Computers & Security 127 (2023) 103102

Contents lists available at ScienceDirect

Computers & Security

journal homepage: www.elsevier.com/locate/cose

A novel model watermarking for protecting generative adversarial network

Tong Qiao[a,b,c], Yuyan Ma[a], Ning Zheng[a], Hanzhou Wu[d], Yanli Chen[a], Ming Xu[a], Xiangyang Luo[b,*]

[a] School of Cyberspace, Hangzhou Dianzi University, Hangzhou 310018, China
[b] Henan Key Laboratory of Cyberspace Situation Awareness, State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou 450001, China
[c] State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China
[d] School of Communication and Information Engineering, Shanghai University, Shanghai 200444, China

ARTICLE INFO

Article history:
Received 23 June 2022
Revised 2 January 2023
Accepted 10 January 2023
Available online 14 January 2023

Keywords:
Artificial intelligence
IP Protection
Model watermarking
GAN
Deep learning

ABSTRACT

With the advance of deep learning, it definitely has achieved the unprecedented success in the community of artificial intelligence. However, the issue of the intellectual property (IP) protection towards deep learning model is usually ignored, which largely threats the interests of the model owner. Currently, although a few schemes of model watermarking have been continuously proposed, in order to protect the specific neural network designed for detection or classification task, most of them are hardly directly applicable to generative adversarial networks (GAN). To our knowledge, the GAN model has plays more and more important role in the computer vision, such as image-to-image translation, text-to-image translation, image inpainting and etc., which remarkably improves the capability of image generation. Similarly, the malicious attackers possibly steal a trained GAN model to infringe the IP of the true model owner. To address that challenging issue, it is proposed to establish the framework of model watermarking towards GAN model. In particular, we first establish the trigger set by combining the watermark label with the verification image. Next, the watermarked generator is efficiently trained on the premise of preserving the original model performance. Finally, only relying on the correct watermark label, the synthetic watermark can be successfully triggered by the model owner for IP protection. The extensive experiments have verified the effectiveness and generalization of our designed method, which can easily be applicable to the benchmark GAN models such as WGAN-GP, ProGAN and StyleGAN2. Moreover, our proposed model watermark is robust enough to resist against the mainstream attacks, such as parameter fine-tuning and model pruning.

1. Introduction

Recently, deep learning (DL) has achieved breakthrough success in the field of artificial intelligence (He et al., 2016; Krizhevsky et al., 2012; LeCun et al., 1989), which has also received increasing attention from IT industry, such as *Google* and *Huawei*. With the advanced DL models, many high-tech companies are committed to providing consumers with intelligent products and higher-quality services. Noticeably, as a key component in products or services, a commercialized DL model is not easy to be built, which not only requires large training datasets and expensive computation resources but high budget from companies. Meanwhile, the commercial value of DL models makes it coveted by adversarial attackers. For example, attackers can utilize model inversion attack (Papernot et al., 2017; Tramèr et al., 2016) or membership inference attack (Juuti et al., 2019; Yu et al., 2020) to easily steal the structure and parameters of the model through accessing the neural network model remotely. The DL model is possibly illegally sold to the unauthorized owner, which immensely infringes the intellectual property (IP) of the authorized owners. Therefore, how to protect the IP of the DL model has been an urgent problem to be solved in both academia and industry. To this end, the studies have been carried out to protect the IP of the DL model, referring to as *model watermarking*. In fact, the digital watermarking has been advanced for many years, which is used for protecting the ownership of the digital media by embedding the watermark into the protected media. As it is, the

* Corresponding author.
E-mail addresses: tong.qiao@hdu.edu.cn (T. Qiao), xiangyangluo@126.com (X. Luo).

https://doi.org/10.1016/j.cose.2023.103102
0167-4048/

model watermarking plays the very similar role. Encouragingly, embedding watermarks to DL models is an attractive solution to solve the problem of IP protection for DL models. According to the embedding manners of model watermarking, we can basically categorize the current methods into three types: *Embedding Watermark into Parameters of Model, Embedding Watermark into Trigger Sets*, and *Embedding Watermark into Outputs of Model*, which will be specifically elaborated in Section 2.

To the best of our knowledge, one of the most popular DL models, referring to as generative adversarial networks (GAN), has been widely adopted, in order to create photorealistic images, which remarkably improves the capability of image generation, such as image-to-image translation (Richardson et al., 2021; Zhu et al., 2017), text-to-image translation (Gou et al., 2020; Zhang et al., 2017), image inpainting (Romero et al., 2022; Yu et al., 2018) and etc. Meanwhile, GAN is receiving increasing attention from researchers and is being more widely deployed in commercial products. However, on the down side, once the GAN model is maliciously and illicitly stolen by the unauthorized third party, the IP of the original GAN model owner cannot be effectively protected, leading to irretrievable economic loss. In fact, different from the traditional DL models, GAN is a special type of DL model in which two typical networks are trained together. One of them focuses on data generation and the other one is used for discrimination. Nevertheless, most current IP protection schemes designed for the DL models of detection or classification task, cannot be directly applied to the GAN model of image generation task. In such scenario, protecting the GAN model has become a knotty task for many industries. Therefore, in this paper, we are committed to proposing a novel watermarking framework, in order to protect the IP of GAN model. For clarity, this paper makes the following contributions:

- We propose a novel GAN watermarking framework, which is capable of preserving the model performance without changing the original network structure.
- The proposed model watermark can realize the remote ownership verification of the authorized model user for IP protection.
- The extensive experimental results verify the effectiveness of the proposed model watermarking scheme, which is also robust enough to resist against parameter fine-tuning and model pruning attacks.

The rest of the paper is organized as follows. Section 3 presents the proposed framework of GAN model watermarking, in which the main stages, referring to as trigger set generation, embedding and verification procedures, are specifically elaborated. Next, Section 4 demonstrate the large-scale experimental results on the realistic datasets. In Section 5, we mainly discuss the superiority of the proposed method. Finally, Section 6 concludes this paper.

2. Related work

In recent years, many methods of embedding watermarks into DL models have been proposed, which are generally classified into the following three categories.

