REVIEW

Survey on deep learning applications in digital image security

Zhenjie Bao[a] and Ru Xue[a,b,*]
[a]Xizang Minzu University, School of Information Engineering, Xianyang, China
[b]Key Laboratory of Optical Information Processing and Visualization Technology of Tibet Autonomous Region, Xianyang, China

Abstract. In the digital era, sharing pictures on social media has become a common privacy issue. To prevent private images from being eavesdropped on and destroyed, developing secure and efficient image steganography, image cryptography, and image authentication has been difficult. Deep learning provides a solution for digital image security. First, we make an overall conclusion on deep learning applications in image steganography to generate five aspects: the cover image, stego-image, embedding change probabilities, coverless steganography, and steganalysis. Second, we also combine and compare deep learning methods used in six aspects: image cryptography from image compression, image resolution improvement, image object detection and classification, key generation, end-to-end image encryption, and image cryptoanalysis. Third, we collect deep learning methods in image authentication from five perspectives: image forgery detection, watermarked image generation, image watermark extraction and detection, image watermarking attack, and image watermark removal. Finally, we summarize future research directions of deep learning utilization in image steganography, image cryptography, and image authentication. [DOI: 10.1117/1.OE.60.12.120901]

Keywords: deep learning; digital image security; image steganography; image cryptography; image authentication.

Paper 20210944V received Aug. 27, 2021; accepted for publication Dec. 7, 2021; published online Dec. 29, 2021.

1 Introduction

In such a highly information-oriented era, digital image transmission and delivery on the Internet have been increasingly frequent. the deliverer hopes that this only occurs in a secure channel while on the Internet; however; there are many eavesdroppers and destroyers. To protect individual privacy on public network platforms, researchers need to find an approach that satisfies both private image security and robustness. Image encryption, image steganography, and image authentication are three efficient methods that balance the characteristics of images and the requirement of security. Deep learning is a powerful tool in image processing that has reached impressive successes in image object detection,[1–3] image classification,[6–9] image segmentation,[10–13] image style transfer,[14–17] image denoising,[18–20] and image compression.[21–23] Applying deep learning methods to the field of image security to solve the traditional problems has also received extensive attention and achieved breakthrough progress recently. But how to better use the advantages of deep learning in image steganography, image cryptography, and image authentication always attracts many scholars' attention. To help relevant researchers understand the field of deep learning applications in digital image security and its future development more quickly, in this paper, we order the origin and development process of deep learning methods in image steganography, cryptography, and authentication from multiple aspects, as can be seen in Fig. 1; we then compare these methods, analyze the advantages and disadvantages of each, and finally suggest future research directions of this field.

This survey covers around 90 papers about deep learning for image steganography, cryptography, and authentication. The main contributions of this paper can be summarized as follows.

*Address all correspondence to Ru Xue, rxue@xzmu.edu.cn

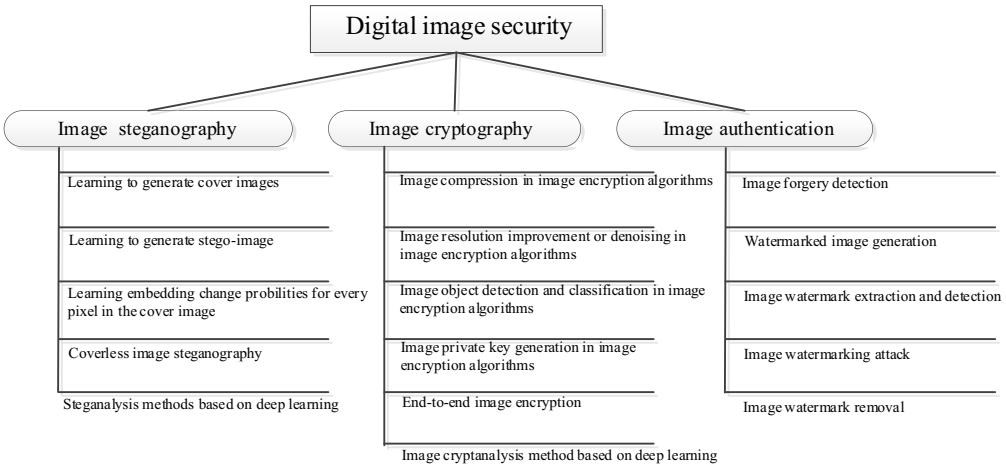

Fig. 1 The overall framework of the survey on digital image security.

1. This survey collects and analyzes the deep learning techniques in the field of digital image security from image steganography, image cryptography, and image authentication.
2. This survey estimates and compares the steganography, encryption, and watermarking performance of these approaches from quantitative indicators to reach the existing challenges.
3. This survey suggests research trends for deep learning in the use of image steganography, cryptography, and authentication to collision sparks in a wider research field.

The remainder of this paper is formed as follows. Section 2 collects the related works for traditional algorithms of image steganography, image encryption, and image authentication and compares them with the deep learning methods. Section 3 illustrates the deep learning applications for image steganography as well as a comparison and future research directions of these methods. Section 4 discusses the deep learning mechanisms in image cryptography, presents a performance comparison of the image encryption methods, and then points out the existing challenges. Section 5 represents the deep learning techniques in image authentication and explains the desire for future research. Section 6 suggests the future scope of deep learning in image security. Section 7 elaborates the survey's conclusions.

2 Related Works

2.1 *Image Steganography*

The purpose of image steganography is to send the stego-images like innocent normal images to the receiver and avoid the secret data being noticed by the attacker. As can be seen in Fig. 2, the process of image steganography is to embed the secret data into the cover image and then arrive

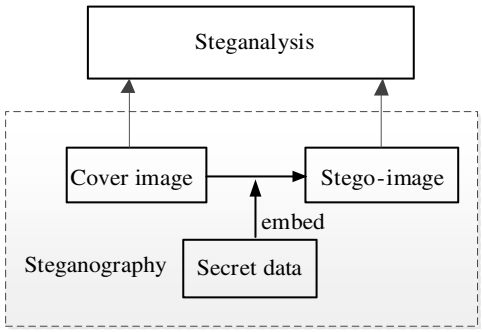

Fig. 2 The process of image steganography and steganalysis.

at the stego-image that steganalysis such as spatial rich model (SRM)[24] and maxSRMd2,[25] which are two feature-based classifiers, and Xu-Net,[26] which is a convolutional neural network classifier, tries to distinguish the stego-image from the cover image.

The least significant bit (LSB) replacement is a classical steganography algorithm that replaces the LSBs of the cover image with the secret data bits.[27] However, the LSBs of the pixel of the cover image take up a small part of the cover image, so the capacity is limited. Meanwhile, if we modify more bits of the pixel of the cover image to get a larger capacity, the possibilities to be detected by the attacker are higher. For higher capacity and higher undetectability, Pevný et al.[28] introduced HUGO, which provided a larger capacity while having equal safety compared with LSB matching.

Considering the relationship between the content of the image itself and the size of the secret image, adaptive steganography selects the edge[29] or texture[30] areas of the cover image as the embedding location as much as possible to strengthen the invisibility of stego-images. Li et al.[31] proposed a cost function for spatial image steganography that used a high-pass filter to locate the less predictable parts in an image, and then utilized two low-pass filters to make the low-cost values more clustered; it achieved better performance on resisting SRM steganalysis over HUGO[28] and S-UNIWARD.[30]

Deep learning image steganography methods can reach a higher capacity and have higher undetectability by not only the traditional feature-based steganalysis model but also deep learning steganalysis models over the traditional steganography algorithms. The capacity is measured by bits-per-pixel (bpp), which is the average number of bits concealed into each pixel of the cover image.[32] An ideal image steganography method could embed as much secret data as possible into the cover image without being detected by steganalysis methods. For example, Ref. 33 described that 0.2, 0.3, 0.4, and 0.5 bpp of secret data were embedded into the image from the BOSSbase[34] dataset using three algorithms. As illustrated in Table 1, the detection error rates of three steganalysis methods SRM,[24] maxSRMd2,[25] and Xu-Net[26] on the deep learning model in Ref. 33 are all higher than on traditional image steganography methods such as HILL[31] and S-UNIWARD,[30] which shows that deep learning-based image steganography has superiority in escape steganalysis.

2.2 *Image Encryption*

Image encryption keeps image content invisible until someone has the correct key. As we can see in Fig. 3, image cryptography involves encrypting the plaintext image to the ciphertext image by the encryption key and decrypting the ciphertext image to the decrypted image by the decryption key. Cryptanalysis involves cracking the cryptography to get the encryption/decryption keys or

Table 1 Different steganalysis models detection error rate of different image steganography methods.

Steganalysis models	Steganography methods	0.2 bpp (%)	0.3 bpp (%)	0.4 bpp (%)	0.5 bpp (%)
SRM	Ref. 33	**38.52**	**33.63**	**29.11**	**24.89**
	HILL	38.40	32.48	27.82	22.88
	S-UNIWARD	33.84	27.41	21.97	17.72
maxSRMd2	Ref. 33	**34.70**	**30.26**	**25.97**	**22.98**
	HILL	32.97	27.53	23.86	20.63
	S-UNIWARD	30.42	25.23	20.54	17.16
Xu-Net	Ref. 33	**42.64**	**38.53**	**33.56**	**29.71**
	HILL	36.80	31.06	25.76	21.91
	S-UNIWARD	37.55	30.43	24.34	19.25

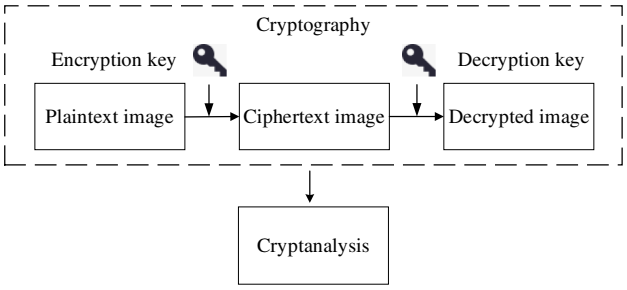

Fig. 3 The process of image cryptography and cryptanalysis.

the secret image; these includes ciphertext only attacks,[35] known-plaintext attacks,[36] chosen plaintext attacks,[37] and chosen ciphertext attacks.[38]

Chaos is sensitive to initial states and possesses a complex and unpredictable long-term behavior,[39] so it is widely used in image encryption. Bentoutou et al.[40] introduced an efficient image encryption method based on chaotic maps and the Advanced Encryption Standard. Naim et al.[41] described a satellite image encryption algorithm based on the linear-feedback shift register generator, SHA 512 hash function, hyperchaotic systems, and Josephus problem.

Based on the characteristic of optical systems,[42–44] cellular automata,[45–47] quantum,[48–50] and DNA computing,[51–53] there are many applications in image encryption.

In most deep learning image encryption methods, deep learning plays a supporting role, which gives full play to their special advantages. End-to-end deep learning image encryption is a new research direction and has achieved security that is the same as or even better than the traditional encryption methods. Table 2 compares three different image encryption methods in ciphertext image entropy; correlation coefficients of two horizontal, vertical, diagonal adjacent pixels; the number of pixels change rate (NPCR), and the unified averaged changed intensity (UACI) on satellite images, and the best values are shown in bold. As we can see in Table 2, the end-to-end deep learning image algorithm in Ref. 54 has better values in correlation coefficients and NPCR than the other two traditional image encryption methods. Furthermore, in other indexes, the deep learning method also has a similar value to the other two methods.

2.3 *Image Authentication*

Checking the image identity or image integrity is the target of image authentication. Image watermarking is a significant technique for authenticating images. As can be seen in Fig. 4, the process of image watermarking is like image steganography. The watermark is embedded into the container and arrives at the watermarked image. Watermark extraction or detection involves extracting or detecting and further authenticating the watermark. From the perspective of watermark visibility, an image watermark can be divided into the visible watermark[55] and the invisible watermark,[56] whereas from the perspective of watermark robustness, an image watermark can be classified by fragile watermark,[57] semifragile watermark,[58] and robust watermark.[59] Furthermore, from the mode of watermark extraction, an image watermark can be itemized into blind,[60] semiblind,[61] and nonblind extraction.[62]

Table 2 Ciphertext image security performance comparisons of different image encryption schemes.

Methods	Image entropy	Horizontal correlation coefficients	Vertical correlation coefficients	Diagonal correlation coefficients	NPCR	UACI
Ref. 40	**7.9993**	0.0008	0.0013	0.0017	0.9963	**0.3527**
Ref. 41	7.9977	0.0018	−0.0020	−0.0012	0.9962	0.3345
Ref. 54	7.9972	**0.0004**	**0.0005**	**−0.0011**	**0.9964**	0.3349

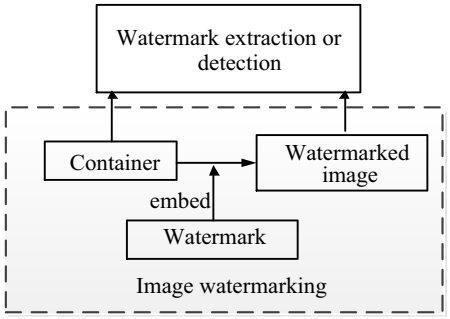

Fig. 4 The scheme of image watermarking and watermark extraction or detection.

