Multilevel Strong Auxiliary Network for Enhancing Feature Representation to Protect Secret Images

Feng Chen®, Qinghua Xing, and Chengli Fan®

Abstract—Image data play an important role in the network information, however, some images containing sensitive or confidential information are easy to attract the attention of malicious attackers. On the basis of deep learning and data hiding technology, in this article, a novel hiding–revealing network is designed to protect these secret images. The sender uses the hiding network to conceal the secret image into an ordinary cover image and the receiver uses the revealing network to recover the secret image. Symmetrical shortcut connection is designed to improve both the hiding and the revealing performances without adding any parameters. Consider that some secret images may have complex spatial features, a multilevel strong auxiliary module is designed to enhance feature representation and boost the restoration quality of the secret image. Then, a lifetime is proposed to transform the image hiding task into a residual identity mapping, which reduces the difficulty of network learning and obviously improves the hiding performance. In addition, a mixed loss function is designed to further improve the perceptual quality of both the hidden image and the revealed image, which further completely eliminates the secret content in the residual image and ensures the hiding security. Experimental results demonstrate that compared with the state-of-the-art methods, our proposed method achieves the best performance in both hidden image synthesis and secret image restoration.

Index Terms—Data hiding technology, deep learning, image steganography, multilevel strong auxiliary (MLSA) module, residual identity mapping.

I. INTRODUCTION

With the proposal of Industry 4.0, the Internet of Things (IoT) and the artificial intelligence technology have entered a rapid development stage [1], [2]. Many intelligent devices are connected to the IoT and users can expediently collect, exchange, transmit, share, and analyze data on the IoT cloud. Massive image data are generated from Internet every day. However, there exists an inevitable security issue that some images such as the identification card images, bank card images, electronic diagnosis images related to personal privacy, and even some high-value remote sensing images that contain sensitive geographic location information and military information are vulnerable to illegal access and malicious attackers. Therefore, how to protect the secret images becomes an urgent issue.

Currently, the most commonly used technology to protect secret data is encryption technology [3]. Although the encrypted data are difficult to be cracked in a short time, its existence in the communication media is easy to attract the attention of attackers. In recent years, a new security technology namely digital steganography [4], also known as data hiding technology [5], has attracted wide attention in various fields. This technology aims to conceal secret data in public digital carrier and achieve the transmission without arousing suspicion of the third party. Therefore, steganography technology can not only encrypt the secret data like encryption technology but also ensure the encrypted data imperceptible. The digital carriers used for steganography include binary text, voice data, image, etc.; among them, the image is widely used because it has rich texture, edge, and color contrast to confuse the human visual system (HVS). Steganography using the image as carrier is usually called image steganography.

Up to now, image steganography methods can be divided into two categories, namely traditional image steganography methods and deep learning-based image steganography methods.

Among the traditional image steganography methods, the least significant bit (LSB) steganography method [6] is widely used because it can embed various forms of secret data, including text and image data. However, its simple and obvious designed rule makes the embedded secret information suffer from a severe risk of being deciphered. Then, the content-adaptive steganography method based on minimal embedding distortion is proposed, which tend to embed the messages into the high-frequency texture region of the image, where the embedding traces are less detectable. Representative examples of this method include S-UNIWARD [7] and HILL [8]. However, the theoretical relationship between the statistical characteristics and the distortion function is not to be taken into consideration. To this end, another statistics-based steganography method is proposed, which relies on the cover model to minimize the statistical distortion. The first method based on the statistical model is called HUGO [9], which uses weighted norm to define the difference between the feature vectors extracted from the cover image and the hidden image, so as to maximize the consistency of the high-order statistical
characteristics. And then, Sedighi et al. [10] proposed a Gaussian statistical model for the cover image pixels and achieved better hiding performance. Although high performance is obtained by the abovementioned traditional steganography methods, both the content-adaptive-based methods and the statistics-based methods are facing the following problems.

1) They are always employed to hide a small amount of text messages (<0.4 bpp), and the hiding capacity is small, where the hiding capacity is usually measured in bits per pixel (bpp). Therefore, they are not suitable for hiding large image data and the application prospect is greatly limited.

2) They always require domain knowledge and construct the distortion model by manually extracting features. However, manually designing the feature model is usually limited to low dimensions and cannot accurately describe the high-order statistical characteristics, which greatly limits the performance improvement. In addition, the process of artificial design is not only time-consuming and labor-consuming but also expensive.