- *Embedding Watermark into Parameters of Model:* Notably, embedding the watermark into the model parameters is usually called white-box watermark. Model watermark was first proposed by Uchida et al. (2017). Through adding the additional regularization term to loss function during neural network training, the model parameters are modified for watermark embedding. However, this method cannot completely resist watermark overwriting attack. Inspired by the method (Uchida et al., 2017), further improvements have been achieved by proposing a fine-tuning scheme with a compensation mechanism (Feng and Zhang, 2020), which makes it difficult for the watermark to be easily overwritten while cannot resist ambiguity attacks. Wang and Kerschbaum (2019) pointed out watermark (Uchida et al., 2017) can be easily detected by the histogram of the weights. Thus Wang and Kerschbaum (2019) proposed to use adversarial network architecture in the watermark embedding phase, in which the generated sample is taken directly from the model parameters. And the discriminator detects whether the model is watermarked. Different from prior studies, a passport-based deep neural networks (DNN) ownership verification method is proposed (Fan et al., 2019). That indeed increases the connection between network performance and correct passports by embedding passports in special normalization layers, resulting in enhancement of the watermarks resisting against ambiguity attack. Moreover, an extended method is further proposed by Zhang et al. (2020b), where the passport could be used for most normalization layers and has better generalization ability.
- *Embedding Watermark into Trigger Sets:* Embedding the watermark into the trigger set of the network is usually called black-box watermark. For instance, Adi et al. (2018) proposed to use abstract color images and random tags as trigger sets to embed watermark. Relying on zero-bit watermarking, a label-to-bit transformation model (Chen et al., 2019) improved the capacity of the watermark. Since that the trigger set watermark is easily forged (Adi et al., 2018), the robust watermark methods are further studied, such as (AprilPyone and Kiya, 2021; Zhu et al., 2020). Through a one-way hash function to generate image labels, Zhu et al. (2020) strengthens the connection between trigger set image and its label. AprilPyone and Kiya (2021) used a block-wise image transformation with a secret key to strengthen the connection between trigger set and training set, which can effectively avoid forgery attack. Besides, Zhao et al. (2021) proposed to generate a random graph with random node feature and labels as the watermark to protect graph neural network. In Szyller et al. (2021), the authors proposed to design a dynamic adversarial watermarking of neural networks (DAWN), which can resist against the surrogate attack by dynamically changing the small subset of queries as trigger set. In Lounici et al. (2021), by studying the various machine learning techniques, the authors extended the application of the model watermarking. Next, to solve the problem that the black-box watermarking behaves too sensitive when the frequent IP verification happens, Lounici et al. (2022) proposed a Blindspot model watermarking scheme. Recently, Yin et al. (2022) proposed a fragile watermarking method to mark the model by constructing the fragile trigger set from a generative model, which can detect malicious fine-tuning without degrading model performance.
- *Embedding Watermark into Outputs of Model:* Embedding the watermark into outputs of the model is independent of the model itself, while focusing on the output results from the protected DL model. For instance, in Wu et al. (2020), the authors proposed a novel model watermarking scheme, where the output results from the trained DNN model contain a certain watermark used for ownership verification. Besides, for protecting the DL model of image processing, Zhang et al. (2020a) proposed a spatial invisible watermarking scheme, which can to some degree resist against the attack from the trained surrogate models. In addition, the natural language watermarking mechanism was also proposed relying on the encoder-decoder network and adversarial training (Abdelnabi and Fritz, 2021).

To the best of our knowledge, the current studies on model watermarking mainly focus on the model of detection or classification; few model watermarking pays attention to GAN model. In fact, Wang and Kerschbaum (2021) made the first attempt to establish the Robust whIte-box GAn watermarking (RIGA) by adopt-

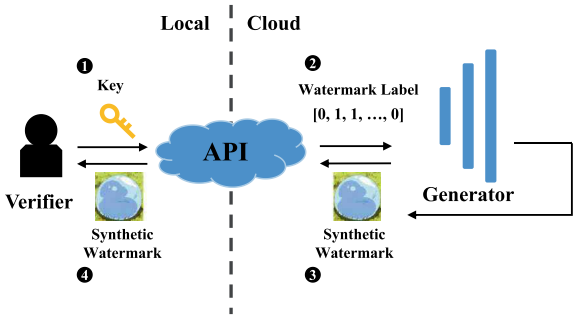

Fig. 1. Application of GAN model watermarking in the practical scenario.

ing the adversarial training, making the watermark hard to be detected. However, it only provides a model watermarking scheme inspired by the adversarial training while not specially protecting GAN model itself. Moreover, the white-box GAN watermarking requires model owners to access the structure and weight of the target model during verification, which to some extent limits its wide application. Additionally, Wu et al. (2020) proposed a watermarking technique towards generative model. Next, Fei et al. (2022) proposed to devise a GAN watermarking scheme by injecting an invisible watermark into the generated image. Both methods need to extract the watermark from the generated image during verification phase, which more or less limits the image quality as embedding capacity increases. To address the aforementioned issues, in this context, we propose to design a novel black-box GAN watermarking scheme in order to protect the IP of GAN model.

3. Proposed method

In order to protect the IP of the GAN model, in this section, we propose the GAN watermarking framework to protect the model. To establish such the framework, we first analyze and summarize the requirements of GAN model watermarking.

3.1. Problem definition

In the community of image generation, the GAN model plays a more and more important role. The well-trained GAN model can easily generate the photorealistic images which is nearly indistinguishable from real photos. Meanwhile, the GAN model is possibly stolen by the malicious attackers for illicit intention, which unavoidably imposes the new challenging issue for the current study. Thus, how to effectively protect the GAN model remains open. On the one hand, the study of the GAN model watermarking can borrow the idea from the current model watermarking scheme designed for image classification or detection; on the other hand, we need to investigate the particular characteristic structure of the GAN model, in order to establish the new paradigm for the design of GAN model watermarking. Straightforward, as Fig. 1 illustrates, we consider an application scenario of watermarking containing the local verifier and model service on the cloud, where the trained generator with model watermark can be triggered by the correct watermark label. It can be observed that the verifier can access the GAN model through the application programming interface (API) to achieve the ownership verification remotely. Specifically, during the verification, the verifier first sends the key generated by the key generation algorithm to the GAN model as watermark label through the API. Then the GAN with the watermark triggers the corresponding synthetic watermark according to the watermark label, in order to complete the verification of the model watermark. It is worth noting that if the incorrect key is input during verification, the correct watermark cannot be successfully triggered. Straightforward, the designed GAN model watermarking should strictly address the following requirements:

- Fidelity: the model performance (diversity and visual quality of the generated image) should be well preserved after watermark embedding.
- Integrity: the triggered watermark has a minimum false alarm rate.
- Robustness: the watermark cannot be easily removed when the model is attacked by parameter fine-tuning or model pruning.
- Security: the watermark cannot be easily detected by malicious attackers.
- Capacity: the watermark consisting of the amount of the payload can be effectively embedded into the protected model.
- Efficiency: the computation cost of watermark embedding and extraction cannot be very high.