Compared with the traditional image watermark methods, the deep learning methods have better imperceptibility and robustness. Table 3 describes the peak signal-to-noise ratio (PSNR) and normalized correlation (NC) between the extracted watermark from the watermarked image under multiattacks and the original watermark. Suppose that x and y are two images with the size of $M \times N$ and (i, j) is the pixel location in an image. The PSNR is calculated as Eq. (1), and NC is computed as Eq. (2), where if $a = b$, then $f(a, b) = 1$, and otherwise, $f(a, b) = 0$:[63]

$$\mathrm{PSNR} = 10\log_{10}\left\{\frac{255^2}{\frac{1}{MN}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}[x(i,j)-y(i,j)]^2}\right\}, \tag{1}$$

$$\mathrm{NC} = \frac{1}{MN}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1} f(a,b). \tag{2}$$

As can be seen in Table 3, when watermarked images face the attacks like median filtering of 3×3, salt and pepper noise with noise level δ of 0.005, and JPEG compression with quality factor 20, the deep learning method in Ref. 63 reaches almost all of the highest watermark quality assessment values, which are shown in bold, while it also obtains the most complete watermark with respect to no attack.

3 Deep Learning in Image Steganography

With regards to the traditional problem of image steganography for embedding capacity and security, the deep learning method in image steganography helps a lot (Table 4).

Table 3 Comparisons of different methods facing multiattacks.

Attack types	Metrics	Ref. 59	Ref. 62	Ref. 63
No attack	PSNR (dB)	48.47	51.45	**58.91**
	NC	**1.00**	**1.00**	**1.00**
Median filtering (3×3)	PSNR (dB)	32.12	**36.71**	36.42
	NC	0.44	0.78	**0.82**
Salt and pepper (δ: 0.005)	PSNR (dB)	32.74	31.66	**54.12**
	NC	0.81	0.92	**1.00**
JPEG compression (QF: 20)	PSNR	25.38	16.34	**30.21**
	NC	0.44	0.32	**0.56**

Table 4 Some deep learning models for image steganography tasks.

Series	References	Characteristics
Learning to generate cover images	Volkhonskiy et al. (2017)[65]	Use deep convolutional GANs to generate the cover image
	Shi et al. (2017)[67]	Use WGAN to generate cover image and GNCNN to analyze steganography
Learning to generate stego-image	Hayes et al. (2017)[70]	Define a game between three parties represented by the neural network
	Baluja (2017)[71]	Embed a color image into another image of the same size using deep neural networks
	Chattopadhyay et al. (2018)[72]	Use multistage feed-forward artificial neural network
	Zhu et al. (2018)[73]	Jointly train encoder and decoder networks
	Hu et al. (2018)[74]	Map the secret data into a noise vector used by the trained generator to produce the carrier image
	Zhang et al. (2019)[76]	Hide binary data in images using GANs
	Wang et al. (2019)[77]	Utilize a new secret information preprocessing method, Inception-ResNet block, GAN, and perceptual loss
	Zhang et al. (2019)[32]	Use an inception-module-based neural network to embed a secret gray image into an image with the same size, which was the Y channel of the cover image, and propose a mixed loss function
	Duan et al. (2020)[78]	Use residual block in the hidden network to generate stego-image
Learning embedding change probabilities for every pixel in the cover image	Tang et al. (2017)[84]	A generator to learn the probability map, then the secret message embedding simulated by the TES that was presented by a neural network
	Yang et al. (2019)[33]	Has three modules: a generator with a U-Net architecture, a no-pretraining-required double-tanh function, and an enhanced steganography analyzer
	Tang et al. (2021)[86]	Employ reinforcement learning to learn the embedding policy
Coverless image steganography based on deep learning	Duan et al. (2018)[87]	A coverless image steganography based on a generative model
	Luo et al. (2020)[88]	A coverless image steganography method based on multiobject recognition
	Liu et al. (2019)[89]	A coverless image steganography algorithm based on image retrieval of DenseNet features and DWT sequence mapping
	Zhang et al. (2019)[91]	A diversity image style transfer network using multilevel noise encoding
	Zhou et al. (2019)[92]	Use a faster region-based CNN and a dictionary that defined the objects and corresponding codes
	Duan et al. (2020)[94]	Coverless information hiding method that constructs improved Wasserstein GAN model

Table 4 (*Continued*).

Series	References	Characteristics
Steganalysis methods based on deep learning	Qian et al. (2015)[69]	A single neural network called GNCNN employing Gaussian function as the activation function
	Xu et al. (2016)[26]	Xu-Net, a CNN architecture in consideration of the knowledge of steganalysis
	Yang et al. (2017)[95]	Incorporate selection-channel awareness into modified Xu-Net
	Zeng et al. (2017)[96]	A hybrid CNN using the domain knowledge behind rich models for JPEG steganalysis
	Ye et al. (2017)[97]	An alternative approach to steganalysis of digital images based on a convolutional neural network named Ye-Net
	Yedroudj et al. (2018)[98]	Yedroudj-Net, which improves the architecture of the convolutional neural network
	Zhang et al. (2018)[99]	Zhu-Net, which adopts small-sized convolutional kernel for preprocessing, separable convolution to enhance the stego-signal, spatial pyramid pooling, and data augmentation
	Boroumand et al. (2019)[100]	SR-Net, which adopts an expanded front part in a deep residual neural network
	Reinel et al. (2021)[101]	GBRAS-Net using filter banks for preprocessing, and depth-wise, separable convolution, skip connections for feature extraction
	Liu et al. (2021)[102]	DFSE-Net involving diverse filter modules and squeeze-and-excitation modules
	Iskanderani et al. (2021)[103]	An efficient θ-nondominated sorting genetic algorithm-III based DCNN model
	Singhal et al. (2021)[104]	Blind steganalysis for multiple categories in spatial and JPEG images by the deep residual network

3.1 *Learning to Generate Cover Images*

As can be seen in Fig. 5, using deep convolutional generative adversarial networks (DCGAN),[64] Volkhonskiy et al.[65] proposed steganographic generative adversarial networks (SGAN), which was trained on the Celebrities dataset. Liu et al.[66] generated cover images and embedded secret messages using the LSB algorithm to deceive the steganography analyzer. Similar to SGAN, secure steganography based on generative adversarial networks (SSGAN)[67] used wasserstein generative adversarial networks (WGAN)[68] as the cover images generator and Gaussian-neuron convolutional neural networks (GNCNN)[69] as the steganography analyzer trained on the CelebA[66] database to improve the training performance and image quality. They opened the field of container generation, although they lacked considerations in steganography properties.

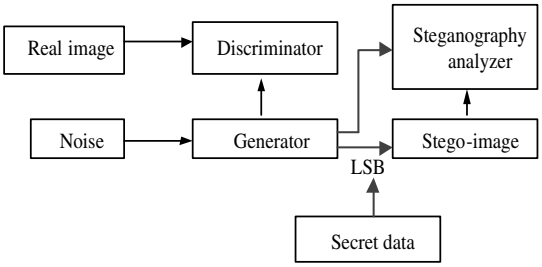

Fig. 5 The structure of SGAN and SSGAN.

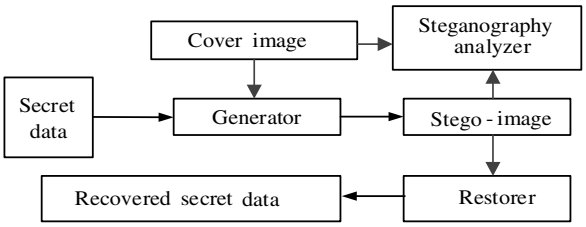

Fig. 6 The framework of the deep learning model in stego-image generation.

3.2 *Learning to Generate Stego-Image*

As can be seen in Fig. 6, the deep learning model in stego-image generation consists of three parts: the generator translates the cover image and secret data to the stego-image, the restorer recovers the secret data from the stego-image, and the steganography analyzer determines if the input image is either the stego-image or the cover image.

Hayes and Danezis[70] simulated a communication scenario in which three neural networks trained on the CelebA dataset engaged: Alice and Bob communicated by concealing a secret message in the carrier image, and Eve eavesdropped on the image and distinguished the embedded image from the innocent image. Baluja[71] designed three neural networks: a preparation network gained the features of the secret image, a hiding network concealed the extracted features into the cover image across all available bits, and a revealing network reconstructed the secret image from the stego-image. Chattopadhyay et al.[72] adopted a multistage feed-forward artificial neural network to complete image steganography. Zhu et al.[73] transferred the input message and cover image to a discriminator-indetectable image by a neural network and recovered the secret message from the encoded image by the other neural network simultaneously. Hu et al.[74] trained a generator to translate a carrier-like image deceiving the discriminator using the vector calculated by the secret information and an extractor to reconstruct the vector and map it to the secret message reversely. The neural networks were trained on the CelebA and Food-101[75] datasets. However, the disadvantages of DCGAN result in some drawbacks. For example, some generated stego-images were not sufficiently natural, the size of the stego-image was small, and the steganography capacity was not big adequate. Zhang et al.[76] hid arbitrary binary data in images using generative adversarial networks (GAN). Wang et al.[77] utilized a secret information preprocessing method, Inception-ResNet block, GAN, and perceptual loss to improve the undetectability, imperceptibility, and capacity, although the model had certain limitations on the length of secret information. Zhang et al.[32] used an inception-module-based neural network to embed a secret gray image into an image with the same size, which was the Y channel of the cover image; a neural network to recover the secret image from the Y channel of the stego-image, which tried to minimize a mean square error (MSE), structural similarity (SSIM), and multiscale SSIM mixed loss function; and another neural network to judge whether the input image was the stego-image or not. Duan et al.[78] employed the residual learning block in the hiding network to directly generate a stego-image that looked like the cover image and designed the reveal network to recover the secret image.

Table 5 estimates PSNR and SSIM indices between the cover image and stego-image obtained by SteGAN,[70] HiDDeN,[73] SteganoGAN,[76] and HidingGAN[77] with embedding capacities of 0.4, 0.4, 4.4, and 4 bpp, respectively, in the COCO[79] dataset; ISGAN[32] with embedding capacity of 8 bpp; Ref. 71 with embedding capacity of close to 24 bpp in the ImageNet[80] dataset; and Ref. 78 with embedding capacity of 23.8 bpp in the ImageNet, LFW,[81] or Pascal-VOC[82] datasets. The invisibility requires the stego-image to be similar to the cover image. And the higher the values of PSNR and SSIM are, the more similar the two images are. PSNR estimates the error between corresponding pixels of two images, but it does not considered the human visual characteristics; the SSIM[83] measures the image similarity from comparisons of luminance, contrast, and structure.

In practice, these two indices should both be considered. Table 6 compares the steganography capacity and cover image size of different deep learning methods. As can be seen in Fig. 7, the method proposed by Ref. 71 has the highest PSNR, SSIM, and capacity values, and its SSIM

Table 5 Comparison of different deep learning methods in PSNR and SSIM between stego-images and cover images on different image datasets.

Methods (capacity, datasets)	PSNR (dB)	SSIM
SteGAN[70] (0.4 bpp, COCO)	21.43	0.69
HiDDeN[73] (0.4 bpp, COCO)	33.40	0.96
SteganoGAN[76] (4.4 bpp, COCO)	36.33	0.88
HidingGAN[77] (4 bpp, COCO)	33.16	0.96
ISGAN[32] (8 bpp, ImageNet)	34.89	0.97
Ref. 78 (23.8 bpp, ImageNet, LFW, or Pascal-VOC)	40.62	0.98
Ref. 71 (close to 24 bpp, ImageNet)	**41.2**	**0.98**

Table 6 Comparison of different deep learning methods in capacity and cover image size.

Methods	Cover image size	Capacity (bits/pixel)
SteGAN	32×32	0.4
HiDDeN	512×512	0.203
Ref. 74	64×64	0.009
SteganoGAN	—	4.4
HidingGAN	256×256	4
ISGAN	256×256	8
Ref. 78	256×256	23.8
Ref. 71	200×200	**Close to 24**

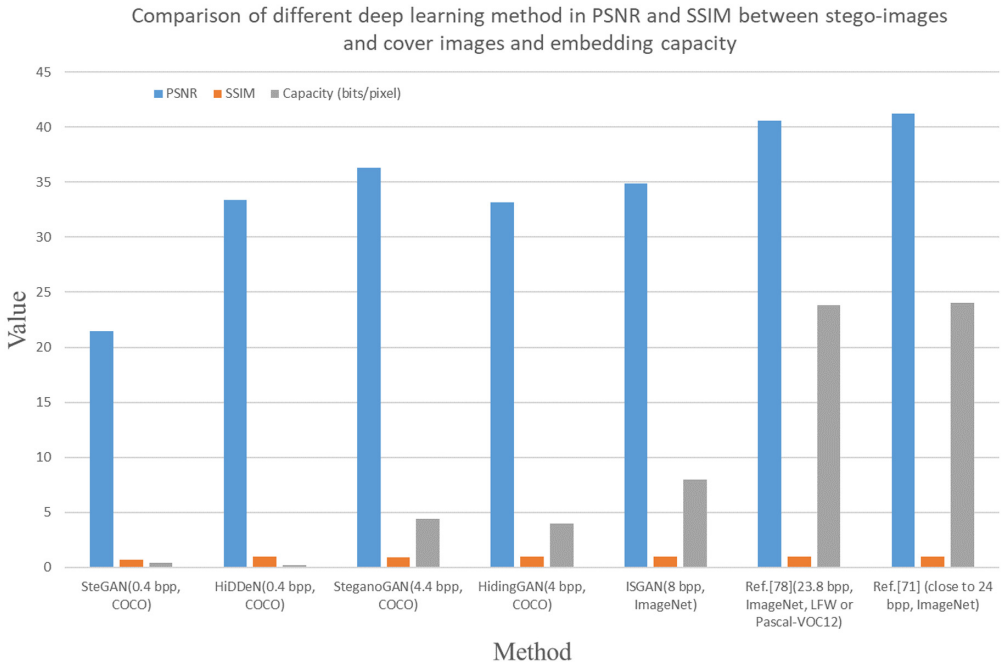

Fig. 7 Histogram comparison of different deep learning methods in PSNR and SSIM between the stego-image and the cover image and embedding capacity.