Deep learning-based steganography frameworks usually use encoder to achieve information hiding, and then use decoder to extract secret information. Zhu et al. [11] proposed a framework called HiDDeN that can be applied in the fields of digital steganography and watermarking. And then, Zhang et al. [12] proposed a new framework called SteganOGAN and achieved high payloads of 4.4 bpp. Although the above methods get rid of the dependence on traditional manual features. They are still only suitable for hiding a small amount of text data. In order to achieve the goal of hiding the large image data, Kumar et al. [13] used convolutional neural networks to conceal a secret image into another cover image with minimum changes in the contents. To further improve the security, Sharma et al. [14] attempted to combine cryptography with the neural networks to hide an image inside another image. However, due to premature model design, the secret images restored by schemes [13], [14] are seriously damaged and have large distortions relative to the original secret images. Then, Rehman et al. [15] designed an encoder–decoder framework based on convolutional neural networks and effectively preserved the secret image’s integrity. However, the synthetic hidden image has poor quality. To further improve the hiding performance, Duan et al. [16] proposed a reversible steganography network based on the U-Net structure, which achieved perceptually pleasing performance on both the hidden image synthesis and the secret image restoration. However, the residual information between the hidden image and the original cover image, which largely determines the hiding security, is not demonstrated in their research. And then, Baluja [17] displayed the residual image obtained by the pixelwise difference between the hidden image and the original cover image, and proposed algorithms of image transformation to obfuscate the secret content in the residual image, thereby making the secret content less discernible. Although above deep learning-based steganography methods have achieved satisfying performance on the natural images, the performance is greatly reduced when they are applied to hiding the image with complex spatial features, such as the remote sensing image that usually contains multiscale targets and long-range spatial correlation.

Aiming at the characteristics that remote sensing images usually have multiscale targets, Chen et al. [18] tried to adopt multiscale convolution kernels and various advanced feature fusion strategies to improve the quality of both the hidden image and the secret image. However, the research on image hiding is still in infancy, especially for the images with complex spatial features, and the hiding performance has a lot of room for improvement, which is the motivation of this article.

Industry 4.0 was motivated by the German manufacturing strategy that focuses on big data analytics, artificial intelligence, cloud computing, and cyber-physical systems to achieve higher automation and data exchange in various devices. To get rid of the dependence on the traditional human labor, a novel end-to-end network based on data-driven is designed in this article, and it can automatically conceal the secret image data without any human intervention, which caters to the developing trend of automation, intellectualization, and information digitization in Industry 4.0. Similar to the process of encryption and decryption, the network consists of two parts, namely the hiding network and the revealing network. Communicators use the network for covert communication in the IoT environment without causing the suspicion of attackers. As shown in Fig. 1, the covert communication indicated by the green path can make the secret image data more secure than the direct transmission indicated by the red path. The sender uses the hiding network to embed the secret image into an ordinary cover image and synthesize a hidden image which looks the same as the cover image, thereby avoiding the surveillance of attackers. And then, the receiver receives the hidden image and reveals the secret image from the hidden image using the revealing network.

Previous studies have shown that the image hiding task still needs to solve two problems. One is to improve the restoration quality of the secret image on the premise of ensuring the visual security of the hidden image, another is to improve the hidden image’s antidetection ability on the premise of embedding large image data. The proposed method in this article obtains a satisfying performance on hiding the secret images with complex spatial features, it not only greatly improves the restoration quality of the secret image but also boosts the hidden image’s quality and ensure its visual security. The major contributions of this article are as follows.

1) We empirically analyze that blindly deepening the network will lose more image details, resulting in the
deterioration of hiding–revealing performances, so the network structure is elaborately constructed.

2) Consider that some secret images may have complex spatial features, such as the remote sensing image that is usually ultra-long-range imaging, a multilevel strong auxiliary (MLSA) module is designed to enhance feature representation and significantly improve the restoration quality of the secret image.

3) We propose a lifeline and verify its crucial role in the image hiding task. It can transform the image hiding task into a residual identity mapping, thus significantly improving the hiding performance.

4) We attempt to introduce structural similarity index (SSIM) to design a mixed loss function to further boost the perceptual quality of the hidden image and the revealed image, thus enhancing the imperceptibility of the secret content.

The rest of this article is organized as follows. Section II presents the detailed idea of our proposed method. In Section III, we describe the implementation details and show the experimental results. Finally, Section IV concludes this article.

II. PROPOSED METHOD

A. Overall Framework

In this part, we first present the overall framework of our proposed method and then make a thorough theoretical analysis of the specific components.

As shown in Fig. 2, the proposed network is composed of a hiding network and a revealing network. The cover image and the secret image are fed into the two branches of the hiding network together, through downsampling and upsampling operations, the size of the intermediate feature maps is reduced, thus reducing the memory occupation. And then, feature maps extracted by these two branches are highly fused by concatenation operation. Finally, the channel number of the feature maps is reduced through conv3 × 3 operation to generate a hidden image. The whole process of the revealing network is similar to that of the hiding network. It should be noted that the downsampling operation and upsampling operation are realized by convolution and deconvolution, respectively, with kernel size of 4 × 4 and stride of 2. And numbers 3, 64, 128, and 256 represent the channel numbers of feature maps.

Researches on convolutional neural network show that deeper network usually has a better performance for the tasks of image recognition and classification because deeper network can usually model more complex feature representations and learn more distinctive high-level features. However, our empirical analysis in this article shows that blindly deepening the network not only degrades the hiding–revealing performances but also dramatically increases the parameters, because a series of convolution operations are essentially similar to filters and the filters will cause image details loss, which explains why only six-layer convolution–deconvolution structure is designed for the network.