3.2. GAN Watermarking

Fig. 2 shows the flow chart of the GAN watermarking framework. Here, the network model is deployed on a remote server, allowing users to access it remotely. Specifically, the whole watermarking framework can be divided into three phases: *Trigger Set Generation, Embedding Phase* and *Verification Phase*, described as follows:

- *Trigger Set Generation*: To verify the ownership of the GAN model, a private trigger set is required to establish, which is elaborated in Algorithm 1. Specifically, we adopt the function f_{wm_key} for generating a watermark label y_{wm} based on the owner signature b. It is worth noting that b denotes a bit stream, which can be generated by a semantic signature labeling the ownership. In the function $f_{wm\_key}(b, L)$, when the length of b is less than L, the function directly outputs b as watermark label y_{wm}. Otherwise, b is divided into b_1 and b_2, where b_1 denotes the first part and b_2 is the second part.

Algorithm 1: GAN watermark embedding.

Input : Training data $D_{train} = \{X_{train}, Y_{train}\}$, consisting of training images X_{train} and training labels Y_{train}, verification image I_{wm}, watermark label length L, owner signature b
Output: Trigger set $D_{wm} = \{X_{wm}, Y_{wm}\}$, watermarked generator G_θ, discriminator D_w

// **Trigger Set Generation**
$f_{wm\_key}(b, L)$ for generating a watermark label y_{wm} based on b
Initialize D_{wm} as $\emptyset$Function f_{wm_key}:
while $length(b) > L$ **do**
  $b_1 \leftarrow f_{substring}(0, L)$
  $b_2 \leftarrow f_{substring}(L, length(b))$
  $b \leftarrow b_1 \oplus b_2$
end
return b**for** i *in* range(I_{wm}) **do**
  $y_{wm} \leftarrow f_{wm\_key}(b, L)$
  $x_{wm} \leftarrow I_{wm}^{\{i\}}$
  $D_{wm} \leftarrow D_{wm} \cup \{x_{wm}, y_{wm}\}$
end
// **Embedding Phase**Initial discriminator parameter ω, initial generator parameter θ, latent vector z, discriminator loss L_D, generator loss L_G, watermark loss L_{wm}, weight parameter λ, batch size m, Adam hyperparameters α, β_1, β_2**for** *number of training epochs* **do**
  for *each batch* **do**
    sample original data $x \in D_{train} \cup D_{wm}$, synthetic data $\widetilde{X} = \{\widetilde{x}\}$
    $\widetilde{x} \leftarrow G_\theta(z)$
    $L_D \leftarrow D_\omega(\widetilde{x}) - D_\omega(x) + \lambda L_{wm}$
    $\omega \leftarrow Adam\left(\nabla_\omega \frac{1}{m} \sum_{i=1}^{m} L_D, \omega, \alpha, \beta_1, \beta_2\right)$
    $L_G \leftarrow -D_\omega(\widetilde{x})$
    $\theta \leftarrow Adam\left(\nabla_\theta \frac{1}{m} \sum_{i=1}^{m} L_G, \theta, \alpha, \beta_1, \beta_2\right)$
  end
end