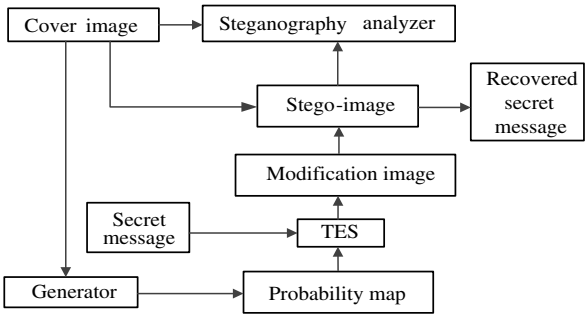

Fig. 8 The structure of deep learning model in probability map generation.

value is close to the theoretical maximum value 1, which is the best stego-image imperceptibility and the highest embedding capacity among all compared methods.

The deep learning methods generate the stego-image and recover the secret data automatically while having a large embedding capacity and a good performance in undetectability. But they do not recover the secret data 100% when the stego-image does not suffer any attacks. Deep learning methods are not completely accurate secret data extraction algorithms in steganography although they perform well in real complex communication situations. The secret data extraction accuracy is limited by the characteristics of the neural network. Furthermore, the fixed input and output size of deep learning model limits the size of the image that it could process.

3.3 *Learning Embedding Change Probabilities for Every Pixel in the Cover Image*

As illustrated in Fig. 8, Tang et al.[84] trained a generator to learn the probability map and then embedded the secret message simulated by the ternary embedding simulator (TES), which was presented by a neural network; finally the obtained modification map was added to the cover image to achieve the stego-image that attempts to deceive the steganography analyzer. On the basis of Ref. 84, Yang et al.[33] presented a U-Net-based[85] generator and replaced the TES with the double-tanh function, which does not need to be trained. In addition, six high-pass filters were integrated into the steganography analyzer. Similar to Refs. 33 and 84, Tang et al.[86] employed reinforcement learning to learn the embedding policy, which performed pixel-level actions and rewards.

The above three methods were tested in embedding 0.1, 0.2, 0.3, 0.4, and 0.5 bpp of secret data into the image from the BOSSbase dataset, and SRM was used to detect them. As illustrated in Table 7, the detection error rates of Ref. 86 are all higher than the other compared algorithms and have the best undetectability.

3.4 *Coverless Image Steganography*

The coverless image steganography does not change the cover image but transmits it directly, so it does not easily raise the suspicion of the attacker, which can effectively resist steganalysis from a new angle. As shown in Fig. 9, Duan et al.[87] generated the stego-image directly using two generators and two discriminators trained on the CelebA dataset. Generator 1 translates the secret image to the cover image, and discriminator 1 distinguishes the stego-image from the cover

Table 7 Detection error rate of different image steganography methods using SRM.

Methods	0.1 bpp (%)	0.2 bpp (%)	0.3 bpp (%)	0.4 bpp (%)	0.5 bpp (%)
ASDL-GAN[84]	36.19	31.24	25.41	21.95	18.04
UT-GAN[33]	43.53	36.87	32.26	27.30	22.52
SPAR-RL[86]	**45.15**	**38.43**	**32.68**	**28.30**	**23.80**

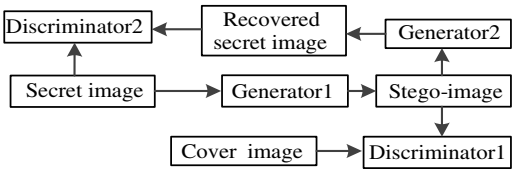

Fig. 9 The structure of deep learning model in stego-image direct generation in Ref. 87.

image. Furthermore, generator 2 translates the stego-image to the secret image, and discriminator 2 determines whether the input image is generated or not. Luo et al.[88] showed a coverless image steganography scheme based on multiobject recognition. Liu et al.[89] retrieved images according to the features DenseNet[90] trained on the ImageNet[80] dataset and generated feature sequences using discrete wavelet transform (DWT) coefficients, which presented the secret data and had good robust and security performance in resisting most image attacks. Zhang et al.[91] discussed a diverse image style transfer network trained on the Microsoft COCO dataset[79] using multilevel noise encoding with the image style transfer results presenting different codewords. However, the images with particularly dense textures were not suitable for datasets of steganography. Zhou et al.[92] created a dictionary that defined the objects and corresponding codes and used a faster region-based convolutional neural network trained on the PASCAL VOC 2007[82] and VOC 2012[93] datasets to detect objects in stego-images that would extract the secret message. Duan et al.[94] trained an improved Wasserstein GAN model on the LFW dataset[81] and transmitted and input the disguised images with the generator outputting the images similar to the secret images in visual mode.

3.5 *Steganalysis Methods Based on Deep Learning*

A deep learning model can automatically learn image features, which contributes to better classification precision. Utilizing the advantages of the deep learning model, Qian et al.[69] applied a single neural network called GNCNN, which employed six convolutional layers for extracting the image features, three fully connected layers for classifying, and a Gaussian function as the activation function for separating the cover and stego-signals. Xu et al.[26] proposed Xu-Net, which took absolute values of elements in the feature maps generated from the first convolutional layer to force the model to take into account the sign symmetry[24] that existed in noise residuals and constrained the range of data values with the saturation regions of tanh in the first two convolutional layers, while constraining the strength of the model using 1×1 convolutions in the last three convolutional layers to avoid overfitting. Yang et al.[95] incorporated selection-channel awareness into a modified Xu-Net architecture trained on the BOSSbase v1.01 dataset,[34] which applied large weights to features learned from complex texture regions and small weights to features learned from smooth regions. Zeng et al.[96] convolved the images with a set of kernels, then calculated the different quantized and truncated features, and finally used the CNN model trained on the ImageNet dataset to process the extracted features for JPEG steganalysis. Jian et al.[97] presented an image steganalysis CNN-based model called Ye-Net, which was initialized with the filters used in calculating residual maps in SRM and integrated them with the truncated linear unit to suit the distribution of embedding signals (with low signal-to-noise ratio) and selection channel awareness. Yedroudj-Net[98] involved a predefined convolutional layer for extracting the noise component residuals and adopted scale operations in the last three convolutional layers. Zhu-Net[99] applied 3×3 convolutional kernels to reduce the number of parameters and model the features in a small local region, used separable convolution to enhance the stego-signal-to-noise ratio, and employed spatial pyramid pooling to aggregate the local features for strengthening the representation ability of features. SR-Net[100] adopted an expanded front part in a deep residual neural network without max-pooling operations to minimize the use of heuristics and externally enforced elements. GBRAS-Net[101] used filter banks in the preprocessing phase and depth-wise, separable convolution, and skip connections in the feature extraction phase. Liu et al.[102] constructed DFSE-Net with diverse filter parts that combined three different scale convolution filters that could process information diversely and squeeze-and-excitation parts that could enhance the effective channels out from diverse filter parts. Based on an efficient θ-nondominated sorting

Table 8 Comparison of detection accuracy of deep learning steganalysis methods for WOW and S-UIWARD at embedding capacity of 0.2 and 0.4 bpp.

Algorithms	WOW 0.2 bpp (%)	WOW 0.4 bpp (%)	S-UIWARD 0.2 bpp (%)	S-UIWARD 0.4 bpp (%)
GNCNN[69]	61.4	70.7	53.7	69.1
Xu-Net[26]	67.5	79.3	60.9	72.7
Ye-Net[97]	66.9	76.7	60.1	68.7
Yedroudj-Net[98]	72.3	85.1	63.5	77.4
Zhu-Net[99]	76.9	88.1	71.4	84.5
SR-Net[100]	75.5	86.4	67.7	81.3
GBRAS-Net[101]	**80.3**	**89.8**	**73.6**	**87.1**
DFSE-Net[102]	75.3	85.1	65.9	78.5

genetic algorithm-III, Iskanderani et al.[103] described a densely connected CNN (DCNN) for image steganalysis. θ NSGA-III was utilized to tune the initial parameters of the DCNN model. It could control the accuracy and f-measure of the DCNN model by utilizing them as the multiobjective fitness function. Singhal and Bedi[104] proposed multiclass blind steganalysis that included 60 layers to record demographic features and residual mapping to retain weak stego-signals generated by embedding payload and thus making classification easier and utilized a high-pass filter to preprocess images.

Table 8 and Fig. 10 illustrate the detection accuracy of different deep learning steganalysis algorithms in multiple steganography conditions, which consist of using the WOW[105] steganography algorithm with 0.2 and 0.4 bpp embedding capacity and using the S-UNIWARD[30] steganography algorithm with 0.2 and 0.4 bpp embedding capacity on the BOSSbase dataset. Compared with other algorithms, GBRAS-Net[101] achieves the highest steganalysis accuracy.

Despite the image steganography method based on deep learning achieving an impressive result in embedding capacity and stego-image quality, these methods have a lack of robustness. The robustness of image steganography algorithms based on deep learning needs to be further discussed and improved in future studies. Furthermore, the input and output of the deep learning model are fixed, so the deep learning models only process the images of a fixed size. Developing a model that could process images of various sizes in the field of image steganography is a meaningful task.

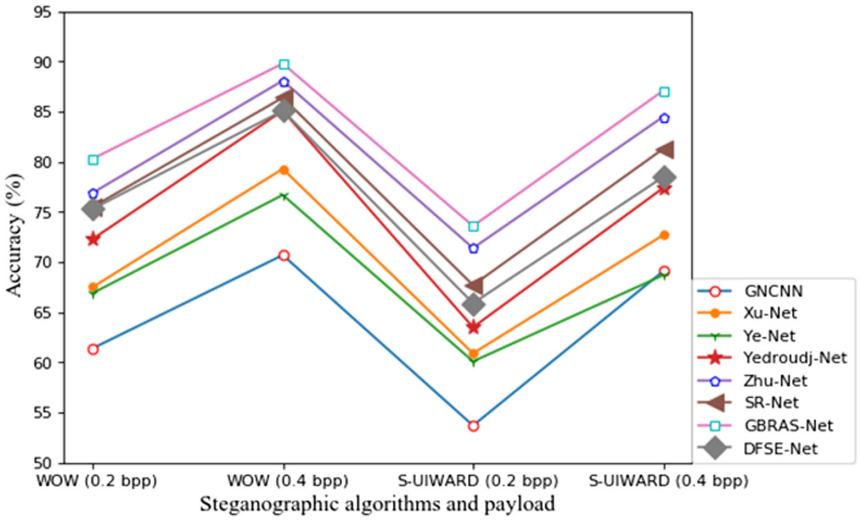

Fig. 10 Steganalysis accuracy comparisons of the deep learning steganalysis techniques against WOW and S-UNIWARD algorithms with the embedding capacity of 0.2 and 0.4 bpp.

4 Deep Learning in Image Cryptography

4.1 *Image Compression in Image Encryption Algorithms*

Image compression can increase the efficiency of image encryption by reducing the size of data. Chen et al.[106] utilized a deep learning model trained on a color image dataset[107] to compress and reconstruct the plaintext image and compound the chaotic system to encryption. Hu et al.[108] employed stacked autoencoder for compression and chaotic logistic map to encrypt the compressed vector. Suhail and Sankar[109] presented an application of image compression and encryption using an autoencoder and chaotic logistic map. Selvi et al.[110] suggested a competent adaptive sigma filterized synorr certificateless signcryptive Levenshtein entropy coding-based deep neural learning technique trained on a dataset of chest x-ray images to develop the image encryption and compression (Table 9).

Table 9 Some deep learning models for image cryptography tasks.

Series	References	Characteristics
Deep learning for image compression in image encryption algorithms	Chen et al. (2020)[106]	Utilize a deep learning model to compress and reconstruct the plaintext image and compound chaotic system to encryption
	Hu et al. (2016)[108]	Stacked autoencoder for compression and chaotic logistic map for encryption
	Suhail et al. (2020)[109]	Autoencoder for compression and chaotic logistic map for encryption
	Selvi et al. (2021)[110]	A competent adaptive sigma filterized synorr certificateless signcryptive Levenshtein entropy coding-based deep learning technique
Deep learning for image resolution improvement or denoising in image encryption algorithms	Zhang et al. (2019)[111]	Employ ghost imaging as a transmission and encryption mode and a CNN to improve the reconstructed image resolution
	Chen et al. (2019)[113]	Utilize a dilated deep CNN denoiser that improves the resolution of the fractional Fourier transform-based decrypted images
Deep learning for image object detection and classification in image encryption algorithms	Zhao et al. (2020)[115]	Utilize the MTCNN to seek key feature points of human faces, then adopt a combination of chaotic logic diagrams and RC4 stream ciphers to encrypt features
	Alqaralleh et al. (2021)[116]	Apply elliptic curve cryptography, employ the neighborhood indexing sequence with burrow wheeler transform to encrypt the hash values, and utilize a deep belief network for the classification process to diagnose the existence of the disease
	Asgari-Chenaghlu et al. (2021)[117]	A method based on YoloV3 object detection and chaotic image encryption
Deep learning for image private key generation in image encryption algorithms	Li et al. (2018)[118]	Train deep learning model to gain the features of iris image, then use the RS error correcting code to calculate the encryption key, finally encrypt the image using XOR operation
	Ding et al. (2021)[120]	Use the GAN to generate the private key
	Jin et al. (2020)[122]	The method based on deep neural network learning to induce the symmetric key creation
	Maniyath et al. (2020)[123]	Adopt a robust deep neural network that generates secret key resistive of different forms of attack and chaotic map to encrypt
	Erkan et al. (2020)[124]	Use sensitive key generation by deep convolution neural network to produce a diverse chaotic sequence for encrypting operations
	Fratalocchi et al. (2021)[125]	Train a neural architecture to learn the mapping algorithm between the key and the physical unclonable function

Table 9 (*Continued*).