To compensate for the image details’ loss caused by the consecutive convolution operations, inspired by ResNet [19], we design the symmetrical shortcut connection to connect the convolution and deconvolution for passing more image details to the upper layers. The shortcut connection was initially proposed in ResNet to solve the gradient disappearing problem. It is achieved using elementwise addition and requires that the two feature maps should have the same dimensionality. After addition operation, the dimensionality of the fused feature maps remains unchanged. It should be emphasized that our proposed shortcut connection aims to preserve more low-level features to synthesize both the hidden and the revealed images’ structural details and it is different from the shortcut connection proposed in ResNet, in which the shortcut connection was considered from the perspective of optimization and used to solve the gradient disappearing problem caused by deep layers. Besides, the skip concatenation indicated by the purple arrow is further designed to directly pass the raw information of the cover image and secret image to the top layer, so as to synthesize the hidden and the revealed images’ content details. Different from the shortcut connection that is achieved using elementwise addition, skip concatenation is achieved through channel concatenation. It assumes that f_{m1} and f_{m2} are two different feature maps with c1, c2 channels, respectively. Then, they are concatenated into a feature vector with c1 + c2 channels.

To further improve the restoration quality of the secret image while ensuring the hidden image’s quality, the synthetic hidden image should appear as similar to the cover image as possible while also containing enough information for recovering the secret image. However, the shallow backbone network designed in Fig. 2 and the fixed size of convolution kernel greatly limit the feature extraction. To this end, an MLSA module is designed to enhance feature representation.
Consider that the ideal synthesis process from the cover image to the hidden image is equivalent to an identity mapping, the lifeline indicated by the red arrow is further proposed in Fig. 2 to transform the complex mapping into a residual identity mapping, thereby reducing the difficulty of network learning.

It should be noted that, the designed elements such as the skip concatenation and MLSA module are not used in the revealing network. Here gives the following two considerations. First, the revealed image is extracted from the hidden image and the hidden image is generated by the hiding network. Therefore, all the designed elements in the hiding network can also influence and promote the revealing performance, repeatedly adding these elements in the revealing network will dramatically increase the network parameters. Second, the revealing network extracts potential information related to the secret image from the hidden image to reconstruct the secret image. Applying these elements in the revealing network will retain more low-level features related to the hidden image, which is useless for the reconstruction of the secret image.

B. MLSA Module

Although the shallow backbone network is conducive to the preservation of more image details, it restricts the extraction of high-level semantic information and its fixed size of convolution kernel also greatly restricts the receptive field to capture multiscale information, long range contextual information, and global information, which is especially unfavorable to the feature extraction of the secret image with rich spatial information. In recent years, atrous/dilated convolution [20]–[22] became an important tool for feature extraction of the secret image with rich spatial information.

In [23], the authors proposed an auxiliary module of the backbone network to gather other distinctive information. The MLSA module (Fig. 4) is mainly composed of an HDC block, global average pooling (GAP) block, and upsampling blocks, and multilevel feature maps generated by the downsampling operations in Fig. 2 are connected in turn. The designed MLSA module mainly has two advantages as follows.

1) Different from the backbone network shown in Fig. 2, which is a bottom-to-top structure to extract increasingly abstract features, the MLSA module is a top-to-bottom structure that can gather other distinctive clues. Smaller scale feature maps usually contain higher level semantic information, by connecting multilevel feature maps in turn, multilevel semantic information can be fully fused and small scale feature maps containing higher level semantic information can guide the low-level feature maps to capture valuable information that are conducive to the restoration of the secret image. Therefore, the MLSA module can be regarded as an auxiliary module of the backbone network to gather other distinctive information.

2) It can enhance feature representation and effectively make up for the defect that the backbone network cannot comprehensively extract the features of the secret image with rich spatial features. Apart from the HDC block that can extract multiscale features through different dilation rates and can enlarge the receptive field to capture surrounding spatial correlation and long-range spatial correlation of the image, the GAP block is also applied as a supplementary block to extract global contextual information. In addition, we also adopt shortcut connection in each stage to preserve the original information of the multilevel features, so as to avoid losing important features as much as possible.

Given the input image \( F_{in} \), and multilevel feature maps \( F_1 \), \( F_2 \), and \( F_3 \), the final output \( F_{out} \) of the MLSA module is obtained as

\[
F_{out} = F_{3in} + HDC(F_{3in}) + GAP(F_{3in})
\]

where \( F_{3in} \) is calculated as follows:

\[
F_{3in} = Up(F_{31} + HDC(F_{31}) + GAP(F_{31})) \Theta F_{in}
\]

\[
F_{31} = Up(F_{32} + HDC(F_{32}) + GAP(F_{32})) + Conv(F_1)
\]

\[
F_{32} = Up(Conv(F_3) + HDC(Conv(F_3)) + GAP(Conv(F_3)) + Conv(F_2))
\]

where \( Conv \) represents the \( Conv1 \times 1 \) operation with 32 channels, it is used to reduce the dimension of feature channels, so as to reduce the amount of parameters; \( Up \) denotes the upsampling.
block which is achieved by deconvolution; and $\Theta$ represents the feature concatenation.

