Huang. 2021. Secure Robust JPEG Steganography Based on AutoEncoder With Adaptive BCH Encoding. *IEEE Transactions on Circuits and Systems for Video Technology* 31, 7 (2021), 2909 – 2922.

[16] Mozilla. 2022. Mozjpeg. https://github.com/mozilla/mozjpeg.

[17] Vahid Sedighi, Rémi Cogranne, and Jessica Fridrich. 2016. Content-Adaptive Steganography by Minimizing Statistical Detectability. *IEEE Transactions on Information Forensics and Security* 11, 2 (2016), 221 – 234.

[18] Xiaofeng Song, Fenlin Liu, Chunfang Yang, Xiang Luo, and Yi Zhang. 2015. Steganalysis of Adaptive JPEG Steganography Using 2D Gabor Filters. In *Proceedings of the 3rd ACM Workshop on Information Hiding and Multimedia Security*. ACM, New York, USA, 15–23.

[19] SOURCEFORGE. 2023. Libjpeg-turbo. https://sourceforge.net/projects/libjpeg-turbo/.

[20] Wenkang Su, Jiangqun Ni, Xianglei Hu, and Jessica Fridrich. 2021. Image Steganography With Symmetric Embedding Using Gaussian Markov Random Field Model. *IEEE Transactions on Circuits and Systems for Video Technology* 31, 3 (2021), 1001–1015.

[21] Xinzhi Yu, Kejiang Chen, Yaofei Wang, Weixiang Li, Weiming Zhang, and Nenghai Yu. 2020. Robust Adaptive Steganography Based on Generalized Dither Modulation and Expanded Embedding Domain. *Signal Processing* 168 (2020), 107343.

[22] Jimin Zhang, Xianfeng Zhao, Xiaolei He, and Hong Zhang. 2022. Improving the Robustness of JPEG Steganography With Robustness Cost. *IEEE Signal Processing Letters* 29 (2022), 164–168.

[23] Yue Zhang, Xiangyang Luo, Yanqing Guo, Chuan Qin, and Fenlin Liu. 2019. Zernike Moment-Based Spatial Image Steganography Resisting Scaling Attack and Statistic Detection. *IEEE Access* 7 (2019), 24282 – 24289.

[24] Yue Zhang, Xiangyang Luo, Jinwei Wang, Chunfang Yang, and Fenlin Liu. 2018. A Robust Image Steganography Method Resistant to Scaling and Detection. *Jounal of Internet Technology* 19, 2 (2018), 607 – 618.

[25] Yue Zhang, Dengpan Ye, Junjun Gan, Zhenyu Li, and Qingfeng Cheng. 2018. An Image Steganography Algorithm Based on Quantization Index Modulation Resisting Scaling Attacks and Statistical Detection. *Computers, Materials and Continua* 56, 1 (2018), 151–167.

[26] Zengzhen Zhao, Qingxiao Guan, Hong Zhang, and Xianfeng Zhao. 2019. Improving the Robustness of Adaptive Steganographic Algorithms Based on Transport Channel Matching. *IEEE Transactions on Information Forensics and Security* 14, 7 (2019), 1843 – 1856.

[27] Liyan Zhu, Xiangyang Luo, Yi Zhang, Chunfang Yang, and Fenlin Liu. 2022. Inverse Interpolation and Its Application in Robust Image Steganography. *IEEE Transactions on Circuits and Systems for Video Technology* 32, 6 (2022), 4052 – 4064.