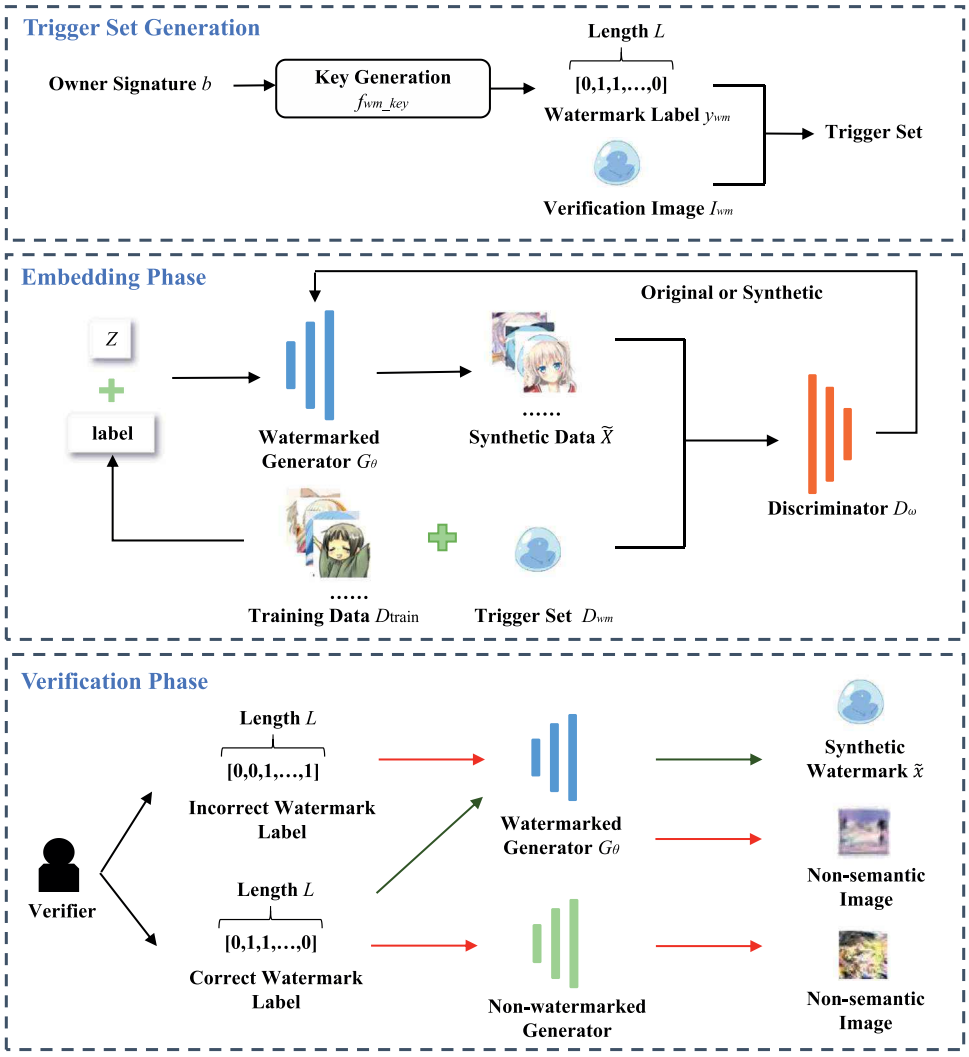

Fig. 2. Illustration of our proposed GAN model watermarking.

The function $f_{substring}$ is used for string truncation and extraction. Finally, let us assign the exclusive OR results of b_1 and b_2 to b. It should be noted that we need to repeat the aforementioned operation until the length of b is not larger than L. The watermark label length L depends on the classes of training data. Then both watermark label y_{wm} and verification image I_{wm} (selected by GAN model server) consist of trigger set.

- *Embedding Phase*: In our proposed GAN model watermarking scheme, we prepare a set of verification images in advance for the IP verification, and meanwhile set the watermark label as the key to trigger the synthetic watermark image from the watermarked generator network in the model. In particular, the latent vector z and label from training data and trigger set are combined as inputs for the generator to synthesize data. And the discriminator is trained to determine whether a sample is original or synthetic. Algorithm 1 shows the watermark embedding phase. In order to train the original data and trigger set at the same time, the loss function is designed to maintain model performance and ensure the quality of embedded watermark. Then the discriminator loss L_D is expressed as:

$$L_D = D_\omega(\tilde{x}) - D_\omega(x) + \lambda L_{wm}, \tag{1}$$

where L_{wm} denotes the additional cross entropy watermark loss and λ is a weight parameter with a default value of 1. When the synthetic watermark does not match the verification image, it will be subject to additional penalties. In fact, the procedure of watermarking the protected GAN model is efficiently completed during the model training.

- *Verification Phase*: The watermarked model is assigned to the authorized user, and the watermark label is conserved by the model owner. When a model IP dispute occurs, the model owner can trigger the synthetic watermark $\tilde{x}$ from the watermarked generator G_θ according to the watermark label y_{wm} for IP verification. Then the similarity between the output watermark $\tilde{x}$ and the verification image I_{wm} provided in advance is evaluated according to E (see Eq. (2)). Here, we combine structural similarity index measure (SSIM) (Wang et al., 2004), peak-signal-to-noise ratio (PSNR), and cosine similarity (COSIN) to evaluate the overall similarity, described as:

$$E = \alpha f_{ssim}(I_{wm}, \tilde{x}) + \beta f_{psnr}(I_{wm}, \tilde{x}) + \gamma f_{cosin}(I_{wm}, \tilde{x}), \tag{2}$$

where E denotes the evaluation value, α, β and γ denote the weight factors, and $\alpha + \beta + \gamma = 1$. If the evaluation value $E \geq \varepsilon$, the model owner can prove the ownership of the model. When a correct watermark label is input to the watermarked model (generator), a correct synthetic watermark can be successfully triggered. On the contrary, if the non-watermarked model is fed with the watermark label, the output result will be a non-semantic image (see Fig. 2). Besides, if the incorrect watermark label is fed into the watermarked model, a non-semantic image will also be output.

Table 1
GAN model performance comparison by adopting different datasets.

		WGAN-GP		ProGAN		StyleGAN2	
		cifar10	Danbooru2018	cifar10	Danbooru2018	cifar10	Danbooru2018
Original	FID	12.4626	-	11.9615	3.4598	11.0168	6.5331
	SWD	7.2899	-	6.9164	2.8467	10.2400	8.7453
Ours	FID	11.8928	-	8.1573	3.2856	7.8946	6.0972
	SWD	7.2252	-	6.6665	3.0911	9.0160	9.1217

4. Experimental results

In order to demonstrate the effectiveness of our method, a series of numerical experiments are conducted on the baseline GAN models, including WGAN-GP (Gulrajani et al., 2017), ProGAN (Karras et al., 2017) and StyleGAN2 (Karras et al., 2020). And we conduct experiments on two image datasets: cifar10 (Krizhevsky et al., 2009) and Danbooru2018 (Wang, 2019). The former consists of 50,000 colour images classified into 10 classes, with 5000 images per class. The latter consists of more than 200,000 cartoon character images classified into 182 classes, with different images per class. During the watermark embedding phase, we strictly follow the architecture in Gulrajani et al. (2017), Karras et al. (2017, 2020) to train our model for generating GAN synthetic images.

4.1. Overall performance

In this subsection, we conduct experiments to evaluate the performance of the GAN models with watermark. We assume that after embedding watermark, the performance of the original GAN model cannot be degraded. Here we consider fréchet inception distance (FID) (Heusel et al., 2017) and sliced wasserstein distance (SWD) (Karras et al., 2017) as metrics of evaluating the performance of the GAN models. The lower the value of FID and SWD, the better the performance of GAN model.

Table 1 shows the performance comparison between the original and the protected model embedded by our proposed watermark via FID and SWD. By observation, regardless of WGAN-GP, ProGAN or StyleGAN2 on cifar10 and Danbooru2018, compared with the original version, our proposed GAN model with watermark can obtain lower FID and SWD value on the whole. That directly verifies the fidelity of our proposed method, meaning that the performance of the GAN model with watermark can be retained. Besides, for WGAN-GP, it should be noted that Danbooru2018 has a problem of overfitting, since that Danbooru2018 has too many labels in the dataset while the WGAN-GP model cannot fit the entire sample space, leading to that the loss is hardly to converge. In such scenario, we cannot acquire the results. Similarly, in Table 2, the specific results also cannot be given.