It should be noted that, although the MLSA module is designed to enhance feature representation of the secret image with complex spatial features, its strong ability is also suitable for the ordinary cover image. Therefore, we introduce it into two branches of the hiding network to enhance feature representation of the secret image and the cover image.

### C. Crucial Role of the Lifeline

The lifeline designed in Fig. 2 is essentially a shortcut connection, but it plays a crucial role in our image hiding task. In this part, we provide a detailed analysis on its crucial role.

If there is no lifeline, the mapping target from the cover image to the hidden image can be described as

$$H_{i,j} = f_{w,b}(C_{i,j} \ast S_{i,j}) + C_{i,j}$$

(6)

where $C_{i,j}$ and $S_{i,j}$ represent the original cover image and secret image, respectively; $H_{i,j}$ indicates the synthetic hidden image; $\ast$ represents the complex interaction between the cover image and the secret image; and $f_{w,b}$ is the mapping function obtained by the network learning.

After introducing the lifeline shown in Fig. 2, the mapping target from the cover image to the hidden image can be described as

$$H_{i,j} = f_{w,b}(C_{i,j} \ast S_{i,j}) + C_{i,j}.$$  

(7)

Equation (7) can be written as the following residual form:

$$f_{w,b}(C_{i,j} \ast S_{i,j}) = H_{i,j} - C_{i,j}.$$  

(8)

The ideal goal of image hiding is to make the synthetic hidden image completely identical to the cover image, which can be regarded as an identity mapping problem. In order to satisfy $H_{i,j} = C_{i,j}$, just let $f_{w,b}(C_{i,j} \ast S_{i,j}) = 0$ in (8). Compared with the direct identity mapping shown in (6), the residual identity mapping shown in (8) is easier to optimize and fit. Because the direct identity mapping in (6) is completed by a stack of convolution–deconvolution layers, which inevitably leads to the loss of input information, and thus, it is difficult to achieve satisfactory results. While the designed lifeline can directly transfer the original cover image to the output, which makes it easier to achieve the identity mapping.

### D. Evaluation Index and Loss Function

To comprehensively evaluate the image hiding performance, not only the subjective visual appearance but also the objective quantitative index should be taken into account. Assume that the size of the two images is $W \times H$, several commonly used evaluation indexes are given as follows:

$$MSE = \frac{1}{WH}\sum_{i=1}^{W}\sum_{j=1}^{H}(O_{i,j} - G_{i,j})^2$$  

(9)

$$PSNR = 20\log_{10}(M) - 10\log_{10}(MSE)$$  

(10)

where $O_{i,j}$ and $G_{i,j}$ represent the original image and the generated image, respectively; $M$ denotes the maximum pixel value.

Usually, HVS is more sensitive to the structural information, thus, the index called SSIM [24] is introduced. Given two images $x$ and $y$, SSIM value is calculated with pixel means $u_x$ and $u_y$, variances $\delta_x^2$ and $\delta_y^2$, and the covariance $\delta_{xy}$

$$SSIM(x,y) = \frac{(2u_xu_y + C_1)\cdot (2\delta_{xy} + C_2)}{(u_x^2 + u_y^2 + C_1)\cdot (\delta_x^2 + \delta_y^2 + C_2)}$$  

(11)

where $C_1 = (k_1M)^2$, $C_2 = (k_2M)^2$; $M$ is the maximum pixel value, set $k_1 = 0.01$, $k_2 = 0.03$ by default [24].

Currently, most of the studies on the image reconstruction always use mean square error (MSE) as loss function for reason that it is a convex function with derivative for backpropagation. However, MSE will make the generated image too smooth and lack of details, therefore, a variety of loss functions (e.g., L1 loss [25], content loss [26], and adversarial loss [27]) are adopted, while SSIM is usually just applied as a metric for evaluating image quality. In this article, we attempt to introduce SSIM and design a mixed steganography loss function to force the network to focus on the structural features of the hidden image and the revealed image, and the mixed loss function is designed as

$$\text{Sumloss} = \text{Loss}(H_{i,j}, C_{i,j}) + \gamma \text{Loss}(R_{i,j}, S_{i,j})$$

$$= \{\alpha \text{MSE}(H_{i,j}, C_{i,j}) + \beta [1 - \text{SSIM}(H_{i,j}, C_{i,j})]\} + \gamma \{\alpha \text{MSE}(R_{i,j}, S_{i,j}) + \beta [1 - \text{SSIM}(R_{i,j}, S_{i,j})]\}$$  

(12)

where $H_{i,j}$ and $C_{i,j}$ represent the hidden image and the cover image, respectively; $R_{i,j}$ and $S_{i,j}$ are the revealed image and the secret image, respectively; $\gamma$ is a parameter to weigh the proportions of the hiding loss and the revealing loss in the mixed loss; and $\alpha$ and $\beta$ are a pair of hyper parameters.
III. EXPERIMENTAL ANALYSIS

In this section, we firstly introduce two data sets used in this article and the experimental setup. Then, we conduct extensive ablation experiments to thoroughly evaluate the effectiveness of our design choices. Finally, we give the state-of-the-art comparison to verify the superiority of our proposed method.