Series	References	Characteristics
Deep learning for end-to-end image encryption	Li et al. (2020)[127]	An optical image encryption learning scheme based on Cycle-GANs
	Ding et al. (2021)[128]	Employ Cycle-GAN to encrypt and decrypt the medical images like a style transfer task
	Bao et al. (2021)[129]	Adversarial autoencoder for image scrambling based on asymmetric encryption
	Bao et al. (2021)[54]	Employ the traditional diffusion technique to enhance the avalanche effect of Cycle-GAN-based image encryption methods
Image cryptanalysis method based on deep learning	Xu et al. (2021)[132]	A deep learning method to attack the phase truncated Fourier transform encryption system
	Hai et al. (2019)[134]	Train a deep neural network model to learn and crack the Random Phase Encoding based optical cryptosystems
	Wu et al. (2020)[136]	A model trained with large numbers of ciphertext-plaintext pairs to crack the modified diffractive-imaging-based image cryptosystem
	Chen et al. (2020)[138]	A CNN directly converting the ciphertext image encrypted by the joint transform correlation structure to the original plaintext image
	He et al. (2019)[139]	A deep learning-based decrypted image generation approach to unravel the image encryption method in Ref. 140

4.2 *Image Resolution Improvement or Denoising in Image Encryption Algorithms*

Zhang et al.[111] employed ghost imaging as a transmission and encryption mode and a CNN that was trained on the IAPR TC-12 benchmark[112] to improve the recovered image resolution. Chen et al.[113] utilized a dilated deep CNN denoiser trained on the Waterloo exploration database,[114] which improved the resolution of the fractional Fourier transform-based decrypted images and resistance against multiclass attacks.

4.3 *Image Object Detection and Classification in Image Encryption Algorithms*

Zhao et al.[115] applied the multitask cascaded convolution network (MTCNN) to seek five key feature points of human faces and then adopted a combination of chaotic logic diagrams and Rivest Cipher 4 stream ciphers to encrypt eigenvalues. At the same time, the face coordinates generated by MTCNN and user passwords were hash-converted and double-encrypted by a hash table, which reduced the size of the images to be encrypted. Alqaralleh et al.[116] employed elliptic curve cryptography with an optimal key generated by hybridization of grasshopper with the fruit fly optimization algorithm, then used the neighborhood indexing sequence with burrow wheeler transform to encrypt the hash values, and finally adopted a deep belief network to classify the existence of the disease. Asgari-Chenaghlu et al.[117] described a technique based on YoloV3 object detection and chaotic image encryption that had the ability of automatic image encryption on both full or user-selected regions.

4.4 *Image Private Key Generation in Image Encryption Algorithms*

Li et al.[118] trained a CNN on the CASIA iris database version 4.0[119] to extract the feature of the iris image, then employed the RS error correcting code to encode the feature vector, and calculated the encryption key that was adopted to encrypt the plaintext image by the XOR operation.

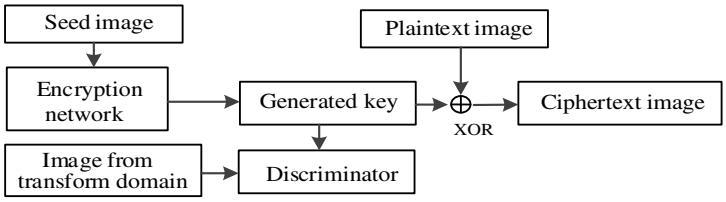

Fig. 11 The structure of the deep learning model in image private key generation and image encryption in Ref. 120.

As can be seen in Fig. 11, Ding et al.[120] trained a GAN trained on Montgomery County's chest x-ray[121] dataset to generate the private key and input a seed image that had a large key space, pseudorandomness, a one-time pad, high sensitivity to change, and resistance to different kinds of attacks; then the bit-wise XOR algorithm was adopted as an encryption and decryption algorithm. Jin and Kim[122] illustrated a method based on deep neural network learning without sharing the preshared key between systems and created and used the key that was variably used through the symmetric key encryption system of the 3D cube algorithm, which provided good security. Maniyath and Thanikaiselvan[123] proposed a robust deep neural network trained on an SIPI image database that generated a secret key resistive of multiattacks and applied a chaotic map to encrypt the image without any negative effect on image quality. Erkan et al.[124] mechanized a CNN trained on the ImageNet database to generate sensitive keys and then produced initial values and controlled parameters for the hyperchaotic log-map; thus they obtained a diverse chaotic sequence for image encryption. Fratalocchi et al.[125] decoupled the design of the physical unclonable functions from the key generation and trained a neural structure to learn the mapping between the key and the physical unclonable function, which could address the shortcomings of unreliability and weak unpredictability of cryptographic keys.

4.5 *End-to-End Image Encryption*

As can be seen in Fig. 12, the main end-to-end image encryption deep learning scheme consists of the encryption network that generates a random-like ciphertext image, the decryption network that reconstructs the plaintext image, and the discriminator that distinguishes the ciphertext images from the pixel-random images.

Furthermore, cycle-consistent generative adversarial network (Cycle-GAN)[126] has a good performance in image style transfer, in which the process of image encryption is regarded as translating the usual images to images with randomly distributed pixels. Thus the neural network structure of Cycle-GAN was widely used as the encryption or decryption network in end-to-end image encryption methods based on deep learning. The neural network structure, described in Fig. 13, down-samples the input, then extracts the feature map through nine residual blocks,[9] and finally up-samples the feature map to output the image with the objective style. One style transfer process of Cycle-GAN can be seen in Fig. 14, where the generator translates the original image to the generated image with objective style and the discriminator distinguishes whether the image is a real image.

Li et al.[127] demonstrated an optical image encryption learning scheme based on Cycle-GAN that was trained by the plaintext-ciphertext training set of satellite images in which the ciphertext images were encrypted by double random phase encoding. Ding et al.[128] employed Cycle-GAN

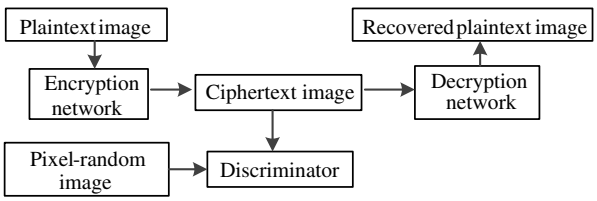

Fig. 12 The main scheme of the deep learning model in end-to-end image encryption.

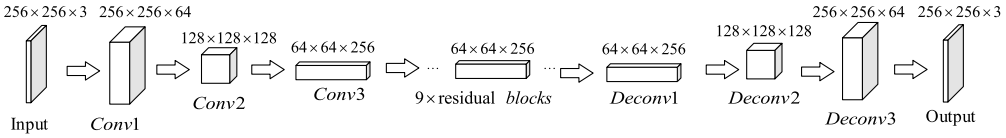

Fig. 13 The generator neural network architecture of Cycle-GAN.

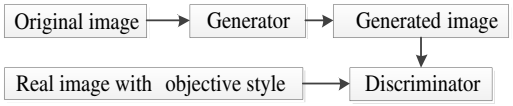

Fig. 14 One style transfer process of Cycle-GAN.

trained on a dataset of chest x-rays[121] to encrypt and decrypt medical images as a style transfer task. In addition, a neural network is projected to gain the interested object from the ciphertext image. Bao et al.[129] constructed an encoder–decoder and discriminator framework trained on the Corel-1000 dataset[130] to imitate the process of image scrambling and reconstruction in which the parameters of the encoder and decoder are different. However, the cipher pixels were not uniformly distributed, the decrypted images quality and the generalization ability of model were not good, and the plaintext and ciphertext image sensitivities were weak. Bao and Xue[54] analyzed the causes of the weak avalanche effect in the neural network of Cycle-GAN and integrated the traditional diffusion algorithm into the Cycle-GAN-based image encryption methods trained on a dataset of satellite images scraped from Google Maps to enhance the avalanche effect, although its decryption performance was not well.

Table 10 compares the image encryption methods illustrated in Refs. 106, 109, 117, 120, 127–129, and 54 in PSNR and SSIM values between recovered decrypted images and plaintext images, the key space, and the encryption speed per plaintext image with 256×256 resolution. Table 11 compares the image entropy, correlation coefficients of two horizontally, vertically, and diagonally adjacent pixels of the ciphertext image obtained by different methods, NPCR values, and UACI values when these methods face a chosen plaintext attack, especially, changing 1% of pixels of the plaintext image in Refs. 128 and 129. As can be seen in Tables 10 and 11, the end-to-end key generation and image encryption methods based on the deep learning model have a large key space and a quick encryption efficiency. However, correlation coefficients of two horizontally, vertically, and diagonally adjacent pixels of the ciphertext image and the resistance to the chosen plaintext attack need to be further improved generally.

Table 10 Comparison of PSNR and SSIM values between the decrypted image and the plaintext image, key space, and encryption efficiency in different deep learning methods.

Method	PSNR (dB)	SSIM	Key space	Efficiency (s)
Ref. 106 (Sec. 4.1)	32.5516	**0.9456**	10^{135}	0.85
Ref. 109 (Sec. 4.1)	—	1.00	—	—
Ref. 117 (Sec. 4.3)	—	—	—	0.097
Ref. 120 (Sec. 4.4)	—	—	$(2^8)^{196608}$	—
Ref. 127 (Sec. 4.5)	30.1664	0.9081	—	0.044
Ref. 128 (Sec. 4.5)	**37.43**	0.93	$(10^{10})^{2757936}$	**0.07**
Ref. 129 (Sec. 4.5)	27.5087	0.9115	$(2^{32})^{6067459}$	0.6423
Ref. 54 (Sec. 4.5)	33.1800	0.9360	$(2^{32})^{16698307} + (2^8)^{196608}$	—

Table 11 Comparison of different deep learning methods in image entropy, correlation coefficients of two horizontally, vertically, and diagonally adjacent pixels of the ciphertext image, NPCR values, and UACI values facing a chosen plaintext attack.

Method	Image entropy	Horizontal correlation coefficients	Vertical correlation coefficients	Diagonal correlation coefficients	NPCR	UACI
Ref. 106 (Sec. 4.1)	7.9944	−0.0024	0.0012	0.0035	0.9961	**0.3357**
Ref. 109 (Sec. 4.1)	—	—	—	—	0.96	0.33
Ref. 117 (Sec. 4.3)	—	−0.0021	0.0014	0.0031	—	—
Ref. 120 (Sec. 4.4)	**7.9986**	0.0383	0.2259	0.1158	0.9959	0.2319
Ref. 127 (Sec. 4.5)	—	0.0877	0.1379	0.0349	—	—
Ref. 128 (Sec. 4.5)	7.96	—	—	—	0.9421 (change 1% pixels)	—
Ref. 129 (section 4.5)	7.9772	0.0291	0.0363	−0.0233	0.9045 (change 1% pixels)	0.1237 (change 1% pixels)
Ref. 54 (Sec. 4.5)	7.9972	**0.0004**	**0.0005**	**−0.0011**	**0.9964**	0.3349

Some problems exist with end-to-end key generation and image encryption based on the deep learning model. For example, the histogram of the generated key and ciphertext image is not seriously randomly distributed, and as can be seen in Table 11, the UACI values facing chosen ciphertext or plaintext attacks is not high in general. Although Ref. 54 proved that the employment of traditional diffusion techniques to Cycle-GAN-based image encryption methods could enhance the ability against chosen plaintext attack, the complexity of the image encryption methods also increased. Using a deep learning method to take the place of the diffusion algorithm is a possible solution to reduce the time costs. Then the decrypted image quality could be further enhanced. In addition, the deep learning model needs a lot of training time, and the huge amount of model calculations results in a poor encryption/decryption speed. Thus the training and encryption/decryption efficiency of the end-to-end image encryption techniques based on Cycle-GAN could be further improved. Furthermore, because the model is trained according to a specific dataset, the generalization ability of the encryption and decryption model should be further discussed, analyzed, and improved. However, the automatic generation of keys and ciphertext images using deep learning methods has the advantages of convience, large key space and reduced reliance on complex cryptography design knowledge. How to realize the image cryptography on an even more profound level using deep learning models and thus making a breakthrough is still a potential research direction in the future.

4.6 *Image Cryptanalysis Method Based on Deep Learning*

Some progress of image cryptanalysis using deep learning has been made, especially in optical image encryption. The deep learning model is trained with large ciphertext and corresponding plaintext images to learn the ability to crack the optical image encryption method.

Because of the nonlinear operation of phase truncation, the cryptography based on phase truncated Fourier transforms (PTFT)[131] has high robustness against existing attacks.

