Image Semantic Steganography: A Way to Hide Information in Semantic Communication

Yanhao Huo, Shijun Xiang, Xiangyang Luo, and Xinpeng Zhang, *Member, IEEE*

***Abstract*— Semantic communication (SC) is an emerging communication paradigm that transmits only task-related semantic features to receivers, offering advantages in speed. However, existing robust steganography cannot extract message correctly after SC. To address this issues, we propose a novel steganography framework based on Generating Adversarial Networks (GANs) for SC, called "Image Semantic Steganography". Our framework embeds message into semantic features to guarantee extraction while considering both pixel-level and semantic-level distortions to enhance security. Experimental results show that our framework not only achieves message extraction successfully and behavioral covertness during and after SC, but also does not impact the implementation of SC.**

***Index Terms*— Semantic steganography, semantic communication, security, GANs.**

I. Introduction

THE advent of artificial intelligence (AI) has spawned many progresses in the field of information communication and security [1]. Semantic communication (SC), as an emerging communication paradigm based on AI, has been considered as the key to 6G wireless [2], [3], [4]. In contrast to traditional communication, SC only extracts and transmits necessary information relevant to the task at the receiver, i.e., semantic features, rather than each symbol or bit [5], [6], [7]. This significantly reduces the resources required for communication [8]. And SC has a high tolerance for syntactic errors and can perform well when the signal-to-noise ratio (SNR) is relatively low [9]. Thus, due to its advantages of high transmission efficiency and reliability [10], SC has been widely researched in the fields of augmented reality [11], virtual reality [12], video conferencing [13], [14], and intellectual transport system [15], [16], etc.

SC also has challenges with data security and privacy protection [17]. The transmitted semantic features may reveal the transmitter's private information, or under malicious attacks, the receiver may fail to recover the information correctly. Thus, a secure and reliable SC environment requires safeguarding the data confidentiality, integrity, availability, authenticity, and privacy [17]. While steganography hides the fact of secret communication by covertly embedding secret message into data media such as digital images, thus fulfilling the requirement of transmitting data confidentiality [18], [19], [20]. However, traditional image steganography assumes that the channel is lossless, whereas SC process performs lossy compression of the information at the transmitter (called semantic compression) and decompression at the receiver (called semantic reconstruction), as shown in Fig. 1 [21]. This requires image robust steganography, which generates stego image by embedding the information into the robust domain of cover image [22], [23]. It is not only resistant to steganalysis detection for security, but also correctly extracts the secret message through a lossy channel [24]. Zhang et al. first proposed the image robust steganographic framework of "robust domain construction + RS-STC coding" (Reed Solomon-Syndrome-Trellis codes), based on the relative relationship of neighboring DCT coefficient blocks [25]. Many advanced steganography methods emerged based on this framework and are categorized to target JPEG compression attack and multiple attacks [26]. For JPEG compression, dither modulation and asymmetric distortion are utilized to construct robust embedding domain, such as DMAS [27], GMAS [28], etc. Based on deep learning, Adaptive BCH [29] further reduced the error rate of message extraction by fitting the inverse process of JPEG compression. For multiple attacks such as compression, scaling, additive noise, etc., DCT coefficient difference is utilized to construct more robust embedding domain, such as MREAS [30], CRPAS [24], etc. Unfortunately, as shown in Fig. 1, none of these methods can extract message successfully after SC due to the fact that semantic compression causes severe loss of bit-level information. Therefore, a new steganographic framework is urgently needed to realize hidden transmission in SC. In addition, semantic compression results in less redundancy of semantic features, which creates a huge challenge for steganography.

Thus, in this paper, we propose a novel AI-based steganographic framework for SC, called "Image Semantic Steganography". As shown by the yellow line in Fig. 1, unlike general steganographic frameworks, semantic steganography embeds message into the semantic features to ensure that security and execution can be realized in the channel and after reconstruction. This framework is enabled to accomplish the following three tasks:

Received 24 July 2024; revised 19 September 2024; accepted 1 October 2024. Date of publication 9 October 2024; date of current version 13 February 2025. This work was supported by the National Key Research and Development Program of China under Grant 2023YFF0905000. This article was recommended by Associate Editor T. Zhang. *(Corresponding author: Shijun Xiang.)*

Yanhao Huo and Shijun Xiang are with the College of Information Science and Technology, Jinan University, Guangzhou 510632, China (e-mail: gzhu_hyh@163.com; Shijun_Xiang@qq.com).

Xiangyang Luo is with the State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou, Henan 450001, China (e-mail: luoxy_ieu@sina.com).

Xinpeng Zhang is with the School of Communication and Information Engineering, Shanghai University, Shanghai 200444, China (e-mail: xzhang@shu.edu.cn).

Digital Object Identifier 10.1109/TCSVT.2024.3476689

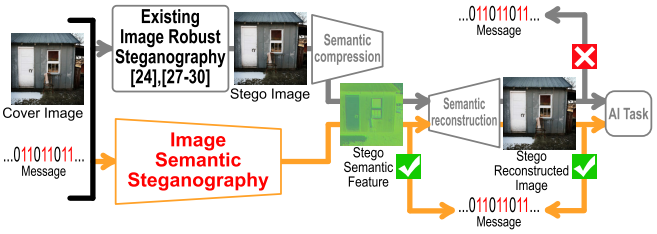

Fig. 1. The difference between image semantic steganography framework and existing robust steganography. Existing robust steganography cannot extract message correctly after semantic communication (SC). The method proposed in this paper not only achieves SC, but also achieves successful extraction of message during transmission and after reconstruction, and behavioral covertness.

- **Two-stage Security**: Image semantic steganography should ensure behavioral covertness both during channel transmission and after semantic reconstruction. This is because secret message can not only be leaked in the transmission channel, but also detected by third parties after reconstruction. In other words, the difference between Stego and Cover is reduced as much as possible, both for semantic features and reconstructed images, so that they cannot be detected by the steganalysis.
- **Two-stage Extraction**: Image semantic steganography should enable to successfully extract secret message both during channel transmission and after semantic reconstruction. That is, the embedded message can be correctly extracted both on the stego semantic features and the stego reconstructed images, which increases the extraction flexibility.
- **SC task**: Image semantic steganography should have virtually no impact on SC, i.e., the stego reconstructed images have very little difference from the original images and is capable to perform the later AI task.

Among them, Two-stage Security and Two-stage Extraction constitute the requirements for the steganography task. Experimental results show that the proposed framework successfully achieve both SC and steganography tasks.

The rest of this paper is organized as follows. Section II introduces the SC framework and problem formulation. Section III proposes the image semantic steganography framework. Section IV presents the experimental results and discussion. Finally conclude in section V.

II. Motivation

In this section, we introduce the SC framework as well as its evaluation metrics in detail, and explore the message extraction of five robust steganography methods after SC. This provides motivation and base for the image semantic steganography afterwards.

A. Semantic Communication (SC)

We consider a scene of end-to-end image SC across an lossless channel. As shown in Fig. 2, the dashed green line denotes the process of SC. First, the semantic encoder extracts the semantic features relevant to the AI task from the original image [31]. After the features pass through the channel, the image is reconstructed by the semantic decoder. Finally, AI task is performed based on the reconstructed image.