A. Data Sets and Experimental Setup

In this part, we give a brief introduction on the experimental data sets and setup as follows.

1) ImageNet [28] is widely used in the image recognition and classification challenge, each image is associated with a label from 1000 predefined classes.
2) NWPU-RESISC45 [29] contains 31,500 aerial images of 45 scene categories and has rich spatial diversity and variations. The spatial resolution varies from about 30 to 0.2 m/pixel.

In our experiments, we use a workstation equipped with an NVIDIA GeForce RTX 2080 Ti graphics processing unit (GPU), Python 3.6 programming languages and a Pytorch 1.3.0 framework under the Ubuntu 18.04 operating system. We select 30,000 images and 1500 images from the ImageNet data set as the training set and testing set of cover images, and then select 30,000 images and 1500 images from the NWPU-RESISC45 data set as the training set and testing set of secret images. ReduceLROnPlateau function is used to adaptively reduce the learning rate (LR). Parameters of the ReduceLROnPlateau function are set as factor = 0.1, patience = 3; α = 1, β = 0.2, γ = 0.6 are set in this article.

B. Backbone Network Design

In this part, we empirically analyze the influence of network depth and different feature fusion strategies on the network performance to demonstrate that the backbone network shown in Fig. 2 is elaborately designed, rather than roughly designed. For the convenience of expression, we use BL to represent the baseline backbone which is just composed of a stack of convolution–deconvolution layers, without adopting any feature fusion strategies; BL-C to represent the baseline backbone adopting symmetrical concatenation to connect the convolution and deconvolution layers; and BL-SC to represent the baseline backbone adopting symmetrical shortcut connection to connect the convolution and deconvolution layers. The experimental results are shown in Fig. 5.

Fig. 5 shows the backbone network with 6-layer, 8-layer and 10-layer symmetrical convolution–deconvolution, respectively. It can be seen from the solid lines that, if we do not take any strategy and blindly deepen the backbone network, not only the sumloss increases but also the amount of parameters dramatically increases as shown in Fig. 5(b). This indicates that although a deeper backbone network can help to extract higher level features and attributes, it will also cause more image details’ loss, resulting in the deterioration of steganography performance. Fig. 5(a) shows that both concatenation and shortcut connection strategies can effectively prevent the deterioration of steganography performance with the deepening of the network, however, even if these two feature fusion strategies are adopted, the steganography performance will encounter a bottleneck and the backbone network with six-layer convolution–deconvolution has achieved saturation performance. If the network is further deepened, it will not continue to reduce the steganography loss, but dramatically increase the amount of parameters, which explains why only six-layer convolution–deconvolution structure is designed for the backbone network. Fig. 5(a) and (b) shows that compared with the widely used concatenation strategy, shortcut connection in this article can still achieve ideal performance without adding any parameters, which explains why the backbone network adopts shortcut connection to connect the convolution and deconvolution instead of adopting the widely used concatenation strategy.

C. Ablation Study

To verify the effectiveness and reliability of our design choices, we analyze the influence of them on the steganography performance by adding or replacing them. Here we give the corresponding framework abbreviations as follows.

BL-SC: Baseline framework with symmetrical shortcut connection (As already stated, the baseline framework is composed of a stack of convolution–deconvolution layers, which is shown in Fig. 2).
BL-SC-SC: Baseline framework with symmetrical shortcut connection and skip concatenation.
BL-SC-SC-MLSA: Baseline framework with symmetrical shortcut connection, skip concatenation, and MLSA module.
BL-SC-SC-MLSA-L: Baseline framework with symmetrical shortcut connection, skip concatenation, MLSA module, and the lifeline.
BL-SC-SC-MLSA-L (mixed loss).

We use the designed mixed loss function to replace the traditional MSE loss function used in 1–4. In other words, the network structure of 5 and 4 are exactly the same, except the loss function. The specific location of different elements are shown in Fig. 2 and the average results are shown in Fig. 6. The followings can be observed from Fig. 6.

1) The black lines and green lines in Fig. 6(a) and (b) show that further introducing skip concatenation into the BL-SC framework can effectively reduce the hiding loss to improve the hiding performance, but slightly sacrifices the revealing performance. Since the image hiding task is different from the general single-object task, it not only pays attention to the synthesis quality of the hidden image but also focuses on the restoration quality of the secret image, that is, the performance of the hiding network will affect the performance of the subsequent revealing network.
Therefore, the strategy suitable for single-target task is not necessarily conducive to both the synthesis of hidden images and the recovery of secret images at the same time, which makes the image hiding task more complex than general computer vision tasks. However, it is worthwhile here to sacrifice part of revealing performance for the obvious improvement of hiding performance, because the hiding performance largely determines hiding security.