Figure 15 illustrates the encryption processes of PTFT, assuming that the original image is $f(x)$ FT$(\cdot)$ and IFT$(\cdot)$ are the Fourier transform and inverse Fourier transform, respectively. The Fourier transform is given in Eq. (3). Equation (4) gives the phase truncation operation PT$(\cdot)$. Suppose that $R1(x)$ and $R2(u)$ are a pair of independent random phase masks; the encryption procedure of ciphertext image $g(x)$ is seen in Eqs. (5) and (6):[131]

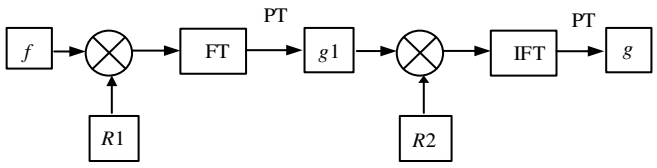

Fig. 15 Process diagram of phase truncation Fourier transform encryption.

$$F(u) = \mathrm{FT}[f(x)] = |F(u)| \exp(i2\pi\varphi(u)), \tag{3}$$

$$\mathrm{PT}[F(u)] = |F(u)|, \tag{4}$$

$$g1(u) = \mathrm{PT}[\mathrm{FT}(f(x) \cdot R1(x))], \tag{5}$$

$$g(x) = \mathrm{PT}[\mathrm{IFT}(g1(u) \cdot R2(u))]. \tag{6}$$

Xu et al.[132] proposed a deep learning method to attack the PTFT encryption system[131] and used a dataset with pairs of plaintext images on the MNIST handwritten dataset[133] and corresponding ciphertext images constructed through the PTFT encryption system to train residual network,[9] which automatically learned the decryption characteristics of the encryption system by reducing the MSE between the decrypted images obtained by the deep learning model with the secret image as shown in Fig. 16. However, the quality of recovered images obtained by this method was not good.

Hai et al.[134] trained a neural network to crack the random phase encoding-based optical cryptosystems.[135] Wu et al.[136] trained a model that involved a module for obtaining the features of the ciphertext image and a module for recovering the plaintext image according to the obtained features with a large numbers of ciphertext-plaintext image pairs to attack the modified diffractive-imaging-based image encryption[137] cryptosystem. Chen et al.[138] demonstrated a CNN trained with a large amount of ciphertext image data encrypted by the joint transform correlation structure and its corresponding plaintext image, directly converting the ciphertext image to the original plaintext image.

Deep learning also shows a certain ability to detect or crack other encryption methods. He et al.[139] mapped the ciphertext images encrypted by the chaos-based image encryption algorithm demonstrated in Ref. 140 into the low-dimensional space and then regenerated visually consistent decrypted images utilizing a deconvolutional generator.

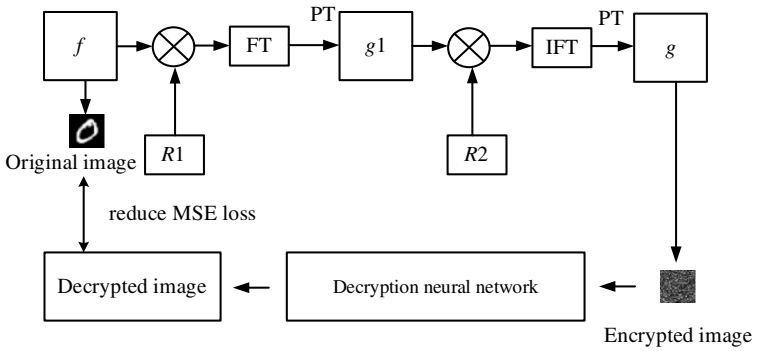

Fig. 16 The training process of deep learning decryption network for phase truncation Fourier transform encryption.

5 Image Authentication

5.1 *Image Forgery Detection*

Bondi et al.[141] employed a CNN trained on the Dresden image database[142] to extract characteristic camera model features, which were analyzed through iterative clustering techniques for image tampering detection and localization using characteristic footprints left on images by different camera models. Elaskily et al.[143] exploited a CNN trained on the MICC-F220,[144] MICC-F2000,[144] MICC-F600,[145] and SATs-130[146] datasets to extract features from images to detect the copy-move forgery. Diallo et al.[147] presented a camera identification CNN model trained with a mixture of different qualities of compressed and uncompressed images on the Dresden dataset[142] for image forgery detection. Patil and Jariwala[63] carried out the intensive and incremental learning phase and then implemented a hybrid CNN to detect the image and video forgery. Bappy et al.[148] introduced a manipulation localization method using resampling features, long-short-term memory cells, and an encoder–decoder network to segment out manipulated regions from nonmanipulated ones. Xiao et al.[149] suggested a splicing forgery detection algorithm with diluted adaptive clustering and a coarse-to-refined CNN trained on the CASIA,[150] COLUMB,[151] and FORENSICS[152] datasets, which cascaded a coarse CNN and a refined CNN and extracted the differences in the image properties between untampered and tampered regions from image patches with different scales. However, the detection only focused on a single tampered region in an image owing to a restriction of the postprocessing approach. Zhang and Ni[153] employed a cross-layer intersection mechanism to dense U-Net[85] for image forgery detection and localization. Biach et al.[154] described an encoder using an architecture that was topologically the same as that of Resnet-50;[9] it analyzed the discriminating characteristics between the manipulated and nonmanipulated regions, and a decoder localized the manipulated regions. However, there were a few poorly detected images, especially on the NIST'16[155] dataset. Moulin and Goel[156] derived locally optimal statistical tests for identifying forgeries and showed a procedure for learning a forgery detector trained on the CIFAR-10[157] and MNIST handwritten datasets. To combat image recapture attacks such as recapturing high-quality images from high-fidelity liquid crystal display screens, Zhu et al.[158] described a recaptured image detection method based on CNN in which the local binary patterns coding coded maps were extracted as the input (Table 12).

F_1-score takes both false negatives and false positives into account. Table 13 shows that Ref. 154 achieves the highest F_1-score and has the best performance in image forgery detection among all compared methods.

Although deep learning has achieved good results in image forgery detection for several types of forgery, there is shortage of large and perfect datasets that include images tampered by methods for more types of forgery and research studies on deep learning forgery detection methods for more complete forgery types.

5.2 *Watermarked Image Generation*

To dynamically adapt image watermarking algorithms, deep learning-based image watermarking schemes have attracted increased attention, and experiments and estimation results have confirmed the advantages of the deep learning mechanisms in image watermarking. Vukotić[159] investigated a new family of transformations based on deep learning networks that were useful in image watermarking. As can be seen in Fig. 17, Kandi et al.[160] proposed an autoencoder CNN for watermark embedding and extraction; the attack layer simulated different attacks, and the strength factor controlled the level of watermarked images robustness versus imperceptibility. Fierro-Radilla et al.[161] demonstrated a zero-watermarking algorithm in which features of the image were gained by the CNN and combined with the watermark sequence using the XOR operation. Ahmadi et al.[162] described two fully CNNs with the residual structure trained on the CIFAR-10 and Pascal VOC2012[93] datasets for watermarks embedding and extraction. Mun et al.[163] exploited a reinforcement learning trained on the BOSSbase dataset for robust and blind watermarking. Zhong et al.[164] introduced an encoder to encode the watermark and input the result into an embedder with the cover image to reach the watermarked image, with the encoder and embedder being two CNNs trained on the ImageNet and CIFAR[157] datasets. Zhang et al.[165]

Table 12 Some deep learning models for image authentication tasks.

Series	References	Characteristics
Deep learning in image forgery detection	Bondi et al. (2017)[141]	Employ a CNN to extract characteristic camera model features that are analyzed through iterative clustering techniques for image tampering detection
	Elaskily et al. (2020)[143]	A CNN is used for detecting the copy-move forgery and original images
	Diallo et al. (2020)[147]	A camera identification CNN model trained with a mixture of different qualities of compressed and uncompressed images
	Pramod et al. (2021)[63]	Ameliorate the image and video forgery detection's efficiency utilizing hybrid CNN
	Bappy et al. (2019)[148]	A manipulation localization method that utilizes resampling features, long-short-term memory cells, an encoder–decoder
	Xiao et al. (2020)[149]	A splicing forgery detection method with diluted adaptive clustering and a coarse-to-refined CNN
	Zhang et al. (2020)[153]	Apply cross-layer intersection mechanism to dense U-Net for image forgery detection and localization
	Biach et al. (2021)[154]	A CNN method based on an encoder/decoder to locate the manipulated regions
	Moulin et al. (2017)[156]	Derive locally optimal statistical tests for identifying forgeries
	Zhu et al. (2019)[158]	A recaptured image detection method based on convolutional neural networks
Deep learning in image watermark generation	Vukotić et al. (2018)[159]	A new family of transformations based on deep learning networks that were useful in image watermarking
	Kandi et al. (2017)[160]	An autoencoder CNN for watermark embedding and extraction
	Fierro-Radilla et al. (2019)[161]	A robust zero-watermarking algorithm
	Ahmadi et al. (2019)[162]	An end-to-end diffusion blind watermarking framework
	Mun et al. (2019)[163]	A reinforcement learning for robust and blind watermarking
	Zhong et al. (2020)[164]	An encoder encodes the watermark and then input to an embedder with the cover image to reach the watermarked image
	Zhang et al. (2021)[165]	A watermarking framework for protecting deep networks
Deep learning in image watermark extraction and detection	Li et al. (2021)[167]	A single-exposure optical image watermarking framework
	Huynh-The et al. (2019)[169]	A blind image watermarking framework based on an encoder–decoder network watermark extraction model
	Li et al. (2018)[170]	A cooperative neural network to recognize the suspected watermark signal
	Hayes et al. (2020)[171]	Resilient signal watermarking via adversarial training
	Chen et al. (2021)[172]	A model based on deep learning technology that accurately identifies the watermark copyright

Table 12 (*Continued*).

Series	References	Characteristics
Deep learning in image watermarking attack	Wang et al. (2021)[173]	Digital image watermark fakers using generative adversarial learning
	Hatoum et al. (2021)[175]	A fully convolutional neural network as a denoising attack on watermarked images
	Sharma et al. (2020)[176]	An adversarial watermarking attack based on a CNN-based autoencoder scheme
Deep learning in image watermark removal	Cheng et al. (2018)[177]	A deep learning model for visible watermark removal task that consists of two parts: watermark detection and removal
	Gandelsman et al. (2019)[178]	A coupled "Deep-Image-Prior" network to remove image watermark
	Hertz et al. (2019)[179]	Estimate the visual motif matte and reconstruct the latent image without opaque and semitransparent visual motifs
	Li et al. (2019)[180]	A watermark processing framework using the conditional GAN
	Pei et al. (2021)[181]	A watermark removal structure including watermark extraction and image inpainting networks
	Cun et al. (2020)[183]	A multitask feature extractor and a watermarked region smoother
	Shafieinejad et al. (2019)[184]	Focus on backdoor-based watermarking
	Chen et al. (2019)[185]	A unified watermark removal framework based on fine-tuning and incorporated with an adaption of the elastic weight consolidation algorithm and unlabeled data augmentation
	William et al. (2021)[186]	A neural network "laundering" algorithm to remove black-box backdoor watermarks from neural networks

Table 13 F_1-score comparisons of different image forgery detection methods on the CASIA[151] and NIST'16[156] datasets.

Datasets	Ref. 153	Ref. 149	Ref. 154
CASIA v1.0	0.5722	0.6758	**0.7362**
NIST'16	0.5140	—	**0.6389**

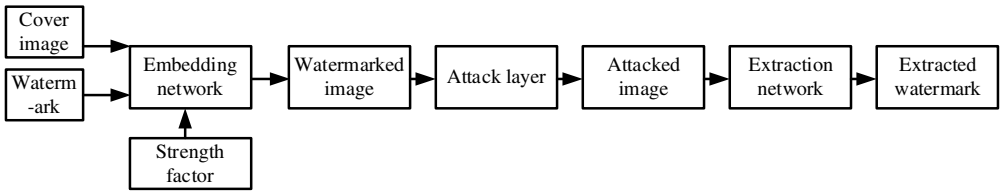

Fig. 17 The watermarking framework of Ref. 160.

exploited an embedding network and an extractor network to embed and gain the watermark, respectively, and a surrogate network to boost the watermark, revealing ability of an extractor network; these were trained on the PASCAL VOC[82] and Chestx-ray8[166] datasets. However, the method was not robust enough to some preprocessing techniques such as random cropping and resizing.

Table 14 Comparison of different deep learning techniques in PSNR between watermarked images and container images, capacity, and BER for the methods recovering watermark information against normal digital image processing operations.

	Robustness to multiattacks				
Method	JPEG compression with factor 10	20% cropping	5% salt and pepper noise	PSNR (dB)	Capacity (bpp)
Ref. 161	**2**	—	—	33.15	3.7×10^{-4}
Ref. 162	—	11.3	—	**44.14**	0.0156
Ref. 163	–	6.61	7.98	38.01	0.0052
Ref. 164	8.16	**0**	**0.97**	39.72	**0.0208**

Table 14 compares the bit error rate (BER) for the method recovering watermark information against normal digital image manipulations (including 20% cropping, 5% pepper and salt noise, and lossy JPEG compression with factor 10), capacity, and the PSNR between the container image and the watermarked image gained by Refs. 161–164. The higher the BER value is, the more robust the watermark recovery ability of the methods is. The higher the PSNR values between the container image and the watermarked image is, the more imperceptible the method is.