Generally, the input image $I_c \in \mathbb{R}^{C\times H\times W}$, is mapped to the features $I_s \in \mathbb{R}^{C'\times H'\times W'}$ via the semantic encoder E_s:

$$I_s = E_s(I_c) \tag{1}$$

where, $C'\times H'\times W' < C\times H\times W$, and the distribution of some channels in C will be changed after semantic compression. Then, the features I_s are mapped to the reconstructed image I_r by the semantic decoder D_s:

$$I_r = D_s(I_s) \tag{2}$$

where, after decoding $I_r \in \mathbb{R}^{C\times H\times W}$.

The main goal of SC is to achieve a trade-off between reconstruction distortion and AI task distortion under a certain compression rate (CR) [2], [32]. The CR represents the level of compression achieved by the model, calculated as the ratio of data size after and before compression. $CR = (C' \times H' \times W')/(C \times H \times W)$ The reconstruction distortion can be calculated via the Mean Squared Error (MSE) between the original and reconstructed image to measure the pixel-level error. And the AI task distortion is based on the type of task, e.g. image classification can be calculated by the cross entropy (CE). Thus, the goal is represented as minimizing the total distortion under a certain compression rate CR_0:

$$\begin{aligned} \min \quad & \lambda \cdot \mathrm{MSE}(I_r, I_c) + (1-\lambda)\cdot \mathrm{CE}(label_r, label_c) \\ \text{s.t. } & CR = CR_0 \end{aligned} \tag{3}$$

where, $\lambda \in [0, 1]$ is the weight coefficient. $\mathrm{CE}(\cdot)$ is calculated based on the pre-trained classifier model, $label_r$ is the predicted label of the reconstructed image I_r, i.e., $label_r = Classifier(I_r)$ and $label_c$ is the true label of the original image I_c.

The quality of reconstructed images is measured by peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM). PSNR measures the error of the corresponding pixel, and SSIM measures the image similarity in terms of luminance, contrast, and structure, respectively, which is closer to the human eye's visual perception. The PSNR is calculated as:

$$\mathrm{PSNR}(I_r, I_c) = 10 \cdot \log_{10}\left[\frac{\mathrm{MAX}^2}{\mathrm{MSE}(I_r, I_c)}\right] \tag{4}$$

where, the MAX is the maximum possible pixel value of the image. And the $\mathrm{MSE}(I_r, I_c)$ is the MSE between the original image I_c and the reconstructed image I_r. The SSIM is calculated as:

$$\mathrm{SSIM}(I_r, I_c) = \frac{(2\mu_r\mu_c + k_1R)(2\sigma_{rc} + k_2R)}{(\mu_r{}^2 + \mu_c{}^2 + k_1R)(\sigma_r{}^2 + \sigma_c{}^2 + k_2R)} \tag{5}$$

where μ_r, μ_c is the means, $\sigma_r{}^2$, $\sigma_c{}^2$ is the variances, and σ_{rc} is the covariance of images.

B. Image Robust Steganography After SC

In order to explore whether existing image robust steganography algorithms are resistant to the SC (Semantic Communication) process, i.e., whether stego images can still

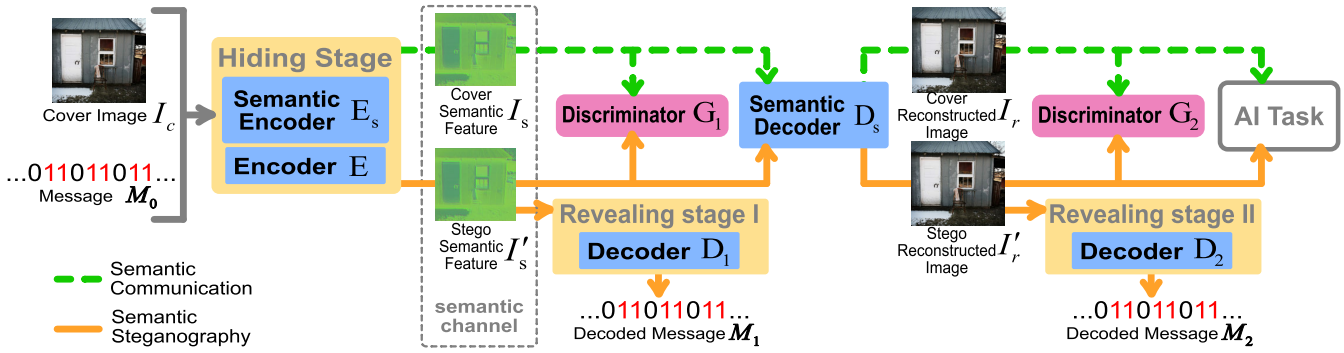

Fig. 2. Image semantic steganography framework. The dashed green line denotes the process of SC. The solid yellow line denotes the image semantic steganography process.

TABLE I
EXTRACTION BIT ACC AFTER ATTACK OF IMAGE ROBUST STEGANOGRAPHY

Payload (bpnzAC)	Method	Bit ACC (%) ↑		
		JPEG	Gaussian Noise	SC
0.1	DMAS [27]	99.91	-	**50.00**
	GMAS [28]	100.00	-	**51.08**
	Adaptive BCH [29]	93.16	-	**49.85**
	MREAS [30]	99.72	99.97	**51.69**
	CRPAS [24]	99.90	99.96	**51.05**

communicate after semantic compression and reconstruction. We use five classical algorithms for testing: DMAS [27], GMAS [28], Adaptive BCH [29], MREAS [30], CRPAS [24]. 10,000 grayscale images in the BOSSbase-1.01 dataset [33] are JPEG compressed with quality factor (QF) = 65 as $I_c \in \mathbb{R}^{1\times360\times360}$ for testing. For SC process, to control variables and simplify model, we only focus on semantic compression and reconstruction. As a result, at $CR_0 = 0.97$ and $\lambda = 1$, the PSNR and SSIM of SC process are 38.41dB and 98.05%, respectively.

When the payload is 0.1 bits per non-zero AC (bpnzAC), the bit accuracy (Bit ACC) of message extraction bit error rates after the attack is shown in Table I. For JPEG compression attack with QF = 50, these algorithms have extraction ACCs higher than 93.16%, and for Gaussian noise attack with mean $\mu = 0$ and variance $\sigma^2 = 0.3$, they have extraction ACCs higher than 99.96%. This indicates that all these algorithms have excellent robust performance after attacks. However, the Bit ACC after SC is around 50%, making it impossible to accurately extract the message, even though the $CR_0 = 0.97$ only. This is because the key to these algorithms is to construct robust domains that are easily maintained before and after attacks, commonly the medium frequency DCT domains or the two-dimensional relationships between DCTs [24]. As for semantic compression, only semantic-level high-dimensional features are transmitted in the channel, resulting in a large bit-level loss of the original image after SC. For instance, here the PSNR and SSIM between the SC reconstructed image and the original image are only 38.41dB and 98.05%. This makes the robust steganography extraction difficult which is based on two-dimensional robust domains. **Therefore, the existing robust steganography algorithms are not sufficient to extract the message correctly after SC and cannot realize the communication successfully.**

III. IMAGE SEMANTIC STEGANOGRAPHY

In this section, we propose "Image Semantic Steganography" as a solution to the vulnerability of robust steganography to SC process. Based on the analysis in Section II-B, our method embeds bit-level message into the semantic features for extraction and security within channel transmission and after reconstruction. Below, we detail the model architecture, loss function, and training process of our method.