2) The blue line in Fig. 6(b) indicates that the MLSA module designed for the secret image with rich spatial features significantly boosts the performance of the revealing network and the final stable revealing loss drops by about 48%, and the performance of the hiding network is also improved to a certain extent as shown in Fig. 6(a), which indicates that the MLSA module designed for secret images also enhances the feature representation of ordinary cover images.

3) The red lines in Fig. 6(a) and (b) demonstrate the crucial role that the proposed lifeline plays in the image hiding task, and the lifeline obviously accelerates the speed of network learning, and significantly reduces the hiding loss and revealing loss, especially the hiding loss that drops by about 50%.

4) Fig. 6(c)–(d) shows that, compared with the traditional MSE loss function, the proposed mixed steganography loss function can effectively improve the SSIM value of the hidden images and revealed images relative to the original images.

However, here comes an intuitive problem. What is the effect of increasing SSIM value on the image hiding performance? To further visually show the influence of our design choices and the SSIM value on the image hiding performance, and provide a fair comparison, we randomly select a same pair of cover images and secret images from the testing set to verify abovementioned frameworks, and the visual results are shown in Fig. 7. To overcome the shortcomings of subjective visual evaluation, Table I gives the specific quantitative indexes of the hidden images and the revealed images displayed in Fig. 7.

In Fig. 7, from left to right are the original cover image, original secret image, hidden image, revealed image, and the residual image obtained by the difference between the hidden image and the original cover image, and the last column is the residual image magnified by 20 times. As shown in Fig. 7, only from the synthetic hidden image and the revealed image, it is difficult to distinguish the performance difference caused by different elements, and even from the residual image, we still cannot find any clues. However, after the residual image is magnified by 20 times; we can observe that; through designing the skip concatenation, MLSA module, and lifeline, the residual content between the hidden image and the original cover image is gradually reduced, especially the designed lifeline which greatly weakens the secret content in the magnified residual image. The last line of Fig. 7 shows that combined with the mixed loss function, our proposed full model completely eliminates the secret content in the magnified residual image, which indicates that our proposed full model with mixed loss function has the highest hiding security, even if the attackers have access to the hidden image and the cover image.

Table I gives the reason for the visual difference displayed in Fig. 7 through specific quantitative indicators. The design of skip concatenation, MLSA module, and mixed loss function significantly improves the SSIM value of the hidden image so that the perceptual quality of the hidden image is improved.
Thus, the visual difference between the hidden image and the original cover image is greatly reduced and this difference is mainly reflected in the embedded secret image. Therefore, as shown in the last column of Fig. 7, the secret content exposed in the magnified residual image is gradually eliminated with the improvement of the hidden image’s quality. Specific observations on Table I are given as follows.

1) The second row shows that further introducing skip concatenation into the BL-SC framework effectively improves the hidden image’s quality, but slightly sacrifices the revealed image’s quality.

2) The third row demonstrates that our proposed MLSA module significantly improves the quality of both the hidden image and the revealed image, especially the revealed image’s quality, which indicates that the MLSA module can indeed enhance the feature representation.

3) The fourth row shows that our proposed lifeline is quite important to the image hiding task and it obviously improves the hidden image’s quality.

4) The last row shows that the designed mixed loss function can force the network to pay more attention to the structural characteristics of images, thereby greatly improving the SSIM value of both the hidden image and the revealed image.

The results of the above observations are completely consistent with the curve display in Fig. 6.

In addition, it can be concluded from Fig. 6(c) and (d) and the last two rows of both Fig. 7 and Table I that in the image hiding task, more focus should be given to the SSIM value rather than the PNSR value. The higher the SSIM value of the synthetic hidden image, the less secret content exposed in the magnified residual image and the higher the security of the hidden image, which is shown in the last row of Fig. 7 and Table I. However, higher PSNR value does not mean better hiding performance, as shown in the fourth row of both Fig. 7 and Table I, the PSNR value of the hidden image is the highest, but slight trace of the secret image is still exposed in the magnified residual image. This also proves the importance of improving SSIM value by designing the mixed loss function.

To further explain that the parameters set in this article are more conducive to the network optimization than the default parameters of ReduceLROnPlateau function. We show their influence on the network performance in Fig. 8.

As shown in Fig. 8, the default parameters make the ReduceLROnPlateau function insensitive to the network loss, as a result, the LR remains unchanged at the initial value, which makes the network loss get stuck at locally optimal value and converge slowly. However, our selected parameters make the function more sensitive to the network loss and the LR is automatically reduced in time, thereby producing a faster and finer optimization.