As can be seen in Table 14, different methods have their distinctive advantages. For example, Ref. 161 has better resistance in JPEG compression with factor 10, whereas Ref. 162 has a higher PSNR index between watermarked images and original images, and Ref. 164 has a better resistance in 5% salt and pepper noise and 20% cropping and larger capacity. As a result, a deep learning method in image watermarking that has a high resistance to multiattacks and a good imperceptibility of the watermarked image while having a large capacity needs to be further developed in future research.

5.3 *Image Watermark Extraction and Detection*

Li et al.[167] extracted a watermark image from a single-frame watermarked hologram by a conditional GAN trained on the fashion-MNIST[168] and MNIST handwritten datasets. Huynh-The et al.[169] exploited a blind image watermarking framework based on a deep convolutional encoder–decoder network watermark extraction model trained with various attacked watermarked images on the BOSSbase v1.01 database. Li et al.[170] embedded a processed watermark image into the block discrete cosine transform component and used a cooperative neural network to recognize the suspected watermark signal with a single hologram, which improved the transmission efficiency. Hayes et al.[171] learned a transformation-resilient watermark detector trained on the CIFAR-10 and ImageNet datasets to detect watermarks that could be employed in various carriers such as in the image, audio, and video domains. Chen et al.[172] performed a simulated process to generate a large number of distorted watermarks and then collected them to form a training dataset to train a CNN model that could accurately identify the watermark copyright.

5.4 *Image Watermarking Attack*

Wang et al.[173] trained a watermark faker based on U-Net trained on the Caltech256[174] dataset with the input being an original image and the output being a fake watermarked image after preprocessing; a set of paired images of original and watermarked images was obtained by the targeted image watermarking algorithms. However, this method did not perform well at generating the watermark in the frequency domain. Hatoum et al.[175] employed a fully CNN trained on th BOSSbase dataset to denoise watermarked images and destroy the watermarks while preserving a satisfied quality of the denoised images. Sharma and Chandrasekaran[176] implemented an enhanced hybrid watermarking scheme using DWT and singular value decomposition methods and proposed an adversarial attack based on a CNN-based autoencoder scheme trained on the CIFAR-10 database that could produce a perceptually close image.

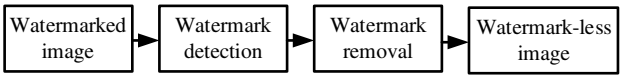

Fig. 18 The general process of deep learning methods for watermark removal.

5.5 *Image Watermark Removal*

As can be seen in Fig. 18, for the visible watermark removal task, deep learning models often consist of two parts:[177] watermark detection and removal. First, the deep learning model detects the watermark object in the watermarked images and then removes the watermark object from the watermarked image to obtain the watermark-less image. Gandelsman et al.[178] proposed a coupled "Deep-Image-Prior" network to remove the image watermark that needed no training examples other than the input image/video. Hertz et al.[179] trained on the Microsoft COCO val2014 dataset[79] and learned to separate the visual motif from the image by estimating the visual motif matte and reconstructing the latent image for blind removal of both opaque and semitransparent motifs. Li et al.[180] suggested a watermark processing framework using the conditional GAN trained on a large-scale visible watermark dataset[177] and the PASCAL VOC2012 dataset for visible watermark removal in a real-world application. The generated watermark-less image had photorealistic quality but not good performance in standard quantitative evaluation metrics such as PSNR. Jiang et al.[181] presented a watermark removal structure consisting of a watermark extraction network that removed the watermark in the watermarked image and an image inpainting network that inpainted the image for a watermark-less image. The two networks were trained on the PASCAL VOC2012 and places2[182] datasets. Cun and Pun[183] introduced a multitask feature extractor and a watermarked region smoother incorporated with multiple perceptual losses trained on the VAL2014 subset of the MSCOCO[79] dataset and a dataset of logos to simulate the procedure of image watermark detection, removal, and refinement. However, when the detection failed or the textures in the watermark and background were similar, the network could not remove the watermark perfectly.

To remove the image watermark generated by the deep learning model, many scholars have proposed their methods from different perspectives. Shafieinejad et al.[184] focused on backdoor-based watermarking, removingd the watermark fully by just relying on public data and proposing an attack that detected whether a model contained a watermark trained on th MNIST and CIFAR-10 datasets. Chen et al.[185] exploited a unified watermark removal framework based on fine-tuning and incorporated it with an adaption of the elastic weight consolidation algorithm and unlabeled data augmentation. William et al.[186] described a neural network "laundering" algorithm to remove black-box backdoor watermarks from neural networks trained on the MNIST and CIFAR-10 datasets.

6 Future Scope

Bringing deep learning methods into the field of image security have solved many problems that cannot be solved by traditional methods. Deep learning image methods need a lot of pretraining time and depend too much on the training datasets, which are its characteristics. Furthermore, being good at using these characteristics or not decides the performance of deep learning models. In the future, the development directions could be summarized into five points.

1. Deep learning models should be designed to take more consideration of the property of image security tasks and balance the model performance of all aspects. For example, to enhance the robustness or antiattack ability of deep learning methods, possible restrictions can be considered and set in advance in the design of the model architecture and training methods. Because the stego-signal representing only a small part in stego-image is not strong, finding a suitable method to enhance the stego-signal-to-noise ratio properly for better performance in image steganalysis is important. Meanwhile, the difference between the tampered image and the original image is very small, and understanding how to better use deep learning according to this property for forgery detection needs to be continually explored. Because the low-avalanche effect of neural network in end-to-end image encryption, considering the diffusion part is essential for the security.

2. The internal working principles of deep learning should be better understood, and the techniques of deep learning models should be develeop for better use. The input and output of deep learning model are usually fixed, and designing a more flexible input and output size contributes to deep learning methods being more widely used in practice. A stronger image features extraction ability will make the action of deep learning more accurate, which is an area that many authors have been studying. Promoting the interpretability of deep learning helps people better design models according to the characteristics of deep learning. Improving the generalization ability of deep learning model will make the deep learning image security methods adapt to images in more scenarios. The large amount of computation of neural networks has been criticized. It is imperative to design a lightweight neural network using knowledge distillation, neural network pruning, and other technologies to reduce the computational complexity, which is conducive to applications especially in industry.
3. There is a shortage of theories for deep learning image security, so establishing and continuously improving the theoretical system of deep learning image security are urgent. Some deep learning methods in image security need to be explained from the view of mathematics, while the targeted tests should be expanded from other angles. A better theoretical basis will guide faster and better development of the field of deep learning image security.
4. Deep learning is a dataset driven technology, but the datasets established for some special tasks are not perfect. It is necessary to establish larger and richer datasets for special tasks. For example, the dataset used in the field of image forgery detection needs to include images tampered by various tampering methods. There is an urgent need to establish a more comprehensive dataset to promote the rapid development and application of diversified tests of deep learning on special tasks.
5. Deep learning should be taken into other areas of image security to solve more traditional image security problems. For example, using neural network to exchange keys in image cryptography and neural network image homomorphic encryption are interesting research directions.

Above all, there are still many challenges and development opportunities in the field of image security for deep learning. It is significant to develop deep learning in image security.

7 Conclusion

This paper describes deep learning with respect to image steganography to generate the cover image, the stego-image, embedding change probabilities, coverless steganography, and steganalysis. As a result, we know that the image steganography method based on deep learning has reached a good performance in embedding capacity and stego-image imperceptibility quality. However, the robustness of deep learning-based image steganography algorithms needs further detailed testing, analysis, and improvements; the embedded secret data extraction should be more accurate; and the input and output size should be more flexible in future studies. Furthermore, this paper combines and compares deep learning techniques used in image cryptography as concerns in image compression, image resolution improvement, image object detection and classification, key generation, end-to-end image encryption, and image cryptoanalysis. We find that end-to-end key generation and image encryption based on the deep learning model have advantages in large key space and automatic generation, with a reduced reliance on complex cryptography design knowledge. Furthermore, the improvement in the randomness of generated ciphertext image and keys, quality of decrypted image, generalization ability and efficiency of the encryption and decryption model, and resistance of facing chosen ciphertext or plaintext attack are still significant research directions for the future. In addition, this paper relates deep learning methods in image authentication from image forgery detection, watermarked image generation, image watermark extraction and detection, image watermarking attack, and image watermark removal and predicts the development of an image watermarking method based on deep learning that has a high resistance to multiattacks, good imperceptibility of the watermarked image, and a large capacity, which are future research directions for this

topic. Finally, we summarize three future development directions through the whole analysis that have enlightening significance for relevant researchers.

Acknowledgements

This project was supported by the Natural Science Foundation of Xizang Autonomous Region of China (Grant No. XZ202001ZR0048G). The authors declare no conflicts of interest.

References

1. W. Liu et al., "SSD: single shot multibox detector," *Lect. Notes Comput. Sci.* **9905**, 21–37 (2016).
2. J. Redmon et al., "You only look once: unified, real-time object detection," in *IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, IEEE, pp. 779–788 (2016).
3. J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," in *IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, IEEE, pp. 6517–6525 (2017).
4. J. Redmon and A. Farhadi, "YOLOv3: an incremental improvement," https://arxiv.org/abs/1804.02767 (2018).
5. A. Bochkovskiy, C. Y. Wang, and H. Y. Liao, "YOLOv4: optimal speed and accuracy of object detection," https://arxiv.org/abs/2004.10934 (2020).
6. K. M. Hosny, M. A. Kassem, and M. M. Fouad, "Classification of skin lesions into seven classes using transfer learning with AlexNet," *J. Digital Imaging* **33**, 1325–1334 (2020).
7. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. NIPS*, pp. 1097–1105 (2012).
8. N. Dey et al., "Customized VGG19 architecture for pneumonia detection in chest X-rays," *Pattern Recognit. Lett.* **143**, 67–74 (2021).
9. K. He et al., "Deep residual learning for image recognition," in *IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, IEEE, pp. 770–778 (2016).
10. R. Ke et al., "Multi-task deep learning for image segmentation using recursive approximation tasks," *IEEE Trans. Image Process.* **30**, 3555–3567 (2021).
11. J. Zhang et al., "LCU-Net: a novel low-cost U-Net for environmental microorganism image segmentation," *Pattern Recognit.* **115**, 107885 (2021).
12. B. Olimov et al., "FU-Net: fast biomedical image segmentation model based on bottleneck convolution layers," *Multimedia Syst.* **27**(4), 637–650 (2021).
13. Y. Lei et al., "Echocardiographic image multi-structure segmentation using Cardiac-SegNet," *Med. Phys.* **48**(5), 2426–2473 (2021).
14. L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, IEEE, pp. 2414–2423 (2016).
15. Z. Ma et al., "Image style transfer with collection representation space and semantic-guided reconstruction," *Neural Networks* **129**, 123–137 (2020).
16. C. T. Lin et al., "GAN-based day-to-night image style transfer for nighttime vehicle detection," *IEEE Trans. Intell. Transp. Syst.* **22**(2), 951–963 (2021).
17. Z. Wang et al., "GLStyleNet: exquisite style transfer combining global and local pyramid features," *IET Comput. Vision* **14**, 575–586 (2020).
18. C. Tian et al., "Designing and training of a dual CNN for image denoising," *Knowl.-Based Syst.* **226**, 106949 (2021).
19. H. Xia et al., "Combination of multi-scale and residual learning in deep CNN for image denoising," *IET Image Process.* **14**, 2013–2019 (2020).
20. P. Jia et al., "Point spread function modelling for wide-field small-aperture telescopes with a denoising autoencoder," *Mon. Not. R. Astron. Soc.* **493**(1), 651–660 (2020).
21. D. Liu et al., "View synthesis-based light field image compression using a generative adversarial network," *Inf. Sci.* **545**, 118–131 (2021).
22. H. Liu et al., "Deep learning-based picture-wise just noticeable distortion prediction model for image compression," *IEEE Trans. Image Process.* **29**, 641–656 (2020).