A. Architecture

As shown in Fig. 2, the image semantic steganography process is illustrated by the solid yellow line, encompassing three distinct stages: the hiding stage, the first revealing stage, and the second revealing stage. There is flexibility in the extraction, which can be performed not only at the semantic level features, but also after the reconstructed image. During the hiding stage, the encoder embeds a message into cover semantic features to generate stego features. Subsequently, after a lossless channel, the decoder extracts the message from the stego features at the first revealing stage. The semantic decoder then reconstructs the stego image for subsequent AI tasks. Finally, at the second revealing stage, the decoder extracts message from the stego image. The primary objective of image semantic steganography is to achieve the steganography task without compromising the SC task. In other words, even if a message is embedded in the semantic features does not affect the subsequent AI tasks, such as image classification, target detection, semantic segmentation, style transformation, and so on.

Specifically, at the hiding stage, the semantic encoder E_s takes the input cover image $I_c \in \mathbb{R}^{C\times H\times W}$ to generate the cover semantic features I_s. And then, the encoder E takes the cover semantic features I_s and the message $M_0 \in \{0,1\}^{D\times H\times W}$ to generate the stego semantic features I'_s:

$$I_s = E_s(I_c), \tag{6a}$$

$$I'_s = E(I_s, M_0) \tag{6b}$$

where I'_s and $I_s \in \mathbb{R}^{C'\times H'\times W'}$ and $C'\times H'\times W' < C\times H\times W$. Then, at the first revealing stage, the decoder D_1 extracts the message M_1 from the stego semantic features I'_s:

$$M_1 = D_1(I'_s) \tag{7}$$

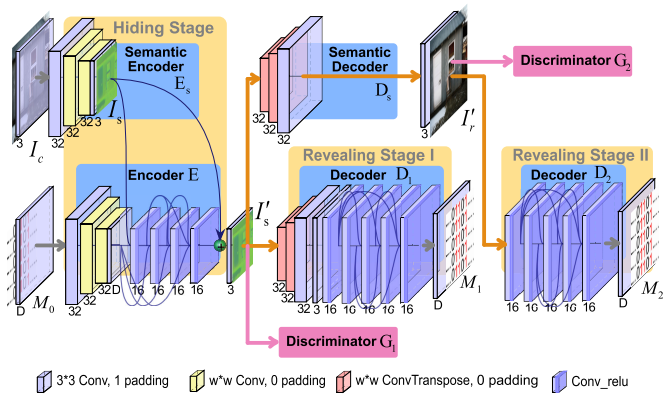

Fig. 3. The architecture of image semantic steganography.

where $M_1 \in \{0,1\}^{D\times H\times W}$. Stego reconstructed image I_r' is obtained by semantic decoder D_s based on stego semantic features I_s', i.e., $I_r' = D_s(I_s')$. At the second revealing stage, the decoder D_2 extracts the message M_2 from the stego reconstructed image I_r':

$$M_2 = D_2(I_r') \tag{8}$$

where $M_2 \in \{0,1\}^{D\times H\times W}$.

The architecture of image semantic steganography is shown in Fig. 3. To improve security, we used Generating Adversarial Networks (GANs) to train our model, the discriminator G_1 and G_2 are set to Sr-Net [34] and the encoder network and decoder network are all convolutional networks. The semantic encoder E_s has two $w\times w$ convolution layers (with 0 padding), shown in yellow, and the semantic decoder D_s has two $w\times w$ transposed convolution layers (with 0 padding), shown in red. Assuming that the height and width of the image are equal and the number of channels is equal before and after compression, i.e., $H = W$ and $C = C'$, the convolutional kernel size w is determined by $w = \lceil H/2\times(1-\sqrt{CR_0})\rceil$, where CR_0 is the compression rate and the symbol $\lceil\cdot\rceil$ denotes upward rounding. Encoder E, decoder D_1, and decoder D_2 are all based on the DenseNet [35] structure, which effectively enhances the features propagation and fully utilizes each layer features.

Note that the semantic features need to be quantized in the transmission channel. In order to standardize the results for security detection, the features are quantized to the unsigned 8-bit integers with the following equations:

$$I_s^{\max} = [\max(|I_s|)], \tag{9a}$$

$$I_s = \left[\left(\frac{1}{2}\cdot\frac{I_s}{I_s^{\max}}+\frac{1}{2}\right)\cdot 255\right] \bmod 256, \tag{9b}$$

$$I_s' = \left[\left(\frac{1}{2}\cdot\frac{I_s'}{I_s^{\max}}+\frac{1}{2}\right)\cdot 255\right] \bmod 256, \tag{9c}$$

$$I_s, I_s' \in \mathbb{Z}_{[0,255]} \tag{9d}$$

and, after I_s and I_s' are transmitted through the channel, they need to be inverse quantized before feeding into the semantic decoder D_s or decoder D_1.

B. Loss Function

The goal of the image semantic steganography framework is not only to accomplish the steganography task, but also to avoid interfering with the SC task. The steganography task centers on balancing steganographic security and extraction accuracy, while the SC task focuses on balancing reconstruction and AI task, as defined in Equation 3 of Section II-A. These two tasks are interrelated and have a close influence on each other. Both the hiding stage and the first revealing stage depend on the features extracted by the semantic encoder. Additionally, the second revealing stage depends on the image reconstructed by the semantic decoder. Conversely, stego semantic features also affect subsequent image reconstruction and AI task. The small redundancy of the compressed semantic features also brings challenges for steganography.

Following this, we divide the image semantic steganography into three distortions: extraction distortion, steganographic distortion and SC distortion. The extraction distortion of the two revealing stages is calculated by Binary Cross Entropy (BCE) between message M_0 with M_1 and M_2, respectively:

$$Loss_1 = \mathrm{BCE}(M_0, M_1), \tag{10a}$$

$$Loss_2 = \mathrm{BCE}(M_0, M_2) \tag{10b}$$

To ensure security during the revealing stage, steganographic distortion is calculated based on three aspects: pixel-level distortion and semantic-level distortion between the cover feature I_s and stego feature I_s', and pixel-level distortion between the cover reconstructed image I_r and stego reconstructed image I_r'.

$$Loss_3 = \mathrm{MSE}(I_s', I_s), \tag{11a}$$

$$Loss_4 = \mathrm{Loss}_{\cos}(I_s', I_s), \tag{11b}$$

$$Loss_5 = \mathrm{MSE}(I_r', I_r) \tag{11c}$$

where, the pixel-level distortion is calculated based on MSE$(\cdot)$. And the semantic-level distortion is calculated based on cosine similarity [36]:

$$\mathrm{Loss}_{\cos}(I_s', I_s) = 1 - \frac{I_s'\cdot I_s}{\|I_s'\|\,\|I_s\|} \tag{12}$$

The SC distortion is the same as the Equation 3 of section II-A:

$$Loss_6 = 0.99\cdot\mathrm{MSE}(I_r', I_c) + 0.01\cdot\mathrm{CE}(label_r', label_c) \tag{13}$$

where, the difference is that I_r' is the stego reconstructed image and $label_r'$ is the predicted labels of I_r'. To balance the order of magnitude between MSE and CE, λ is set to 0.99.