4.2. Watermark quality evaluation

In practice, when a model IP dispute occurs, the model owner needs to trigger the synthetic watermark according to the watermark label for verifying its ownership. Therefore, the synthetic watermark by the generator from GAN model needs to meet two requirements. The synthetic watermark can be extracted correctly as the key is correct. Otherwise the protected model cannot generate the corresponding synthetic watermark. Meanwhile, the synthetic watermark should be reliable and can be successfully detected.

Here, we consider three quantitative indicators (i.e., PSNR, SSIM and COSIN) to evaluate the watermark quality. The combination of these three indicators can effectively and intuitively evaluate the watermark quality. In Table 2, we compare the verification image from trigger set with the synthetic watermark, in order to calculate the values of PSNR, SSIM and COSIN. By observation, the average SSIM is above 0.85; the average COSIN is above 0.99; the average PSNR is above 26. It means that the deviation between the verification image and the synthetic watermark is small and the similarity is very high. The experimental results demonstrate that the synthetic watermark is of high quality. Moreover, according to the value E from Eq. (2), we calculate the watermark detection rate. It can be observed that regardless of WGAN-GP, ProGAN or StyleGAN2 on cifar10 and Danbooru2018, the watermark detection rate is always equal to 100%, meaning that the GAN model can return reliable watermark for IP protection.

4.3. Robustness comparison

4.3.1. Parameter fine-tuning

In general, parameter fine-tuning uses less computing resources and time to retune the model parameters, in order to confirm a new local minimum while maintaining model performance. In the model watermarking attack, parameter fine-tuning can be used to remove watermark data. Thus in such scenario, we assume that the attacker can obtain the original training data and settings of the training model to fine-tune the model parameters. Fig. 3 presents the performance for the original model without watermark, the protected model with watermark, and the model attacked by fine-tuning. By observation, after fine-tuning, the detection rate of watermark still remains over 80% for WGAN-GP and ProGAN, 72% for StyleGAN2. Moreover, for FID, the model after fine-tuning obtains the more high value, meaning that the protected GAN model cannot be effectively adopted by the attacker.

4.3.2. Model pruning

Model pruning can reduce redundant parameters while ensuring the performance of the original model. Similar to image com-

Table 2
Watermark quality evaluation, where the average PSNR, SSIM and COSIN values are obtained between the verification image and the synthetic watermark. (see Fig. 2).

Model	Dataset	PSNR	SSIM	COSIN	Detection Rate
WGAN-GP	cifar10	24.0088	0.8571	0.9966	100%
	Danbooru2018	-	-	-	-
ProGAN	cifar10	26.8268	0.8548	0.9978	100%
	Danbooru2018	21.1382	0.7006	0.9948	100%
StyleGAN2	cifar10	29.3933	0.9062	0.9989	100%
	Danbooru2018	29.3638	0.9068	0.9981	100%

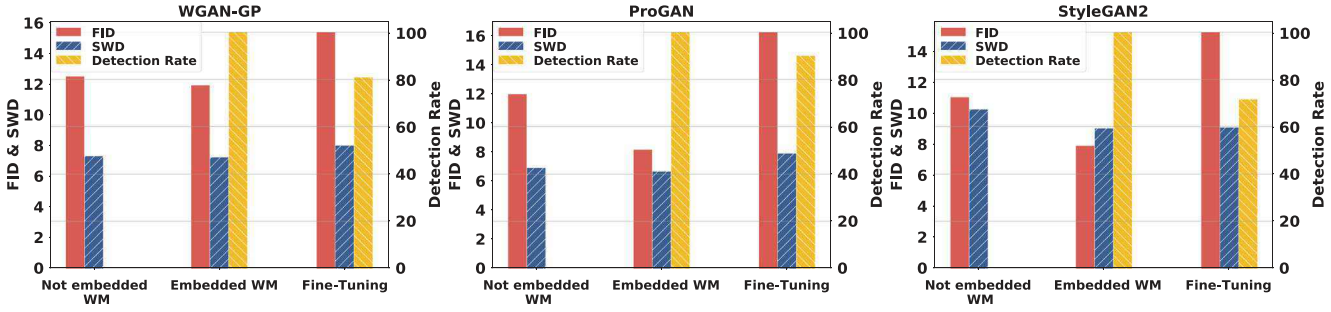

Fig. 3. GAN model robust evaluation and watermark performance on cifar10 after fine-tuning attack, in which "Not embedded WM" represents the original model without watermark, "Embedded WM" for the protected model with watermark, and "Fine-Tuning" for the model attacked by fine-tuning.

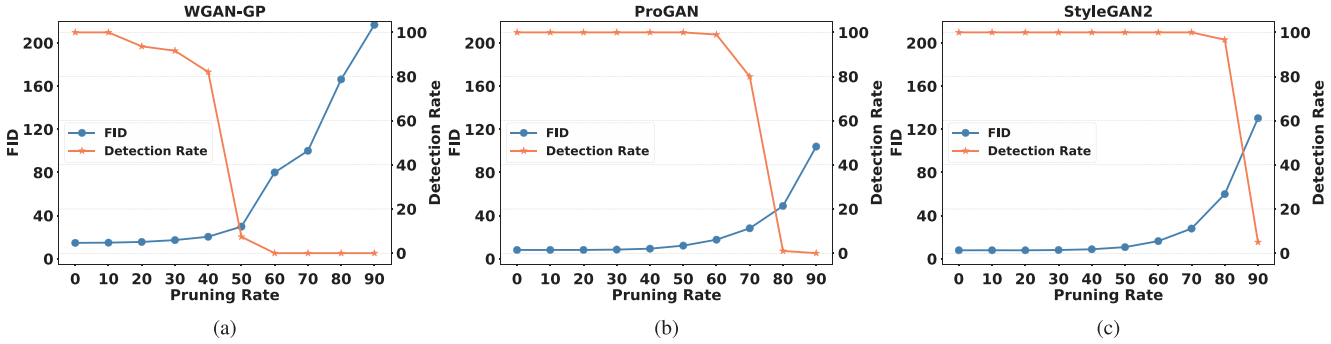

Fig. 4. GAN model robust evaluation and watermark performance on cifar10 after pruning attack.

pression, model pruning might also affect the watermark performance. Thus in such scenario, we use the method proposed in Han et al. (2015) to prune the parameters in the model by using different pruning rates for testing our model robustness. As Fig. 4 illustrates, the watermark can still be effectively detected under the pruning attack with the pruning rate even up to 40%. Besides, for ProGAN and StyleGAN2 models, the watermark is still effective when the pruning rate as high as 60% and 80% respectively. In fact, it should be noted that when the pruning rate arrives at the upper bound, the performance of the original GAN model is greatly reduced, leading to its invalidation. In such scenario, it is of insignificance to study the performance of watermark robustness.