D. State-of-the-Art Comparison

To comprehensively verify the superiority of our proposed method, we compared it with the typical representative of traditional steganography methods, namely LSB method, and the state-of-the-art image-to-image steganography methods [15]–[18] based on deep learning. It should be noted that, all of the deep learning-based methods [13]–[18] have achieved the goal of hiding the image data. However, we introduce methods [15]–[18] instead of methods [13] and [14] for comparison because methods [15]–[18] have achieved perceptually pleasing performance in the image hiding task and the secret images restored by them are of high visual quality. While the secret images restored by schemes [13] and [14] are seriously damaged in visual appearance and have large distortions relative to the original secret images, which means the secret image loses its integrity. To provide a fair comparison, we randomly select the same pair of cover images and secret images from the testing set for comparison. The results are shown in Fig. 9 and Table II.

As shown in Fig. 9, the traditional LSB steganography method is not suitable for hiding large image data and it leaves obvious modification traces in the hidden image, which is highlighted by the red box in row 1, column 3. Compared with the traditional steganography method, although deep learning-based method [15] gets rid of the dependence on manual feature extraction and greatly improves the hidden image’s quality, it still leaves
slight modification traces in the hidden image, which is highlighted by the red box in row 2, column 3. Rows 3–5 show that methods [16]–[18] further improve the hiding performance due to their mature algorithm design, they can imperceptibly embed the secret image into the cover image and ensure the visual quality of the hidden image because the modification traces are completely eliminated from the hidden image. Only from the synthetic hidden image and the revealed image, it is difficult to distinguish the performance difference caused by different methods. However, after the residual image is magnified, methods [16] and [17] display the secret content in the residual image magnified by ten times, which indicates that the hidden images generated by methods [16] and [17] have large distortions relative to the original cover images. Method [18] further improves the hidden image’s quality and the secret content in the residual image is further weakened. However, it cannot satisfy the visual security because the secret content is still exposed in the residual image magnified by 30 times. Our proposed method completely eliminates the secret content in the magnified residual image and only the contour of the ordinary cover image is displayed, which shows that our method has the highest visual security.

Table II gives the specific quantitative indexes of hidden images and revealed images displayed in Fig. 9 and overcomes the deficiency of subjective visual evaluation. The first row of Table II shows that LSB steganography method generates the hidden image and revealed image in poor quality when it is used to hide the image data, especially the revealed image, of which the PSNR value is only 26.6, and the SSIM value is only 0.86. Rows 2-5 of Table II show that, compared with the traditional steganography method, deep learning–based methods greatly improve the hiding performance. And the latest method maintains the PSNR value of both the hidden image and revealed image around 40, and the SSIM value about 0.99. In addition, although the method [17] can blur the secret content in the magnified residual image through algorithms of image transformation, it is not to improve the image hiding performance from the source by improving the image quality, and the SSIM value of the hidden image is only about 0.9722. The last row shows that both the hidden image and the revealed image generated by our proposed method have highest SSIM value, which indicates that our method improves the hiding performance from the source by improving the image quality.

Combined with the last two rows of both Fig. 9 and Table II, we can draw the conclusion again that, it is SSIM value that determines the visual security, not PSNR value, which again verifies the necessity of designing mixed loss function to improve SSIM value of the hidden image and revealed image.

Due to high embedding capacity of our proposed method and methods [15]–[18], they are not expected to counter the detection of the threat model namely steganalysis that can detect whether an image is embedded with secret information. Nevertheless, the presentation of their anti-detection ability is still of significance for future research. For this purpose, we firstly use each of above image hiding methods to generate 3500 hidden images, separately, and match these hidden images with their corresponding cover images to constitute the datasets. Then, we use these datasets to pre-train the modern steganalysis model called Ye-Net [30], which is then applied to detect the hidden images generated by above-mentioned methods. Table III presents average detection accuracy for different models. The lower the accuracy, the higher the security against steganalysis. The followings can be observed from Table III.

1) The proposed method performs the best among six methods and the main reason is that: it produces higher quality and more realistic hidden images to deceive steganalysis.

2) For the proposed method, its security against steganalysis needs to be further improved in future research.

To further demonstrate that the design choices shown in Table I have a wide compatibility, we attempt to apply them to other models. Due to the limitation of article space, the lifeline and mixed loss function are used for comparison, because neither of them adds any parameters to the original model, thereby guaranteeing a fair comparison. The average results on the testing set are shown in Table IV.

Table IV shows three groups of comparisons between the original model, the original model with lifeline, and the original model with lifeline and mixed loss function. The followings can be observed from Table IV.

1) The proposed lifeline significantly promotes the hiding performance.

2) The designed mixed loss function further improves the SSIM value of both hidden images and revealed images.

The above results are completely consistent with the conclusion drawn from Table I, which indicates that the proposed elements in this article have a wide compatibility.