Moreover, the goal of the discriminator G_1 is to maximize the $G_1(I_s), G_1(I_s')$ score difference, while the goal of adversarial discriminator G_1 is to minimize this difference. For example, the I_s and I_s' are designated as positive and negative samples, respectively, and the predicted labeling loss is calculated via CE$(\cdot)$. In training mode of discriminator G_1,

Algorithm 1 Image Semantic Steganography Training Algorithm

Input: cover image set I_C, message M_0, compression rate CR_0

Output: cover semantic features I_s, stego semantic features I'_s, cover reconstructed image I_r, stego reconstructed image I'_c, extracted message M_1 and M_2

Independently pre-trained AI task, e.g., $Classifier$.
while the training stop condition is not met **do**
  Train G_1, ***Eval*** E_s, E:
  for each batch $I_c \in I_C$ **do**
    $I_s = E_s(I_c)$, $I'_s = E(I_s, M_0)$
    Loss: $Loss_G = \text{CE}(label_{pred}, label_{truth})$
    optimize G_1 to minimize Loss
  end for
  Train G_2, ***Eval*** E_s, D_s, E:
  for each batch $I_c \in I_C$ **do**
    $I_s = E_s(I_c)$, $I'_s = E(I_s, M_0)$
    $I_r = D_s(I_s)$, $I'_r = D_s(I'_s)$
    Loss: $Loss_G = \text{CE}(label_{pred}, label_{truth})$
    optimize G_2 to minimize Loss
  end for
  Train E_s, D_s, E, D_1, D_2***Eval***G_1, G_2, $Classifier$:
  for each batch $I_c, label_c \in I_C$ **do**
    $I_s = E_s(I_c)$, $I'_s = E(I_s, M_0)$
    $I_r = D_s(I_s)$, $I'_r = D_s(I'_s)$
    $M_1 = D_1(I'_s)$, $M_2 = D_2(I'_r)$
    Loss: FAMO$(\sum_{i=1}^{8} \lambda_i \cdot loss_i)$
    optimize $E_s + D_s + E + D_1 + D_2$ to minimize Loss
  end for
end while

the loss is

$$Loss_G = \text{CE}(label_{pred}, label_{truth}), \tag{14}$$

while in adversarial mode, the loss is

$$Loss_7 = \frac{1}{loss_{G_1} + \varepsilon} \tag{15}$$

where, to avoid division by zero, a small positive constant, $\varepsilon = 10^{-6}$, is added to the denominator. Similarly, the training loss for discriminator G_2 is predicted labeling loss of I_r and I'_r as positive and negative samples, and the adversarial loss is

$$Loss_8 = \frac{1}{loss_{G_2} + \varepsilon}. \tag{16}$$

Finally, the total loss function is a weighted sum between these distortions:

$$\begin{aligned} \min \quad & \sum_{i=1}^{8} \lambda_i \cdot loss_i \\ \text{s.t.} \quad & CR = CR_0 \end{aligned} \tag{17}$$

where λ_i is the weight coefficient corresponding to $loss_i$.

C. Training

The training algorithm for the image semantic steganography is described in Algorithm 1. The semantic encoder E_s, semantic decoder D_s, encoder E, decoder D_1, and decoder D_2 are trained jointly. The AI task network is externally trained initially as the guide, and the GAN model is employed for adversarial training. During training and testing, the message M_0 is generated randomly and is different for each batch. This ensures that the model is exposed to a wide variety of messages, which enhances the model's generalization and reduces the risk of overfitting to specific message. In order to obtain the optimal loss decrease approach, we introduce Fast Adaptive Multitask Optimization (FAMO) [37] to address multitask learning, which is a dynamic weighting method that decreases task losses in a balanced way. The model was trained for 150 epochs with an initial learning rate of 1e-3, which was reduced by a factor of 0.1 every 30 epochs until it reached 1e-5.

IV. Experimental Results and Analysis

In this section, we provide a detailed description of the experimental settings, including datasets, AI tasks, evaluation metrics, and so on. In order to verify the effectiveness and security of the image semantic steganography framework, image classification and target detection are selected as the AI tasks in this paper.

A. Settings

AI Tasks: When image classification is used as the AI task to guide training, the classifier is trained based on GoogleNet [38] with 200 epochs and the classification accuracy (ACC) reaches 88.28%. When target detection is used as the AI task to guide training, the detector is trained based on Yolov5 [39] with 100 epochs and the mean average precision with IOU = 50 (mAP50) reaches 70.22%.

Dataset: For image classification, the CIFAR-10 [40] dataset is used as the training and testing set for the experiments. This dataset contains 60,000 32×32 color images in 10 classes, 50,000 for training and 10,000 for testing. In addition, the images are reshaped to 64×64 for the experiments. For target detection, the VOC2007 [41] is used as the training and testing set. This dataset contains 7,462 color images in 20 classes, 4,952 for training and 2,510 for testing. And the images are reshaped to 256×256 with padding for the experiments.

Compression Rate (CR): This experiment aims to analyze the performance of semantic steganography, and the CR is not discussed. As this model is a basic model, a lower compression rate will affect the image reconstruction and message extraction. Thus, $CR_0 = 85\%$ for this experiment.

Payload: As $M_0 \in \{0, 1\}^{D \times H \times W}$, it matches the size of the input cover image. The channel number D determines the payload of the generated stego semantic features, calculated using bits per pixel (bpp).

Evaluation Metrics:

Revealing Stage I: The similarity between the cover semantic features I_s and the stego features I'_s is evaluated by

PSNR(I_s, I'_s) and SSIM(I_s, I'_s) in Equation 4 and 5, denoted as PSNR_1 and SSIM_1. Bit accuracy is denoted as Bit ACC_1 and is calculated as:

$$\text{Bit ACC}(M_1, M_0) = \frac{\sum M_1 \odot M_0}{len(M_0)} \tag{18}$$

where $\odot$ denotes the XNOR operation and $len(M_0)$ denotes the length of message M_0.

Revealing Stage II: Same as stage I, the similarity between the cover reconstructed image I_r and the stego image I'_r is evaluated by PSNR(I_r, I'_r) and SSIM(I_r, I'_r) in Equation 4 and 5, denoted as PSNR_2 and SSIM_2. Bit accuracy is evaluated by Bit ACC(M_2, M_0) in Equation 18 and denoted as Bit ACC_2.

Impact on SC: To analyze the impact of steganography on SC, the similarity between the stego reconstructed image I'_r and the original image I_c can be evaluated by PSNR(I'_r, I_c) and SSIM(I'_r, I_c) in Equation 4 and 5. When image classification is the AI task, the impact on the AI task can be assessed by the classification accuracy gap denoted as ACC Gap:

$$\begin{aligned}\text{ACC Gap} = \text{ACC}(&Classifier(I_r), label_c) \\ &- \text{ACC}(Classifier(I'_r), label_c)\end{aligned} \tag{19}$$

where the $Classifier$ is pre-trained, $label_c$ is the true label of the original image I_c, and the predicted labels $Classifier(I_r)$ and $Classifier(I'_r)$ of the reconstructed images I_r and I'_r are obtained by the $Classifier$. When target detection is the AI task, the impact on the AI task can be assessed by the mean average precision gap with IOU = 50 denoted as mAP50 Gap:

$$\begin{aligned}\text{mAP50 Gap} = \text{mAP50}(&Detector(I_r), label_c) \\ &- \text{mAP50}(Detector(I'_r), label_c)\end{aligned} \tag{20}$$

where the $Detector$ is pre-trained, $label_c$ is the true label of the original image I_c, and the predicted labels $Detector(I_r)$ and $Detector(I'_r)$ of the reconstructed images I_r and I'_r are obtained by the $Detector$.