4.4. Comparison with SOTAs

In this subsection, let us carry out the comparison experiments, in which the methods (Lounici et al., 2021; 2022; Szyller et al., 2021) are respectively compared. To our knowledge, the tasks of the compared watermarking schemes are totally different, leading to that the comprehensive comparison of all the methods is very difficult to carry out. Nevertheless, we still proposed to compare our proposed GAN model watermarking with the aforementioned SOTAs in some aspects.

As Fig. 5 illustrates, we first compare the watermark integrity, referring to as evaluating the quality of the model watermark by detection rate. In Szyller et al. (2021), "9L" denotes 9-layer DNN model and "RN34" denotes the classic ResNet34 (He et al., 2016), in which the compared models are also trained by cifar10 dataset as in our prior experiments. Basically, the watermark detection rate is satisfying. In Lounici et al. (2021), we select the representative internal watermark method on image classification and reinforcement learning. The watermark detection rate over 98% is nearly perfect, implying its good performance of watermark integrity. In Lounici et al. (2022), the fairness-based watermarking scheme is selected to compare, where the perfect 100% detection rate can be obtained. Similarly, "RN18" denotes the classic ResNet18 (He et al., 2016). Moreover, our proposed GAN watermarking scheme can also obtain the perfect detection rate. It should be noted that although the model watermarking schemes of Szyller et al. (2021) and Lounici et al. (2022) are applied into the classification models, the same training dataset, referring to as cifar10, further guarantees the fairness of performance comparison in the similar experimental settings. Besides, for (Lounici et al., 2021), the only experimental results of image classification and reinforcement learning are illustrated. By comparison, we can clearly observe that our proposed watermarking scheme is as good as the SOTAs, which can be smoothly used for GAN model.

Next, in Fig. 6, we compare our proposed method with (Szyller et al., 2021) on the robustness performance when the model pruning attack happens. It is still proposed to adopt the cifar10 dataset for model training. By observation, as the pruning rate increases, the model watermark proposed in this context basically always can be effectively detected, implying that our proposed GAN model watermarking performs better robust, which is better than its counterpart.

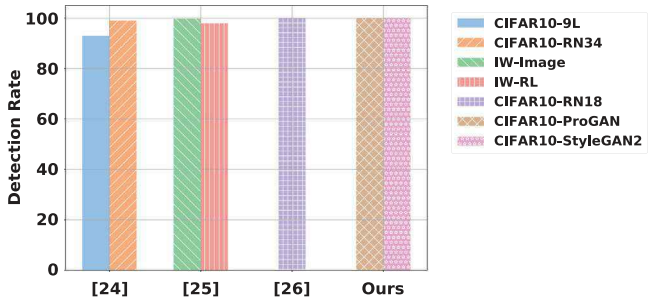

Fig. 5. Integrity comparison of different model watermarking schemes.

Table 3
Requirement check of our proposed GAN model watermarking scheme.

Requirements	Metrics	Check
Fidelity	FID & SWD	✓ The GAN watermark embedding nearly cannot affect the performance of the GAN model itself. In the subsection 4–4.1 (see Table 1 for details), the low value of FID and SWD verifies the fidelity of the GAN model.
Integrity	PSNR, SSIM and COSIN	✓ The GAN Watermark can be accurately and effectively extracted from the model. In the subsection 4–4.2, the high quality of the extracted watermark can be effectively guaranteed, which is evaluated by three indicators (see Table 2 for details).
Robustness	Detection Rate	✓ When encountering malicious attacks, the GAN model watermark can still be exactly extracted. In the subsection 4–4.3, as we set the mainstream attacks including parameter fine-tuning and model pruning to the GAN model, our proposed GAN watermark is still valid (see Figs. 3 and 4 for details).
Security	–	✓ The GAN watermark can only be triggered by a completely correct key; otherwise the GAN model can only trigger a non-semantic image (see Fig. 2 for details). Meanwhile, it should be noted that only the model owner owns the verification image, which should be securely preserved.
Capacity	Bits	✓ The model watermarking scheme has the satisfying capacity, where the synthetic image serves as watermark depends on the GAN model itself.
Efficiency	Time	✓ The training time between the GAN model with carrying the proposed watermark and the GAN model without the watermark is very similar.

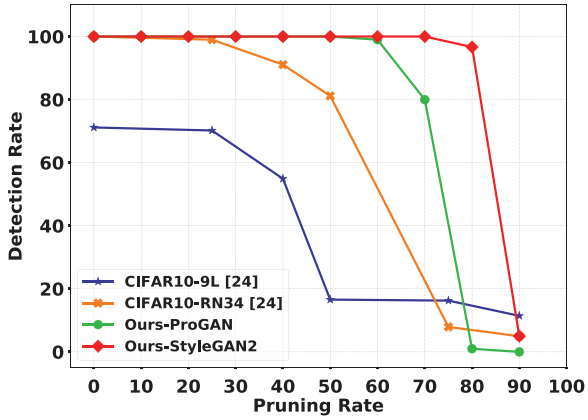

Fig. 6. Robustness comparison of different model watermarking schemes.

5. Discussion

In this context, we proposed to design the GAN model watermarking scheme, whose effectiveness has been comprehensively verified in the experiments. For clarity, we further list the requirements of the model watermarking to check if the proposed method reaches that in Table 3. Obviously, it can be observed that the proposed GAN model watermarking scheme is good enough to be applied to protect the IP of the GAN model.