23. I. Schiopu and A. Munteanu, "Residual-error prediction based on deep learning for lossless image compression," *Electron. Lett.* **54**, 1032–1034 (2018).
24. J. Fridrich and J. Kodovsky, "Rich models for steganalysis of digital images," *IEEE Trans. Inf. Forensics Secur.* **7**(3), 868–882 (2012).
25. T. Denemark et al., "Selection-channel-aware rich model for steganalysis of digital images," in *Proc. IEEE Int. Workshop Inf. Forensics Secur. (WIFS)*, pp. 48–53 (2014).
26. G. Xu, H. Wu, and Y. Shi, "Structural design of convolutional neural networks for steganalysis," *IEEE Signal Process. Lett.* **23**(5), 708–712 (2016).
27. M. Jarno, "LSB matching revisited," *IEEE Signal Process. Lett.* **13**(5), 285–287 (2006).
28. T. Pevný, T. Filler, and P. Bas, "Using high-dimensional image models to perform highly undetectable steganography," *Lect. Notes Comput. Sci.* **6387**, 161–177 (2010).
29. W. Luo, F. Huang, and J. Huang, "Edge adaptive image steganography based on LSB matching revisited," *IEEE Trans. Inf. Forensics Secur.* **5**(2), 201–214 (2010).
30. V. Holub, J. Fridrich, and T. Denemark, "Universal distortion function for steganography in an arbitrary domain," *EURASIP J. Inf. Secur.* **2014**(1), 1–13 (2014).
31. B. Li et al., "A new cost function for spatial image steganography," in *Proc. IEEE ICIP*, Paris, pp. 4206–4210 (2014).
32. R. Zhang, S. Dong, and J. Liu, "Invisible steganography via generative adversarial networks," *Multimedia Tools Appl.* **78**(7), 8559–8575 (2019).
33. J. Yang et al., "An embedding cost learning framework using GAN," *IEEE Trans. Inf. Forensics Secur.* **15**, 839–851 (2019).
34. P. Bas, T. Filler, and T. Pevný, "Break our steganographic system: the ins and outs of organizing BOSS," in *Proc. 13th Int. Conf. Inf. Hiding*, Prague, pp. 59–70 (2011).
35. S. Jiao et al., "Known-plaintext attack and ciphertext-only attack for encrypted single-pixel imaging," *IEEE Access* **7**, 119557–119565 (2019).
36. S. K. Rajput and N. K. Nishchal, "Known-plaintext attack on encryption domain independent optical asymmetric cryptosystem," *Opt. Commun.* **309**(Complete), 231–235 (2013).
37. I. E. Hanouti, H. E. Fadili, and K. Zenkouar, "Cryptanalysis of an embedded systems' image encryption," *Multimedia Tools Appl.* **80**(9), 13801–13820 (2021).
38. E. J. Yoon and K. Y. Yoo, "Cryptanalysis of a modulo image encryption scheme with fractal keys," *Opt. Lasers Eng.* **48**(7–8), 821–826 (2010).
39. J. J. Chen et al., "Memristor-based hyper-chaotic circuit for image encryption," *Chin. Phys. B* **29**(11), 299–310 (2020).
40. Y. Bentoutou et al., "An improved image encryption algorithm for satellite applications," *Adv. Space Res.* **66**, 176–192 (2020).
41. M. Naim, A. A. Pacha, and C. Serief, "A novel satellite image encryption algorithm based on hyperchaotic systems and Josephus problem," *Adv. Space Res.* **67**(7), 2077–2103 (2021).
42. M. D. Zhao et al., "Image encryption based on fractal-structured phase mask in fractional Fourier transform domain," *J. Opt.* **20**(4), 045703 (2018).
43. F. A. Yatish and N. K. Nishchal, "Optical image encryption using triplet of functions," *Opt. Eng.* **57**(3), 033103 (2018).
44. Y. Shi et al., "Multiple-image double-encryption via 2D rotations of a random phase mask with spatially incoherent illumination," *Opt. Express* **27**(18), 26050–26059 (2019).
45. P. Ping, F. Xu, and Z. J. Wang, "Color image encryption based on two-dimensional cellular automata," *Int. J. Mod. Phys. C* **24**(10), 1350071 (2013).
46. A. Souyah and K. M. Faraoun, "An image encryption scheme combining chaos-memory cellular automata and weighted histogram," *Nonlinear Dyn.* **86**, 639–653 (2016).
47. P. Naskar et al., "A robust image encryption scheme using chaotic tent map and cellular automata," *Nonlinear Dyn.* **100**, 2877–2898 (2020).
48. H. S. Li et al., "Quantum image encryption based on phase-shift transform and quantum Haar wavelet packet transform," *Mod. Phys. Lett. A* **34**(26), 1950214 (2019).
49. C. Hou, X. Liu, and S. Feng, "Quantum image scrambling algorithm based on discrete Baker map," *Mod. Phys. Lett. A* **35**(17), 2050145 (2020).
50. G. Ye, K. Jiao, and X. Huang, "Quantum logistic image encryption algorithm based on SHA-3 and RSA," *Nonlinear Dyn.* **104**(3), 2807–2827 (2021).

51. M. Guan, X. Yang, and W. Hu, "Chaotic image encryption algorithm using frequency-domain DNA encoding," *IET Image Process.* **13**, 1535–1539 (2019).
52. X. Chai et al., "A novel image encryption algorithm based on the chaotic system and DNA computing," *Int. J. Mod. Phys. C* **28**(5), 1750069 (2017).
53. D. Ravichandran et al., "An efficient medical image encryption using hybrid DNA computing and chaos in transform domain," *Med. Biol. Eng. Comput.* **59**, 589–605 (2021).
54. Z. Bao and R. Xue, "Research on the avalanche effect of image encryption based on the Cycle-GAN," *Appl. Opt.* **60**(18), 5320–5334 (2021).
55. F. H. Hsu et al., "Visible watermarking with reversibility of multimedia images for ownership declarations," *J. Supercomput.* **70**(1), 247–268 (2014).
56. J. C. Patra, J. E. Phua, and C. Bornand, "A novel DCT domain CRT-based watermarking scheme for image authentication surviving jpeg compression," *Digital Signal Process.* **20**(6), 1597–1611 (2010).
57. T. S. Nguyen, "Fragile watermarking for image authentication based on DWT-SVD-DCT techniques," *Multimedia Tools Appl.* **80**, 25107–25119 (2021).
58. W. Zhang and F. Y. Shih, "Semi-fragile spatial watermarking based on local binary pattern operators," *Opt. Commun.* **284**(16–17), 3904–3912 (2011).
59. Y. Pathak, Y. K. Jain, and S. Dehariya, "A secure transmission of medical images by single label SWT and SVD based non-blind watermarking technique," *Infocomp J. Comput. Sci.* **14**(1), 50–59 (2015).
60. B. Z. Li and Y. Guo, "Blind image watermarking method based on linear canonical wavelet transform and QR decomposition," *IET Image Process.* **10**(10), 773–786 (2016).
61. A. Sleit et al., "An enhanced semi-blind. DWT-SVD-based watermarking technique for digital images," *Imaging Sci. J.* **60**(1), 29–38 (2012).
62. H. Agarwal, P. Atrey, and B. Raman, "Image watermarking in real oriented wavelet transform domain," *Multimedia Tools Appl.* **74**(23), 10883–10921 (2015).
63. S. P. Patil and K. N. Jariwala, "Improving the efficiency of image and video forgery detection using hybrid convolutional neural networks," *Int. J. Uncertain. Fuzz. Knowl.-Based Syst.* **29**(Supp1), 101–117 (2021).
64. A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," https://arxiv.org/abs/1511.06434 (2015).
65. D. Volkhonskiy, B. Borisenko, and E. Burnaev, "Generative adversarial networks for image steganography," in *ICLR 2017 Open Rev.* (2017).
66. Z. Liu et al., "Deep learning face attributes in the wild," in *IEEE Int. Conf. Comput. Vision (ICCV)*, pp. 3730–3738 (2015).
67. H. Shi et al., "SSGAN: secure steganography based on generative adversarial networks," in *Pac. Rim Conf. Multimedia*, Springer, pp. 534–544 (2017).
68. M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN," https://arxiv.org/abs/1701.07875 (2017).
69. Y. Qian et al., "Deep learning for steganalysis via convolutional neural networks," *Proc. SPIE* **9409**, 94090J (2015).
70. J. Hayes and G. Danezis, "Generating steganographic images via adversarial training," in *NIPS 2017: Neural Inf. Process. Syst.*, pp. 4–9 (2017).
71. S. Baluja, "Hiding images within images," *IEEE Trans. Pattern Anal. Mach. Intell.* **42**(7), 1685–1697 (2020).
72. S. Chattopadhyay, P. P. Acharjya, and C. Koner, "A deep learning approach to implement slab based image steganography algorithm of RGB images," in *3rd Int. Conf. Invent. Comput. Technol. (ICICT)* (2018).
73. J. Zhu et al., "Hidden: hiding data with deep networks," in *Proc. Eur. Conf. Comput. Vision (ECCV)*, Springer, pp. 657–672 (2018).
74. D. Hu et al., "A novel image steganography method via deep convolutional generative adversarial networks," *IEEE Access* **6**, 38303–38314 (2018).
75. L. Bossard, M. Guillaumin, and L. V. Gool, "Food-101—mining discriminative components with random forests," *Lect. Notes Comput. Sci.* **8694**, 446–461 (2014).

76. K. A. Zhang et al., "SteganoGAN: high capacity image steganography with GANs," https://arxiv.org/abs/1901.03892v2 (2019).
77. Z. Wang et al., "HidingGAN: high capacity information hiding with generative adversarial network," *Comput. Graphics Forum* **38**(7), 393–401 (2019).
78. X. Duan et al., "High-capacity information hiding based on residual network," *IETE Tech. Rev.* **38**(1), 172–183 (2021).
79. T. Y. Lin et al., "Microsoft COCO: common objects in context," in *Proc. Eur. Conf. Comput. Vision*, Zurich (2014).
80. J. Deng et al., "ImageNet: a large-scale hierarchical image database," in *IEEE Conf. Comput. Vision and Pattern Recognit.*, IEEE, pp. 248–255 (2009).
81. G. B. Huang et al., *Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments*, University of Massachusetts, Amherst, Massachusetts (2007).
82. M. Everingham et al., "The Pascal visual object classes (VOC) challenge," *Int. J. Comput. Vision* **88**(2), 303–338 (2010).
83. Z. Wang et al., "Image quality assessment: from error measurement to structural similarity," *IEEE Trans. Image Process.* **13**, 600–612 (2004).
84. W. Tang et al., "Automatic steganographic distortion learning using a generative adversarial network," *IEEE Signal Process. Lett.* **24**(10), 1547–1551 (2017).
85. O. Ronneberger, P. Fischer, and T. Brox, "U-Net: convolutional networks for biomedical image segmentation," *Lect. Notes Comput. Sci.* **9351**, 234–241 (2015).
86. W. Tang et al., "An automatic cost learning framework for image steganography using deep reinforcement learning," *IEEE Trans. Inf. Forensics Secur.* **16**, 952–967 (2021).
87. X. Duan et al., "Coverless steganography for digital images based on a generative model," *Comput. Mater. Contin.* **55**(3), 483–493 (2018).
88. Y. Luo et al., "Coverless image steganography based on multi-object recognition," *IEEE Trans. Circuits Syst. Video Technol.* **31**, 2779–2791 (2020).
89. Q. Liu et al., "Coverless steganography based on image retrieval of DenseNet features and DWT sequence mapping," *Knowl.-Based Syst.* **192**, 105375 (2019).
90. H. Gao, L. Zhuang, and M. D. V. Laurens, "Densely connected convolutional networks," in *IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, IEEE, pp. 2261–2269 (2017).
91. S. Zhang et al., "An image style transfer network using multilevel noise encoding and its application in coverless steganography," *Symmetry* **11**(9), 1152 (2019).
92. Z. Zhou et al., "Faster-RCNN based robust coverless information hiding system in cloud environment," *IEEE Access* **7**, 179891–179897 (2019).
93. M. Everingham et al., "The PASCAL visual object classes challenge 2012 (VOC2012) results," 2012, http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.
94. X. Duan et al., "A coverless steganography method based on generative adversarial network," *J. Image Video Proc.* **2020**, 18 (2020).
95. J. Yang et al., "Steganalysis based on awareness of selection-channel and deep learning," *Lect. Notes Comput. Sci.* **10431**, 263–272 (2017).
96. J. Zeng et al., "Large-scale jpeg steganalysis using hybrid deep-learning framework," *IEEE Trans. Inf. Forensics Secur.* **13**(5), 1200–1214 (2017).
97. Y. Jian, J. Ni, and Y. Yang, "Deep learning hierarchical representations for image steganalysis," *IEEE Trans. Inf. Forensics Secur.* **12**(11), 2545–2557 (2017).
98. M. Yedroudj, F. Comby, and M. Chaumont, "Yedroudj-Net: an efficient CNN for spatial steganalysis," in *IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, IEEE, pp. 2092–2096 (2018).
99. R. Zhang et al., "Efficient feature learning and multi-size image steganalysis based on CNN," https://arxiv.org/abs/1807.11428 (2018).
100. M. Boroumand, M. Chen, and J. Fridrich, "Deep residual network for steganalysis of digital images," *IEEE Trans. Inf. Forensics Secur.* **14**(5), 1181–1193 (2019).
101. T. S. Reinel et al., "GBRAS-Net: a convolutional neural network architecture for spatial image steganalysis," *IEEE Access* **9**, 14340–14350 (2021).