B. Performance

In order to verify the effectiveness and performance of the proposed framework, we test the metrics as well as the resistance to steganalysis under different AI tasks, respectively. Table II shows the performance metrics of different payloads under the image classification task. The extraction accuracy (Bit ACC) and image quality (PSNR and SSIM) of the two revealing stages increase with the payload. Notably, the extraction accuracy of both stages reaches 99.98% at 0.1bpp payload, while it decreases to 98.60% and 97.73% at 1bpp payload. This indicates that under the image classification task, the increase in payload has little influence on the extraction of both stages, but the stage II is relatively affected compared to the stage I. This is due to the fact that when reconstructing the image, some of the embedded information is mapped to the details or colors of the original image, which adds the difficulty to the extraction.

When the payload is 0.1bpp, the PSNR and SSIM of the two stages are 53.94dB, 99.65%, 45.22dB, 99.22% respectively. And when the payload is 1bpp, the PSNR and SSIM of the two stages just decrease to 50.56dB, 99.32%, 41.57dB and 98.37%. We observe that the SSIM for the two stages is similar, but there is a significant difference in the PSNR. The discrepancy is due to the fact that image reconstruction amplifies differences from stage I, even though it improves color and details. This leads to approximately 8% decrease in PSNR on average. Fig. 4 shows the corresponding generated cover and stego images for 1bpp payload. The difference between cover and stego images is subtle to the naked eye. In addition, the semantic features are not only smaller in size but also significantly compress the color information compared to the original image.

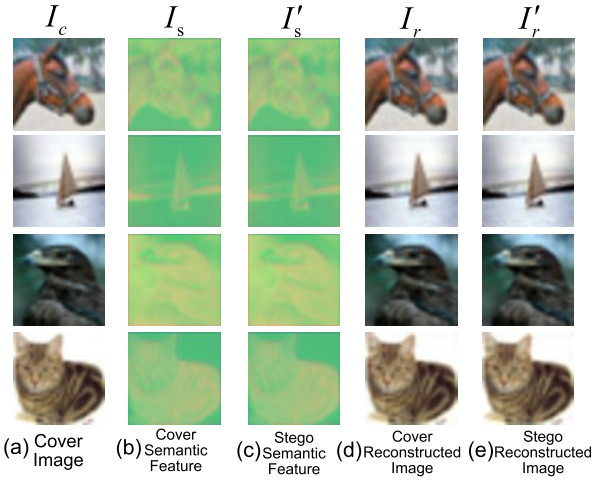

Fig. 4. The visualization of stego and cover images with 1bpp payload under the image classification task. (a) Cover image. (b) Cover semantic feature. (c) Stego semantic feature. (d) Cover reconstructed image. (e) Stego reconstructed image.

The impact of semantic steganography on the SC process is assessed by its influence on the subsequent AI tasks and the reconstructed image quality, as shown in Table II. For the image classification task, the requirements for image reconstruction quality are relatively low, given the simplicity of the task and the small image size (64×64). Taking the payload of 1 bpp as an example, although the PSNR and SSIM of the reconstructed image are only 37.23 dB and 97.39%, the gap in classification accuracy is very small, only 0.05%.

Similarly, Table III shows the performance metrics of different payloads under the target detection task. The extraction accuracy (Bit ACC) of the two revealing stages increases with the payload, while the image quality (PSNR and SSIM) remains almost unchanged. At 0.1bpp payload, the Bit ACC of the two stages reaches 99.97%, while at 1bpp payload, the extraction accuracy decreases to 93.00% and 92.23%, respectively. Compared to the image classification task, the increase in payload has a greater impact on the extraction of both stages under the target detection task. This is because during image reconstruction, the target detection task depends on the larger image size (256×256), leading to more embedded information mapped to the image color and details for security, which affects the extraction accuracy. Moreover, the extraction of stage II is more affected compared to stage I.

When the payload is 0.1bpp, the PSNR and SSIM of these two stages are 53.99dB, 99.70%, 45.62dB and 98.78%,

TABLE II
THE METRIC VALUES OF DIFFERENT PAYLOADS UNDER THE IMAGE CLASSIFICATION TASK

AI task	Dataset	Payload (bpp)	Revealing Stage I			Revealing Stage II			Impact on SC		
			Bit ACC_1 (%)↑	$PSNR_1$ (dB)↑	$SSIM_1$ (%)↑	Bit ACC_2 (%)↑	$PSNR_2$ (dB)↑	$SSIM_2$ (%)↑	ACC Gap (%)↓	PSNR (dB)↑	SSIM (%)↑
Image Classification	CIFAR-10	0.1	**99.98**	**53.94**	**99.65**	**99.98**	**45.22**	**99.22**	**0.00**	**38.14**	98.14
		0.2	**99.98**	52.64	99.57	99.95	44.15	99.04	0.02	37.83	98.14
		0.3	99.90	52.52	99.56	99.80	43.99	99.01	0.17	37.85	98.14
		0.4	99.67	52.51	99.56	99.36	44.00	99.01	0.05	37.84	98.13
		0.5	98.99	51.67	99.47	98.32	43.86	99.00	0.12	37.81	98.13
		0.6	98.32	52.49	99.56	97.55	44.02	99.02	0.06	37.85	98.14
		0.7	98.89	51.60	99.46	97.94	43.76	98.98	**0.00**	37.82	**98.98**
		0.8	98.60	51.60	99.46	97.48	43.76	98.98	**0.00**	37.82	98.13
		0.9	98.41	50.79	99.46	97.09	42.67	98.98	**0.00**	37.82	98.13
		1.0	98.60	50.56	99.32	97.73	41.57	98.37	0.05	37.23	97.39

TABLE III
THE METRIC VALUES OF DIFFERENT PAYLOADS UNDER THE TARGET DETECTION TASK

AI task	Dataset	Payload (bpp)	Revealing Stage I			Revealing Stage II			Impact on SC		
			Bit ACC_1 (%)↑	$PSNR_1$ (dB)↑	$SSIM_1$ (%)↑	Bit ACC_2 (%)↑	$PSNR_2$ (dB)↑	$SSIM_2$ (%)↑	mAP50 Gap (%)↓	PSNR (dB)↑	SSIM (%)↑
Target Detection	VOC2007	0.1	**99.97**	53.99	**99.70**	**99.97**	45.62	98.78	0.49	37.57	97.33
		0.2	99.80	53.99	**99.70**	99.78	45.65	98.79	0.52	37.58	97.34
		0.3	99.33	**54.00**	**99.70**	99.26	45.62	98.78	0.45	37.57	97.33
		0.4	98.50	**54.00**	**99.70**	98.35	45.63	98.79	0.57	37.57	97.33
		0.5	97.28	53.99	99.69	97.00	**45.69**	**98.80**	0.51	**37.59**	**97.36**
		0.6	96.16	53.99	**99.70**	95.90	45.62	98.79	0.50	37.58	97.34
		0.7	94.91	**54.00**	**99.70**	94.60	45.67	**98.80**	0.44	37.58	97.34
		0.8	94.00	**54.00**	**99.70**	93.57	45.67	98.79	**0.42**	37.58	97.34
		0.9	93.52	53.99	**99.70**	92.98	45.62	98.78	0.47	37.58	97.34
		1.0	93.00	53.97	**99.70**	92.23	45.65	**98.80**	0.45	**37.59**	97.35