### IV. CONCLUSION

On the basis of convolutional neural network, a novel end-to-end hiding–revealing network was designed to protect the secret images with complex spatial features. Designing symmetrical shortcut connection effectively improved the hiding and revealing performances without adding any parameters. The designed skip concatenation retained more raw information and further improved the hiding performance. The designed MLSA module enhanced feature representation of the secret images with complex spatial features. And then, the lifeline was
proposed to transform the image hiding task into residual identity mapping, which significantly improved the hidden image’s quality. Finally, a mixed loss function was designed to further improve the perceptual quality of both the hidden image and the revealed image, thereby ensuring the visual security.

Nevertheless, the proposed method depends on GPU due to massive parameters, therefore, a more lightweight model needs to be designed in future research to reduce the dependence on the hardware and its security against steganalysis needs to be further improved.

References

[1] M. Aazam, S. Zeadally, and K. A. Harras, “Deploying fog computing in industrial Internet of Things and Industry 4.0,” IEEE Trans. Ind. Informat., vol. 14, no. 10, pp. 4674–4682, Oct. 2018.

[2] X. Li, J. Niu, M. Z. A. Bluian, F. Wu, M. Karuppinah, and S. Kumar, “A robust ECC based provable secure authentication protocol with privacy protection for industrial Internet of Things,” IEEE Trans. Ind. Informat., vol. 14, no. 8, pp. 3599–3609, Aug. 2018.

[3] K. Muhammad, R. Hamza, J. Ahmad, J. Llorot, H. Wang, and S. W. Baik, “Secure surveillance framework for IoT systems using probabilistic image encryption,” IEEE Trans. Ind. Informat., vol. 14, no. 8, pp. 3679–3689, Aug. 2018.

[4] H. Zhou, K. Chen, W. Zhang, Y. Yao, and N. Yu, “Distortion design for secure adaptive 3D mesh steganography,” IEEE Trans. Multimedia, vol. 21, no. 6, pp. 1384–1398, Jun. 2019.

[5] J. Wang, L. Y. Zhang, J. Chen, G. Hua, Y. Zhang, and Y. Xiang, “Compressed sensing based selective encryption with data hiding capability,” IEEE Trans. Ind. Informat., vol. 15, no. 12, pp. 6560–6571, Dec. 2019.

[6] J. Mielenkainen, “LSB matching revisited,” IEEE Signal Process. Lett., vol. 13, no. 5, pp. 285–287, May 2006.

[7] V. Holub, J. Fridrich, and T. Denemark, “Universal distortion function for steganography in an arbitrary domain,” EURASIP J. Inf. Secur., vol. 1, pp. 1–13, Dec. 2014.

[8] B. Li, M. Wang, J. Huang, and X. Li, “A new cost function for spatial image steganography,” in Proc. IEEE Int. Conf. Image Process., 2014, pp. 4206–4210.

[9] T. Pevný, T. Filler, and P. Bas, “Using high-dimensional image models to perform highly undetectable steganography,” in Proc. Int. Workshop Inf. Hiding, 2016, pp. 161–177.

[10] V. Sedighi, J. Fridrich, and R. Cogranne, “Content-adaptive pentary steganography using the multivariate generalized Gaussian cover model,” in Proc. SPIE, 2015, Art. no. 94090H.

[11] J. Zhu, R. Kaplan, J. Johnson, and L. Fei-Fei, “HiDDeN: Hiding data with deep networks,” Jul. 2018, arXiv:1807.09937.

[12] K. A. Zhang, A. Cuesta-Infante, L. Xu, and K. Veeramachaneni, “SteganoGAN: High capacity image steganography with GANs,” 2018, arXiv:1901.03892.

[13] V. Kumar, S. Laddha, A. Niket, and N. Dogra, “Steganography techniques using convolutional neural networks,” Rev. Comput. Eng. Stud., vol. 7, no. 3, pp. 66–73, Sep. 2020.

[14] K. Sharma, A. Aggarwal, T. Singhania, D. Gupta, and A. Khanna, “Hiding data in images using cryptography and deep neural network,” J. Artif. Intell. Syst., vol. 1, no. 1, pp. 143–162, Dec. 2019.

[15] A. Rehman, R. Rahim, M. Nadeem, and S. Hussain, “End-to-end trained CNN encode-decoder networks for image steganography,” in Proc. Comput. Vis. ECCV 2018 Workshops, 2018, pp. 723–729.

[16] X. Duan, K. Jia, B. Li, D. Guo, E. Zhang, and C. Qin, “Reversible image steganography scheme based on a U-Net structure,” IEEE Access, vol. 7, pp. 9314–9323, Jan. 2019.

[17] S. Baluja, “Hiding images within images,” IEEE Trans Pattern Anal. Mach. Intell., vol. 42, no. 7, pp. 1685–1697, Jul. 2020.

[18] F. Chen, Q. Xing, and F. Liu, “Technology of hiding and protecting the secret image based on two-channel deep hiding network,” IEEE Access, vol. 8, pp. 21966–21979, Jan. 2020.

[19] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 770–778.

[20] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 4, pp. 834–848, Apr. 2018.