Furthermore, in the compared experiments, we compare the proposed GAN model watermarking scheme with the SOTAs (Lounici et al., 2021; 2022; Szyller et al., 2021) in some respects. Meanwhile, the experimental results directly verify the superiority of the proposed method. In fact, due to the different types of the input and output data, most of the current model watermarking schemes such as (Lounici et al., 2021; 2022; Szyller et al., 2021), cannot be directly applied to the GAN model. To address that challenging issue, the GAN model watermarking method is proposed in this context. Only dependent on the correct watermark label, the synthetic watermark can be successfully triggered by the model owner for IP protection. More importantly, the designed GAN model watermarking fully meets the requirements of the model watermarking.

In addition, it should be noted that embedding watermark into parameters of model needs to be verified by accessing the model parameters and structure, which is not conducive to ownership verification. Moreover, embedding watermark into outputs of the protected model needs to extract watermark from the generated image, which avoidably affects the quality of the output data to a certain extent. In such case, the users possibly would not like to obtain the output with the specific watermark, which limits the wide application of the model watermarking. Thus, it is proposed to design the black-box watermarking scheme towards the GAN model. In particular, the verifier only needs to access the API provided by the GAN model as normal users, and inputs the correct watermark label to trigger the model watermark. That is, the GAN model does not need to be obtained in advance before verification, and the fidelity of the GAN model can be guaranteed.

6. Conclusion

In this paper, we propose a novel framework of model watermarking to solve the issue of the IP protection towards GAN model. Unlike the most existing methods of model watermarking that are well devised to protect the specific network for detection or classification task, our proposed framework are committed to protecting GAN model. Specifically, the synthetic watermark is generated in the generator through model training and trigger set. During the verification of model ownership, the model owner can trigger the watermark in the generator, only if the correct watermark label is acquired. The extensive experiments validate the effectiveness of our proposed model watermarking, which is robust enough to resist against parameter fine-tuning and model pruning attacks. Moreover, we need to address that our proposed GAN watermarking scheme is not only adopted in the aforementioned GAN models but also the other baselines. Besides, in our proposed GAN watermarking framework, we can also use the other benchmark image datasets, not limited to cifar10 or Danbooru2018.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Tong Qiao: Conceptualization, Methodology, Software. **Yuyan Ma:** Data curation, Writing – original draft. **Ning Zheng:** Supervision. **Hanzhou Wu:** Visualization, Investigation. **Yanli Chen:** Software, Validation. **Ming Xu:** Writing – review & editing. **Xiangyang Luo:** Supervision.

Data availability

Data will be made available on request.

Acknowledgments

This work was supported by Zhejiang Provincial Natural Science Foundation of China (No. LZ23F020006), the Fundamental Research Funds for the Provincial Universities of Zhejiang (Grant No. GK219909299001-007), the Open Projects Program of National Laboratory of Pattern Recognition, Open Foundation of Henan Key Laboratory of Cyberspace Situation Awareness (HNTS2022016), the National Key R&D Program of China (No. 2022YFB3102900), the National Natural Science Foundation of China (Grant No. U1804263, 62172435) and the Zhongyuan Science and Technology Innovation Leading Talent Project of China (Grant No. 214200510019).

References

Abdelnabi, S., Fritz, M., 2021. Adversarial watermarking transformer: towards tracing text provenance with data hiding. In: 2021 IEEE Symposium on Security and Privacy (SP). IEEE, pp. 121–140.

Adi, Y., Baum, C., Cisse, M., Pinkas, B., Keshet, J., 2018. Turning your weakness into a strength: Watermarking deep neural networks by backdooring. In: 27th {USENIX} Security Symposium ({USENIX} Security 18), pp. 1615–1631.

AprilPyone, M., Kiya, H., 2021. Piracy-resistant DNN watermarking by block-wise image transformation with secret key. arXiv preprint arXiv:2104.04241.

Chen, H., Rouhani, B. D., Koushanfar, F., 2019. Blackmarks: blackbox multibit watermarking for deep neural networks. arXiv preprint arXiv:1904.00344.

Fan, L., Ng, K.W., Chan, C.S., 2019. Rethinking deep neural network ownership verification: embedding passports to defeat ambiguity attacks. Adv. Neural Inf. Process. Syst. 32.

Fei, J., Xia, Z., Tondi, B., Barni, M., 2022. Supervised GAN watermarking for intellectual property protection. arXiv preprint arXiv:2209.03466.

Feng, L., Zhang, X., 2020. Watermarking neural network with compensation mechanism. In: International Conference on Knowledge Science, Engineering and Management. Springer, pp. 363–375.

Gou, Y., Wu, Q., Li, M., Gong, B., Han, M., 2020. SegAttnGAN: text to image generation with segmentation attention. arXiv preprint arXiv:2005.12444.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A., 2017. Improved training of Wasserstein GANs. arXiv preprint arXiv:1704.00028.

Han, S., Pool, J., Tran, J., Dally, W. J., 2015. Learning both weights and connections for efficient neural networks. arXiv preprint arXiv:1506.02626.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S., 2017. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. Adv. Neural Inf. Process. Syst. 30.

Juuti, M., Szyller, S., Marchal, S., Asokan, N., 2019. PRADA: protecting against DNN model stealing attacks. In: 2019 IEEE European Symposium on Security and Privacy (EuroS&P). IEEE, pp. 512–527.

Karras, T., Aila, T., Laine, S., Lehtinen, J., 2017. Progressive growing of GANs for improved quality, stability, and variation. arXiv preprint arXiv:1710.10196.

Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., Aila, T., 2020. Analyzing and improving the image quality of StyleGAN. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8110–8119.

Krizhevsky, A., Hinton, G., et al., 2009. Learning multiple layers of features from tiny images.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. ImageNet classification with deep convolutional neural networks. Adv. Neural Inf. Process. Syst. 25, 1097–1105.

LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D., 1989. Backpropagation applied to handwritten zip code recognition. Neural Comput. 1 (4), 541–551.

Lounici, S., Njeh, M., Ermis, O., Önen, M., Trabelsi, S., 2021. Yes we can: watermarking machine learning models beyond classification. In: 2021 IEEE 34th Computer Security Foundations Symposium (CSF). IEEE, pp. 1–14.