TABLE IV
THE METRIC VALUES OF DIFFERENT METHODS AFTER SC

Payload (bpp)	Dataset	Method	PSNR of SC (dB)	SSIM of SC (%)	Bit ACC after SC (%)↑	Communication
0.1	CIFAR-10	Image robust steganography [24], [27]–[30]	38.89	98.05	51.23	✘
		Our framework	38.14	98.14	**99.98**	✔
	VOC2007	Image robust steganography [24], [27]–[30]	38.01	97.89	50.07	✘
		Our framework	37.57	97.33	**99.97**	✔

respectively. And when the payload is 1bpp, the PSNR and SSIM of the two stages are 53.97dB, 99.70%, 45.65dB and 98.80%, respectively. It is noticed that the increase in payload does not affect the PSNR and SSIM. This is because image reconstruction amplifies the differences from stage I, and the large image size necessitates higher similarity for security, which consequently results in a lower Bit ACC. In addition, compared to stage I, the PSNR of stage II decreases by about 8% on average, which is the same loss as the image classification task. Fig. 5 shows the cover and stego images at 1bpp payload, which are difficult to distinguish them from the naked eyes. Compared to the original cover image, the semantic features are not only smaller in size and compress the color information, but also retain the details (as shown in the red box).

Like image classification task, the impact of semantic steganography on the SC process is assessed by its influence on the subsequent AI task and the reconstructed image quality, as shown in Table III. For the target detection task, the task requires larger image sizes (256×256), and thus requires a relatively high quality of image reconstruction. However, the pre-embedded information leads to the PSNR and SSIM of the reconstructed image to be only 37.57 dB and 97.33%, respectively, thus the gap of map50 has about 0.50%. That is, the map50 of the reconstructed image without embedding is 70.22%, and the mAP50 after embedding drops to 69.72%, which is an unavoidable loss within the acceptable range for practical applications.

In summary, the image semantic steganography successfully extracts information correctly after SC, while the image robust steganography fails, as demonstrated in Table IV. Furthermore, our framework ensures high extraction accuracy and similarity in the semantic channel and after reconstruction under multiple AI tasks, thus guaranteeing effective communication and security.

C. Steganalysis Resistance

To assess the security of the proposed framework, we apply two advanced steganalysis methods, Sr-Net [34] and Xu-Net [42], to evaluate the two revealing stages at 1bpp payload. For image classification and target detection tasks, we use 2000 CIFAR-10 sample pairs and 800 VOC2007 sample pairs for training, respectively. In this case, sample pairs are shuffled to better align with observed sample leakage

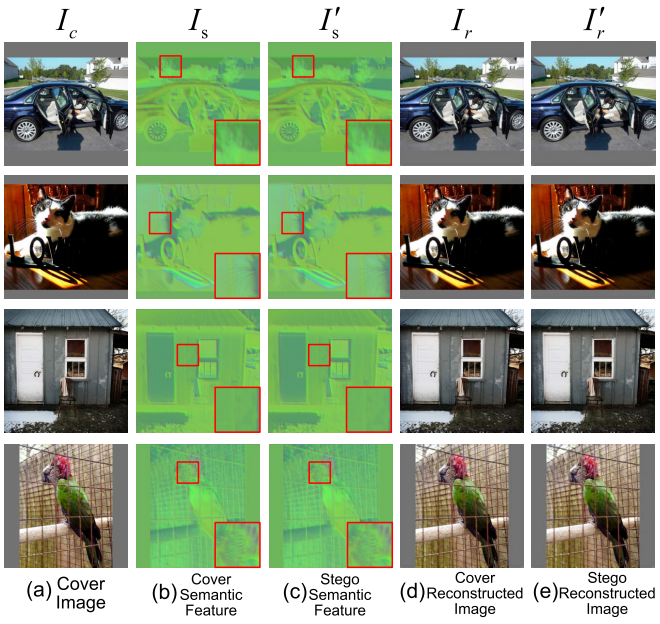

Fig. 5. The visualization of stego and cover images with 1bpp payload under the target detection task. (a) Cover image. (b) Cover semantic feature. (c) Stego semantic feature. (d) Cover reconstructed image. (e) Stego reconstructed image.

TABLE V
DETECTION ACCURACY OF STEGANALYSIS UNDER THE IMAGE CLASSIFICATION TASK

AI task	Steganalysis	Detection ACC (%) → 50.00	
		Revealing Stage I	Revealing Stage II
Image Classification	Sr-Net [34]	50.00±0.89	50.00±0.17
	Xu-Net [42]	50.00±0.19	50.00±0.23

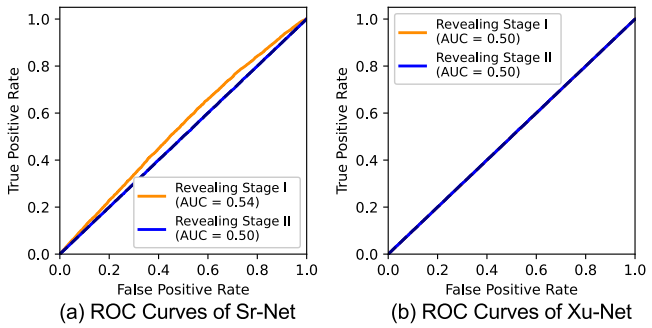

Fig. 6. The ROC curves generated by Sr-Net and Xu-Net on the image classification with 1bpp payload. (a) ROC curves of Sr-Net. (b) ROC curves of Xu-Net.

patterns. Both analyzers undergo training for 150 epochs, with 10 experimental iterations conducted to minimize experimental uncertainties.

Table V shows the detection accuracies of the two stages of image classification. The mean values of both are 50.00% and the maximum standard deviation is 0.89%. Fig. 6 shows the receiver operating characteristic (ROC) curves of the two analyzers in different stages, and the AUC is 0.5 for all of them. The above results indicate that 2000 leaked sample pairs are not enough for the analyzers to learn effective classification features, and the detection results are almost like random

TABLE VI
DETECTION ACCURACY OF STEGANALYSIS UNDER THE TARGET DETECTION TASK

AI task	Steganalysis	Detection ACC (%) → 50.00	
		Revealing Stage I	Revealing Stage II
Target Detection	Sr-Net [34]	50.00±0.32	50.46±4.09
	Xu-Net [42]	50.00±2.31	50.00±0.32

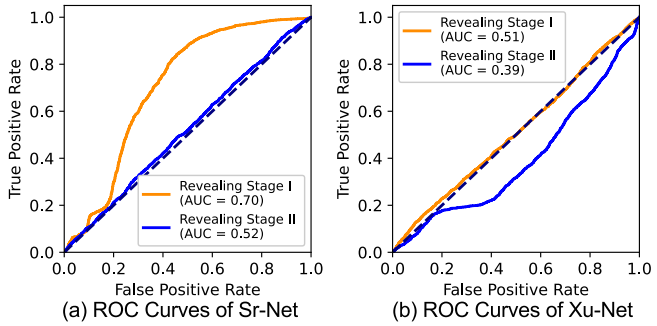

Fig. 7. The ROC curves generated by Sr-Net and Xu-Net on the target detection with 1bpp payload. (a) ROC curves of Sr-Net. (b) ROC curves of Xu-Net.

guesses. In addition, the resistance to steganalysis is similar for both stages, but the PSNR of stage 1 is significantly higher than that of stage 2, i.e., 50.56 dB > 41.57 dB. This suggests that while the semantic decoder causes stage II amplify the difference in PSNR, it is also the involvement of the decoder which hinders the analyzer from learning useful information efficiently.