102. F. Liu et al., "Image steganalysis via diverse filters and squeeze-and-excitation convolutional neural network," *Mathematics* **9**(2), 189 (2021).
103. A. I. Iskanderani et al., "Artificial intelligence-based digital image steganalysis," *Secur. Commun. Networks* **2021**(11), 1–9 (2021).
104. A. Singhal and P. Bedi, "Multi-class blind steganalysis using deep residual networks," *Multimedia Tools Appl.* **80**, 13931–13956 (2021).
105. V. Holub and J. Fridrich, "Designing steganographic distortion using directional filters," in *IEEE Int. Workshop Inf. Forensics Secur., WIFS'2012*, IEEE, pp. 234–239 (2012).
106. W. Chen, Y. Guo, and S. W. Jing, "General image encryption algorithm based on deep learning compressed sensing and compound chaotic system," *Acta Phys. Sin.* **69**(24), 240502 (2020).
107. Q. S. Lian et al., "A compressed sensing algorithm based on multi-scale residual reconstruction network," *Acta Autom. Sin.* **45**(11), 2082–2091 (in Chinese) (2019).
108. F. Hu et al., "An image compression and encryption scheme based on deep learning," https://arxiv.org/abs/1608.05001 (2016).
109. K. M. A. Suhail and S. Sankar, "Image compression and encryption combining autoencoder and chaotic logistic map," *Iran J. Sci. Technol. Trans. Sci.* **44**, 1091–1100 (2020).
110. C. T. Selvi, J. Amudha, and R. Sudhakar, "Medical image encryption and compression by adaptive sigma filterized synorr certificateless signcryptive Levenshtein entropy-coding-based deep neural learning," *Multimedia Syst.* **27**, 1059–1074 (2021).
111. L. Zhang et al., "Optical image compression and encryption transmission-based on deep learning and ghost imaging," *Appl. Phys. B* **126**(1), 061802–993 (2019).
112. M. Grubinger et al., "The IAPR TC-12 benchmark: a new evaluation resource for visual information systems," in *Int. Workshop OntoImage*, pp. 13–23 (2006).
113. J. Chen, X. W. Li, and Q. H. Wang, "Deep learning for improving the robustness of image encryption," *IEEE Access* **7**, 181083–181091 (2019).
114. K. Ma et al., "Waterloo exploration database: new challenges for image quality assessment models," *IEEE Trans. Image Process.* **26**(2), 1004–1016 (2017).
115. X. Zhao et al., "Application of face image detection based on deep learning in privacy security of intelligent cloud platform," *Multimedia Tools Appl.* **79**, 16707–16718 (2020).
116. B. A. Y. Alqaralleh et al., "Blockchain-assisted secure image transmission and diagnosis model on Internet of Medical Things Environment," *Pers Ubiquit Comput.* (2021).
117. M. Asgari-Chenaghlu et al., "C_y: Chaotic yolo for user intended image encryption and sharing in social media," *Inf. Sci.* **542**, 212–227 (2021).
118. X. Li et al., "Research on iris image encryption based on deep learning," *J. Image Video Proc.* **2018**(1), 1–10 (2018).
119. L. Debiasi and A. Uhl, "Techniques for a forensic analysis of the CASIA-IRIS V4 database," in *3rd Int. Workshop Biometrics and Forensics (IWBF 2015)* (2015).
120. Y. Ding et al., "DeepKeyGen: a deep learning-based stream cipher generator for medical image encryption and decryption," *IEEE Trans. Neural Networks Learn. Syst.* (early access) (2021).
121. S. Jaeger et al., "Two public chest X-ray datasets for computer-aided screening of pulmonary diseases," *Quant. Imaging Med. Surg.* **4**, 475–477 (2014).
122. J. Jin and K. Kim, "3D CUBE algorithm for the key generation method: applying deep neural network learning-based," *IEEE Access* **8**, 33689–33702 (2020).
123. S. R. Maniyath and V. Thanikaiselvan, "An efficient image encryption using deep neural network and chaotic map," *Microprocess. Microsyst.* **77**, 103134 (2020).
124. U. Erkan et al., "An image encryption scheme based on chaotic logarithmic map and key generation using deep CNN," https://arxiv.org/abs/2012.14156v1 (2020).
125. A. Fratalocchi et al., "NIST-certified secure key generation via deep learning of physical unclonable functions in silica aerogels," *Nanophotonics* **10**(1), 457–464 (2021).
126. J. Y. Zhu et al., "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *IEEE Int. Conf. Comput. Vision*, IEEE, pp. 2223–2232 (2017).
127. J. Li, J. Zhou, and X. Di, "A learning optical image encryption scheme based on CycleGAN," *J. Jilin Univ. (Eng. Technol. Ed.)* **51**(3), 1060–1066 (in Chinese) (2021).

128. Y. Ding et al., "DeepEDN: a deep-learning-based image encryption and decryption network for internet of medical things," *IEEE Internet Things J.* **8**(3), 1504–1518 (2021).
129. Z. Bao, R. Xue, and Y. D. Jin, "Image scrambling adversarial autoencoder based on the asymmetric encryption," *Multimedia Tools Appl.* **80**(18), 28265–28301 (2021).
130. J. Z. Wang, J. Li, and G. Wiederhold, "SIMPLIcity: semantics-sensitive integrated matching for picture libraries," *IEEE Trans. Pattern Anal. Mach. Intell.* **23**(9), 947–963 (2001).
131. W. Qin and X. Peng, "Asymmetric cryptosystem based on phase-truncated Fourier transforms," *Opt. Lett.* **35**(2), 118–120 (2010).
132. Z. Xu et al., "Attacking the asymmetric cryptosystem based on phase truncated Fourier transforms by deep learning," *Acta Phys. Sin.* **70**(14), 226–232 (2021).
133. D. Ciresan, U. Meier, and J. Schmidhuber, "Multi-column deep neural networks for image classification," in *IEEE Int. Conf. Comput. Vision and Pattern Recognit. (CVPR)*, Providence, Rhode Island, pp. 3642–3649 (2012).
134. H. Hai et al., "Cryptanalysis of random-phase-encoding-based optical cryptosystem via deep learning," *Opt. Express* **27**(15), 21204–21213 (2019).
135. P. Refregier and B. Javidi, "Optical image encryption based on input plane and Fourier plane random encoding," *Opt. Lett.* **20**(7), 767–769 (1995).
136. C. Wu et al., "Cryptoanalysis of the modified diffractive-imaging-based image encryption by deep learning attack," *J. Mod. Opt.* **67**(17), 1398–1409 (2020).
137. Q. A. Gong et al., "Modified diffractive-imaging-based image encryption," *Opt. Lasers Eng.* **121**, 66–73 (2019).
138. L. Chen et al., "Plaintext attack on joint transform correlation encryption system by convolutional neural network," *Opt. Express* **28**(19), 28154–28163 (2020).
139. C. He et al., "A deep learning based attack for the chaos-based image encryption," https://arxiv.org/abs/1907.12245 (2019).
140. Z. H. Guan, F. Huang, and W. Guan, "Chaos-based image encryption algorithm," *Phys. Lett. A* **346**(1–3), 153–157 (2005).
141. L. Bondi et al., "Tampering detection and localization through clustering of camera-based CNN features," in *IEEE Conf. Comput. Vision and Pattern Recognit. Workshops.*, pp. 1855–1864 (2017).
142. T. Gloe and R. Ğohme, "The Dresden image database for benchmarking digital image forensics," *J. Digital Forensic Pract.* **3**, 150–159 (2010).
143. M. A. Elaskily et al., "A novel deep learning framework for copy-move forgery detection in images," *Multimedia Tools Appl.* **79**, 19167–19192 (2020).
144. I. Amerini et al., "A SIFT-based forensic method for copy–move attack detection and transformation recovery," *IEEE Trans. Inf. Forensics. Secur.* **6**(3), 1099–1110 (2011).
145. I. Amerini et al., "Copy-move forgery detection and localization by means of robust clustering with J-linkage," *Signal Process.: Image Commun.* **28**(6), 659–669 (2013).
146. V. Christlein et al., "An evaluation of popular copy-move forgery detection approaches," *IEEE Trans. Inf. Forensics Secur.* **7**(6), 1841–1854 (2012).
147. B. Diallo et al., "Robust forgery detection for compressed images using CNN supervision," *Forensic Sci. Int.: Rep.* **2**, 100112 (2020).
148. J. H. Bappy et al., "Hybrid LSTM and encoder–decoder architecture for detection of image forgeries," *IEEE Trans. Image Process.* **28**(7), 3286–3300 (2019).
149. B. Xiao et al., "Image splicing forgery detection combining coarse to refined convolutional neural network and adaptive clustering," *Inf. Sci.* **511**, 172–191 (2020).
150. J. Dong and W. Wang, "CASIA tampered image detection evaluation (tide) database v1.0 and v2.0," 2011, http://forensics.idealtest.org/.
151. Y. Hsu and S. Chang, "Detecting image splicing using geometry invariants and camera characteristics consistency," in *IEEE Int. Conf. Multimedia and Expo*, IEEE, pp. 549–552 (2006).
152. "Forensics," 2014, https://signalprocessingsociety.org/newsletter/2014/01/ieeeifs-tc-image-forensics-challenge-website-new-submissions.
153. R. Zhang and J. Ni, "A dense U-Net with cross-layer intersection for detection and localization of image forgery," in *ICASSP 2020-2020 IEEE Int. Conf. Acoust Speech and Signal Process. (ICASSP)*, IEEE, pp 2982–2986 (2020).

154. F. Z. E. Biach et al., "Encoder–decoder based convolutional neural networks for image forgery detection," *Multimedia Tools Appl.*, pre publish (2021).
155. Nist, "Nimble," 2016, https://www.nist.gov/sites/default/files/documents/2016/11/30/shouldibelieveornot.pdf.
156. P. Moulin and A. Goel, "Locally optimal detection of adversarial inputs to image classifiers," in *IEEE Int. Conf. Multimedia & Expo Workshops (ICMEW)*, IEEE, pp. 459–464 (2017).
157. A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," 2009, http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=0D60E5DD558A91470E0EA1725FF36E0A?doi=10.1.1.222.9220&rep=rep1&type=pdf.
158. N. Zhu, M. Qin, and Y. Yin, "Recaptured image detection based on convolutional neural networks with local binary patterns coding," *Proc. SPIE* **11198**, 1119804 (2019)
159. V. Vukotić, V. Chappelier, and T. Furon, "Are deep neural networks good for blind image watermarking?" in *IEEE Int. Workshop Inf. Forensics and Secur. (WIFS)*, IEEE, pp. 1–7 (2018).
160. H. Kandi, D. Mishra, and S. Gorthi, "Exploring the learning capabilities of convolutional neural networks for robust image watermarking," *Comput. Secur.* **65**(March), 247–268 (2017).
161. A. Fierro-Radilla et al., "A robust image zero-watermarking using convolutional neural networks," in *7th Int. Workshop Biometrics and Forensics (IWBF)*, IEEE, pp. 1–5 (2019).
162. M. Ahmadi et al., "Redmark: framework for residual diffusion watermarking based on deep networks," *Expert Syst. Appl.* **146**, 113157 (2019).
163. S. M. Mun et al., "Finding robust domain from attacks: a learning framework for blind watermarking," *Neurocomputing* **337**(April 14), 191–202 (2019).
164. X. Zhong et al., "An automated and robust image watermarking scheme based on deep neural networks," *IEEE Trans. Multimedia* **23**, 1951–1961 (2021).
165. J. Zhang et al., "Deep model intellectual property protection via deep watermarking," *IEEE Trans. Pattern Anal. Mach. Intell.* (early Access) (2021).
166. X. Wang et al., "ChestX-ray8: hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases," in *CVPR* (2017).
167. J. Li et al., "Single exposure optical image watermarking using a cGAN network," *IEEE Photonics J.* **13**(2), 6900111 (2021).
168. H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms," https://arxiv.org/abs/1708.07747v2 (2017).
169. T. Huynh-The et al., "Robust image watermarking framework powered by convolutional encoder-decoder network," in *Digital Image Comput.: Tech. and Appl. (DICTA)*, IEEE, pp. 1–7 (2019).
170. D. Li et al., "A novel CNN based security guaranteed image watermarking generation scenario for smart city applications," *Inf. Sci.* **479**, 432–447 (2018).
171. J. Hayes et al., "Towards transformation-resilient provenance detection of digital media," https://arxiv.org/abs/2011.07355v1 (2020).
172. Y. Chen, T. Fan, and H. Chao, "WMNet: a lossless watermarking technique using deep learning for medical image authentication," *Electronics* **10**(8), 932–932 (2021).
173. R. Wang et al., "Watermark faker: towards forgery of digital image watermarking," https://arxiv.org/abs/2103.12489 (2021).
174. G. Griffin, A. Holub, and P. Perona, "Caltech-256 object category dataset," 2007, https://authors.library.caltech.edu/7694/.
175. M. W. Hatoum et al., "Using deep learning for image watermarking attack," *Signal Process. Image Commun.* **90**, 116019 (2021).
176. S. S. Sharma and V. Chandrasekaran, "A robust hybrid digital watermarking technique against a powerful CNN-based adversarial attack," *Multimedia Tools Appl.* **79**(43), 32769–32790 (2020).
177. D. Cheng et al., "Large-scale visible watermark detection and removal with deep convolutional networks," in *Chin. Conf. Pattern Recognit. And Comput. Vision (PRCV)*, Springer, pp. 27–40 (2018).

178. Y. Gandelsman, A. Shocher, and M. Irani, "'Double-DIP': unsupervised image decomposition via coupled Deep-Image-Priors," in *IEEE/CVF Conf. Comput. Vision and Pattern Recognit. (CVPR)*, IEEE, pp. 11018–11027 (2019).
179. A. Hertz et al., "Blind visual motif removal from a single image," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, pp. 6858–6867 (2019).
180. X. Li et al., "Towards photo-realistic visible watermark removal with conditional generative adversarial networks," in *Int. Conf. Image and Graphics*, Springer, pp. 345–356 (2019).
181. P. Jiang et al., "Two-stage visible watermark removal architecture based on deep learning," *IET Image Process.* **14**(15), 3819–3828 (2021).
182. B. Zhou et al., "Places: a 10 million image database for scene recognition," *IEEE Trans. Pattern Anal. Mach. Intell.* **40**(6), 1452–1464 (2017).
183. X. Cun and C. M. Pun, "Split then refine: stacked attention-guided ResUNets for blind single image visible watermark removal," https://arxiv.org/abs/2012.07007v1 (2020).
184. M. Shafieinejad et al., "On the robustness of the backdoor-based watermarking in deep neural networks," https://arxiv.org/abs/1906.07745v2 (2019).
185. X. Chen et al., "REFIT: a unified watermark removal framework for deep learning systems with limited data," in *ASIA CCS '21: ACM Asia Conf. Comput. And Commun. Secur.*, ACM (2021).
186. A. William et al., "Neural network laundering: removing black-box backdoor watermarks from deep neural networks," *Comput. Secur.* **106**, 102277 (2021).

Biographies of the authors are not available.