Lounici, S., Önen, M., Ermis, O., Trabelsi, S., 2022. BlindSpot: watermarking through fairness. In: Proceedings of the 2022 ACM Workshop on Information Hiding and Multimedia Security, pp. 39–50.

Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z.B., Swami, A., 2017. Practical black-box attacks against machine learning. In: Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security, pp. 506–519.

Richardson, E., Alaluf, Y., Patashnik, O., Nitzan, Y., Azar, Y., Shapiro, S., Cohen-Or, D., 2021. Encoding in style: a StyleGAN encoder for image-to-image translation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2287–2296.

Romero, A., Castillo, A., Abril-Nova, J., Timofte, R., Das, R., Hira, S., Pan, Z., Zhang, M., Li, B., He, D., et al., 2022. NTIRE 2022 image inpainting challenge: Report. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1150–1182.

Szyller, S., Atli, B.G., Marchal, S., Asokan, N., 2021. DAWN: dynamic adversarial watermarking of neural networks. In: Proceedings of the 29th ACM International Conference on Multimedia, pp. 4417–4425.

Tramèr, F., Zhang, F., Juels, A., Reiter, M.K., Ristenpart, T., 2016. Stealing machine learning models via prediction APIs. In: 25th {USENIX} Security Symposium ({USENIX} Security 16), pp. 601–618.

Uchida, Y., Nagai, Y., Sakazawa, S., Satoh, S., 2017. Embedding watermarks into deep neural networks. In: Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval, pp. 269–277.

Wang, T., Kerschbaum, F., 2019. Robust and undetectable white-box watermarks for deep neural networks. arXiv preprint arXiv:1910.14268.

Wang, T., Kerschbaum, F., 2021. RIGA: covert and robust white-box watermarking of deep neural networks. In: Proceedings of the Web Conference 2021, pp. 993–1004.

Wang, Y., 2019. Danbooru 2018 anime character recognition dataset. https://github.com/grapeot/Danbooru2018AnimeCharacterRecognitionDataset.

Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P., 2004. Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. 13 (4), 600–612.

Wu, H., Liu, G., Yao, Y., Zhang, X., 2020. Watermarking neural networks with watermarked images. IEEE Trans. Circuits Syst. Video Technol. 31 (7), 2591–2601.

Yin, Z., Yin, H., Zhang, X., 2022. Neural network fragile watermarking with no model performance degradation. In: 2022 IEEE International Conference on Image Processing (ICIP). IEEE, pp. 3958–3962.

Yu, H., Yang, K., Zhang, T., Tsai, Y.-Y., Ho, T.-Y., Jin, Y., 2020. CloudLeak: large-scale deep learning models stealing through adversarial examples. NDSS.

Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., Huang, T.S., 2018. Generative image inpainting with contextual attention. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5505–5514.

Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., Metaxas, D.N., 2017. StackGAN: text to photo-realistic image synthesis with stacked generative adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 5907–5915.

Zhang, J., Chen, D., Liao, J., Fang, H., Zhang, W., Zhou, W., Cui, H., Yu, N., 2020. Model watermarking for image processing networks. In: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, pp. 12805–12812.

Zhang, J., Chen, D., Liao, J., Zhang, W., Hua, G., Yu, N., 2020. Passport-aware normalization for deep model protection. Adv. Neural Inf. Process. Syst. 33, 22619–22628.

Zhao, X., Wu, H., Zhang, X., 2021. Watermarking graph neural networks by random graphs. In: 2021 9th International Symposium on Digital Forensics and Security (ISDFS). IEEE, pp. 1–6.

Zhu, J.-Y., Park, T., Isola, P., Efros, A.A., 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2223–2232.

Zhu, R., Zhang, X., Shi, M., Tang, Z., 2020. Secure neural network watermarking protocol against forging attack. EURASIP J. Image Video Process. 2020 (1), 1–12.

Tong Qiao received the BS degree in Electronic and Information Engineering in 2009 from Information Engineering University, Zhengzhou, China, and the MS degree in Communication and Information System in 2012 from Shanghai University, Shanghai, China, and the PhD degree in University of Technology of Troyes, Laboratory of Systems Modelling and Dependability, Troyes, France, in 2016. He currently works as an Associate Professor in School of Cyberspace from Hangzhou Dianzi University. His current research interests focus on media forensics, AI security and data hiding. He has published over 60 peer-reviewed papers on journals and conferences.

Yuyan Ma received the BS degree in computer science and technology from Zhejiang University City College, Hangzhou, China, in 2020. She is currently pursuing the MS degree with Hangzhou Dianzi University. Her current research interests focus on AI security.

Ning Zheng received his MS degree from Zhejiang University in 1987. He is currently a Professor of Hangzhou Dianzi University. His research interest is digital forensics.

Hanzhou Wu received the BS and PhD degrees from Southwest Jiaotong University, Chengdu, China, in June 2011 and June 2017, respectively. From October 2014 to October 2016, he was a Visiting Scholar with the New Jersey Institute of Technology, NJ, USA. He was a Research Staff with the Institute of Automation, Chinese Academy of Sciences, Beijing, China, from July 2017 to March 2019. He is currently an Associate Professor with Shanghai University, Shanghai, China. He has authored three book chapters and published around 20 papers in peer journals and conferences, such as the IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, IEEE WIFS, and ACM IH&MMSec. His research interests include information hiding, graph theory, and deep learning.

Yanli Chen received the BS degree in Electronic Information Science and Technology in 2015 from Qingdao University, Qingdao, China, and the MS degree in Signal and Information Processing in 2018 from Xidian University, Xi'an, China, and the PhD degree in the Laboratory of Computer Science and Digital Society, University of Technology of Troyes, Troyes, France, in 2016. She currently works as an Assistant

Professor in School of Cyberspace from Hangzhou Dianzi University. Her research interests focus on digital image forensics..

Ming Xu received his PhD degrees from Zhejiang University in 2004. He is currently a Professor of Hangzhou Dianzi University. His research interest is digital forensics.

Xiangyang Luo received his BS, MS, and PhD degrees from the State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou, China, in 2001, 2004, and 2010, respectively. He is the author or co-author of more than 100 refereed international journal and conference papers. He is currently a Professor of the State Key Laboratory of Mathematical Engineering and Advanced Computing. His research interests are image steganalysis and Forensics.