Table VI shows the detection accuracies for the two stages of target detection. The mean value for both is around 50.00%, but the maximum standard deviation is 4.09%. Figure 7 shows the ROC curves of the two analyzers at different stages. For Sr-Net, the AUC is 0.7 for stage I and 0.52 for stage II, while for Xu-Net, the AUC is 0.51 for stage I and 0.39 for stage II. The above results show that 800 pairs of leakage samples have enabled Sr-Net to learn some of the effective classification features of stage I. But the mean value of detection accuracy as well as the standard deviation (50.00 ± 0.32) indicate that it is still unable to discriminate the stego features. And Sr-Net's detection results for the stage II are almost random guesses, which is also shown in the stage II of Xu-Net. Moreover, the AUC of the stage II of Xu-Net is 0.39, which is lower than 0.5, i.e., the classification performance is even worse than random guess, and the learned features are counterproductive. Unlike the image classification task, the two stages of target detection have different resistance to steganalysis, stage I being lower than stage II. This further indicates that the involvement of the semantic decoder increases the security of stage II. Furthermore, compared to small-size images, it is easier for steganalysis to extract more information related to classification on large-size images. Thus, the target detection task requires higher security than image classification.

In summary, our framework is resistant to steganalysis detection in the semantic channel as well as after image reconstruction.

V. Conclusion

This paper introduces an AI-based image semantic steganography framework to address the problem that robust steganography cannot extract correctly after semantic communication. Our framework, based on GANs, proposes a two-stage extraction process for added flexibility. During hiding, message is embedded into semantic features, enhancing security by incorporating both pixel and semantic-level distortions. Experimental results show that our framework exhibits good performance under different AI tasks and has resistance to steganalysis.

However, despite these advancements, our framework remains a basic model with limited compression capacity and payload. These constraints highlight several critical challenges that need to be addressed, including enhancing robustness against noise attacks, increasing compression capacity while maintaining AI task performance, improving embedding capacity to resolve issues of limited semantic feature redundancy, and so on. Addressing these challenges is essential for advancing the field of semantic steganography.

References

[1] H. Xie, Z. Qin, G. Y. Li, and B.-H. Juang, "Deep learning enabled semantic communication systems," *IEEE Trans. Signal Process.*, vol. 69, pp. 2663–2675, 2021. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9398576

[2] X. Luo, H. Chen, and Q. Guo, "Semantic communications: Overview, open issues, and future research directions," *IEEE Wireless Commun.*, vol. 29, no. 1, pp. 210–219, Feb. 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9679803

[3] B. Tang, L. Huang, Q. Li, A. Pandharipande, and X. Ge, "Cooperative semantic communication with on-demand semantic forwarding," *IEEE Open J. Commun. Soc.*, vol. 5, pp. 349–363, 2024.

[4] W. Yu and J. Zhao, "Semantic communications, semantic edge computing, and semantic caching with applications to the metaverse and 6G mobile networks," in *Proc. IEEE 43rd Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jul. 2023, pp. 983–984.

[5] H. Xie, Z. Qin, and G. Y. Li, "Semantic communication with memory," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 8, pp. 2658–2669, Aug. 2023. [Online]. Available: https://ieeexplore.ieee.org/document/10159023

[6] R. Cheng, K. Liu, N. Wu, and B. Han, "Enriching telepresence with semantic-driven holographic communication," in *Proc. 22nd ACM Workshop Hot Topics Netw.*, New York, NY, USA, Nov. 2023, pp. 147–156, doi: 10.1145/3626111.3628184.

[7] T. Ren and H. Wu, "Asymmetric semantic communication system based on diffusion model in IoT," in *Proc. IEEE 23rd Int. Conf. Commun. Technol. (ICCT)*, Oct. 2023, pp. 1–6.

[8] M. Li, Q. Xiong, and L. Yuan, "A network interconnection system and framework based on semantic communication," in *Proc. VI Int. Conf. Netw., Commun. Comput.*, New York, NY, USA, 2017, pp. 187–190, doi: 10.1145/3171592.3171617.

[9] M. Chen, M. Liu, W. Wang, H. Dou, and L. Wang, "Cross-modal semantic communications in 6G," in *Proc. IEEE/CIC Int. Conf. Commun. China (ICCC)*, Dalian, China, Aug. 2023, pp. 1–6.

[10] H. Zhang, S. Shao, M. Tao, X. Bi, and K. B. Letaief, "Deep learning-enabled semantic communication systems with task-unaware transmitter and dynamic data," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 1, pp. 170–185, Jan. 2023. [Online]. Available: https://ieeexplore.ieee.org/document/9953099

[11] J. Strecker, K. García, K. Bektaş, S. Mayer, and G. Ramanathan, "SOCRAR: Semantic OCR through augmented reality," in *Proc. 12th Int. Conf. Internet Things*, New York, NY, USA, 2023, pp. 25–32, doi: 10.1145/3567445.3567453.

[12] X. Qian, F. He, X. Hu, T. Wang, A. Ipsita, and K. Ramani, "ScaLAR: Authoring semantically adaptive augmented reality experiences in virtual reality," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, New York, NY, USA, 2022, doi: 10.1145/3491102.3517665.

[13] L. Galteri, M. Bertini, L. Seidenari, T. Uricchio, and A. Del Bimbo, "Increasing video perceptual quality with GANs and semantic coding," in *Proc. 28th ACM Int. Conf. Multimedia*, New York, NY, USA, Oct. 2020, pp. 862–870, doi: 10.1145/3394171.3413508.

[14] B. Li, B. Chen, Z. Wang, S. Wang, and Y. Ye, "Semantic face compression for metaverse: A compact 3D descriptor based approach," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 9, pp. 8978–8982, Sep. 2024.

[15] Y. Li, J. Cai, Q. Zhou, and H. Lu, "Joint semantic-instance segmentation method for intelligent transportation system," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 12, pp. 15540–15547, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/9831011

[16] P. Zhang, S. Wang, M. Wang, J. Li, X. Wang, and S. Kwong, "Rethinking semantic image compression: Scalable representation with cross-modality transfer," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 8, pp. 4441–4445, Jul. 2023.

[17] M. Shen et al., "Secure semantic communications: Challenges, approaches, and opportunities," *IEEE Netw.*, vol. 38, no. 4, pp. 197–206, Jul. 2024. [Online]. Available: https://ieeexplore.ieee.org/document/10292907

[18] J. Fridrich, *Steganography in Digital Media: Principles, Algorithms, and Applications*, 1st ed. Cambridge, U.K.: Cambridge Univ. Press, 2009.

[19] V. Holub and J. Fridrich, "Digital image steganography using universal distortion," in *Proc. 1st ACM Workshop Inf. Hiding Multimedia Secur.*, New York, NY, USA, 2013, pp. 59–68, doi: 10.1145/2482513.2482514.

[20] D. Wang, G. Yang, Z. Guo, and J. Chen, "Enhancing adversarial embedding based image steganography via clustering modification directions," *ACM Trans. Multimedia Comput., Commun. Appl.*, vol. 20, no. 1, pp. 1–20, Sep. 2023, doi: 10.1145/3603377.

[21] H. Gilbert, M. Sandborn, D. C. Schmidt, J. Spencer-Smith, and J. White, "Semantic compression with large language models," in *Proc. 10th Int. Conf. Social Netw. Anal., Manage. Secur. (SNAMS)*, Nov. 2023, pp. 1–8.

[22] Q. Liu, J. Ni, and X. Hu, "Robust image steganography against general scaling attacks," in *Proc. 31st ACM Int. Conf. Multimedia*, New York, NY, USA, 2023, pp. 8233–8241, doi: 10.1145/3581783.3612267.

[23] K. Zeng, K. Chen, W. Zhang, Y. Wang, and N. Yu, "Robust steganography for high quality images," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 9, pp. 4893–4906, Sep. 2023.

[24] Y. Zhang, X. Luo, J. Wang, Y. Guo, and F. Liu, "Image robust adaptive steganography adapted to lossy channels in open social networks," *Inf. Sci.*, vol. 564, pp. 306–326, Jul. 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:233548503

[25] Y. Zhang, X. Luo, C. Yang, D. Ye, and F. Liu, "A framework of adaptive steganography resisting JPEG compression and detection," *Secur. Commun. Netw.*, vol. 9, no. 15, pp. 2957–2971, Oct. 2016, doi: 10.1002/sec.1502.

[26] K. Zeng, K. Chen, W. Zhang, Y. Wang, and N. Yu, "Improving robust adaptive steganography via minimizing channel errors," *Signal Process.*, vol. 195, Jun. 2022, Art. no. 108498. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0165168422000457

[27] Y. Zhang, X. Zhu, C. Qin, C. Yang, and X. Luo, "Dither modulation based adaptive steganography resisting JPEG compression and statistic detection," *Multimedia Tools Appl.*, vol. 77, no. 14, pp. 17913–17935, Jul. 2018, doi: 10.1007/s11042-017-4506-3.

[28] X. Yu, K. Chen, Y. Wang, W. Li, W. Zhang, and N. Yu, "Robust adaptive steganography based on generalized dither modulation and expanded embedding domain," *Signal Process.*, vol. 168, Mar. 2020, Art. no. 107343, doi: 10.1016/j.sigpro.2019.107343.

[29] W. Lu, J. Zhang, X. Zhao, W. Zhang, and J. Huang, "Secure robust JPEG steganography based on autoencoder with adaptive BCH encoding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 7, pp. 2909–2922, Jul. 2021.

[30] Y. Zhang, X. Luo, Y. Guo, C. Qin, and F. Liu, "Multiple robustness enhancements for image adaptive steganography in lossy channels," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 8, pp. 2750–2764, Aug. 2020.

[31] C. Liu, C. Guo, S. Wang, Y. Li, and D. Hu, "Task-oriented semantic communication based on semantic triplets," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Mar. 2023, pp. 1–6.

[32] W. Tong, F. Liu, Z. Sun, Y. Yang, and C. Guo, "Image semantic communications: An extended rate-distortion theory based scheme," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Dec. 2022, pp. 1723–1728.

[33] P. Bas, T. Filler, and T. Pevný, "'Break our steganographic system': The ins and outs of organizing BOSS," in *Information Hiding*, T. Filler, T. Pevný, S. Craver, and A. Ker, Eds. Berlin, Germany: Springer, 2011, pp. 59–70.

[34] M. Boroumand, M. Chen, and J. Fridrich, "Deep residual network for steganalysis of digital images," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 5, pp. 1181–1193, May 2019, doi: 10.1109/TIFS.2018.2871749.

[35] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2261–2269.

[36] S. Czolbe, O. Krause, and A. Feragen, "DeepSim: Semantic similarity metrics for learned image registration," in *Proc. Med. Imag. Deep Learn.*, 2021, pp. 105–118.

[37] B. Liu, Y. Feng, P. Stone, and Q. Liu, "FAMO: Fast adaptive multitask optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 36, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds. Red Hook, NY, USA: Curran Associates, 2023, pp. 57226–57243.

[38] C. Szegedy et al., "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9.

[39] G. Jocher et al., "Ultralytics/YOLOv5: V6.2—YOLOv5 classification models, Apple M1, Reproducibility, ClearML and Deci.ai integrations," Ultralytics, MD, USA, Tech. Rep., Aug. 2022, doi: 10.5281/zenodo.7002879.

[40] A. Krizhevsky, "Learning multiple layers of features from tiny images," in *Handbook of Systemic Autoimmune Diseases*, 2009, pp. 32–33. [Online]. Available: https://api.semanticscholar.org/CorpusID:18268744

[41] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. *The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results*. Accessed: Jun. 7, 2007. [Online]. Available: http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html

[42] G. Xu, H.-Z. Wu, and Y.-Q. Shi, "Structural design of convolutional neural networks for steganalysis," *IEEE Signal Process. Lett.*, vol. 23, no. 5, pp. 708–712, May 2016.

Yanhao Huo received the M.S. degree from the School of Cyberspace Security, Guangzhou University, China, in 2023. She is currently pursuing the Ph.D. degree with Jinan University, Guangzhou. Her current research interests include image semantic steganography and watermarking.

Shijun Xiang received the B.S. degree from Chang'an University in 1997, the M.S. degree from Guizhou University in 2000, and the Ph.D. degree from Sun Yat-sen University, China, in 2006. From 2006 to 2007, he was a Post-Doctoral Researcher with Korea University, Seoul, South Korea. He is currently a Full Professor with the College of Information Science and Technology, Jinan University, Guangzhou, China. He has authored or co-authored over 100 peer-reviewed articles, including IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, and IEEE TRANSACTIONS ON MULTIMEDIA. His current research interests include robust watermarking, reversible data hiding, and face spoofing.

Xiangyang Luo received the B.S., M.S., and Ph.D. degrees from the State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou, China, in 2001, 2004, and 2010, respectively. He is currently a Professor with the State Key Laboratory of Mathematical Engineering and Advanced Computing. He has authored or co-authored more than 100 refereed international journals and conference papers. His research interests include image steganography and steganalysis technique.

Xinpeng Zhang (Member, IEEE) received the B.S. degree from Jilin University, China, in 1995, and the M.S. and Ph.D. degrees from Shanghai University in 2001 and 2004, respectively. He was a Visiting Scholar with The State University of New York at Binghamton from 2010 to 2011 and an experienced Researcher with Konstanz University, sponsored by the Alexander von Humboldt Foundation, from 2011 to 2012. Since 2004, he has been a Faculty Member with the School of Communication and Information Engineering, Shanghai University, where he is currently a full-time Professor. He is a Faculty Member with the School of Computer Science, Fudan University. He has published more than 200 research articles. His research interests include multimedia security, image processing, and digital forensics. He was an Associate Editor of IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY from 2014 to 2017.