Signal, Image and Video Processing (2025) 19:1261
https://doi.org/10.1007/s11760-025-04832-y

ORIGINAL PAPER

MDD-watermark: multi-domain decoupled watermarking for deepfake detection and source tracing

Mengya Zhang[1,2] · Xiaohan Wang[1] · Qinghui Zhang[1,2] · Huaming Wu[3] · Wenjuan Li[4]

Received: 4 August 2025 / Revised: 6 September 2025 / Accepted: 22 September 2025

Abstract

In recent years, malicious exploitation of Deepfakes technology has occurred frequently, posing a serious threat to society. Although many post-facto detection methods have been developed for Deepfakes, these passive forensic techniques do not take any preventive measures on the original face images before tampering occurs. To bridge this gap and improve the forensic ecosystem, we propose a forward-looking solution called Multi-Domain Decoupled Watermarking (MDD-Watermark), which aims to provide a unified framework for source tracking and Deepfake detection. MDD-Watermark is constructed by multi-domain decoupling of the original image; when the image is forged, the image reconstructed based on the decoupling information of the original image will be significantly different from the forged image in terms of features. This difference can be quantitatively analyzed using traditional image evaluation metrics (e.g., PSNR, SSIM). We also design a deep learning-based framework, XUNet. It can efficiently embed the MDD-Watermark into the carrier image and still stably extract the watermark information in the face of multiple perturbations (e.g., noise, compression, rotation, etc.). Experimental results demonstrate that while maintaining high visual quality, the proposed method not only effectively resists deepfake attacks and preserves watermark robustness, but also enables significant stratification in image quality metrics such as PSNR when comparing watermarked images with forged images against their respective reconstructed counterparts.

Keywords Watermark robustness · Deep face forgery · Active defense · Passive detection

1 Introduction

With the breakthroughs in deep generation technologies such as Generative Adversarial Networks (GANs) [1] and Diffusion Models [2], deep forgery technology has realized a significant simplification of the operation process and a leapfrog enhancement of the generation effect. Digital content creation centered on deep face-swapping is accelerating the penetration of the film and television entertainment field [3]. However, social risks hidden behind technological empowerment should not be ignored [4]. The serious disconnect between the current regulatory system and the speed of technological development has made it possible for unscrupulous elements to take advantage of the seamless connection between open-source algorithms and cloud computing power so that malicious forgers can easily access high-performance forgery models and then batch-generate hyper-realistic digital doppelgangers of political figures and public figures. This technological abuse has given rise to new forms of crime, such as political manipulation, commercial

✉ Qinghui Zhang
zqh131@haut.edu.cn

Mengya Zhang
monicazhang@haut.edu.cn

Xiaohan Wang
2023920217@stu.haut.edu.cn

Huaming Wu
whming@tju.edu.cn

Wenjuan Li
wenjuan.li@ia.ac.cn

[1] Key Laboratory of Grain Information Processing and Control, Henan University of Technology, Ministry of Education, Zhengzhou 450001, China

[2] Henan Key Laboratory of Grain Photoelectric Detection and Control, Henan University of Technology, Zhengzhou 450001, China

[3] Center for Applied Mathematics, Tianjin University, Tianjin 300072, China

[4] National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China

Published online: 06 October 2025

fraud, and identity theft, posing a serious challenge to the trust system in the digital age.

The current defense system against deep face forgery mainly covers active defense and passive detection. Among them, the mainstream method of passive detection technology is to train a forgery detector, which can determine whether an image has been modified [5]. Researchers have proposed many deep forgery detection methods from the perspectives of deep feature discriminant analysis [6], deep forgery technique defect analysis [7], and cross-data task model generalization [8, 9]. However, the effectiveness of their methods is overly dependent on high-quality datasets with high resolution and clarity or on whether the detected samples have a certain step of technical defects.

Watermark traceability is a widely studied active defense scheme, through the special invisible logo information deeply embedded in the carrier image, under the premise of maintaining the indistinguishability of the carrier image to achieve reliable storage and accurate extraction of hidden information [10, 11]. According to the form of expression of the information carrier, the current watermarking system is mainly divided into the following: meaningful watermarking will be digital images or audio clips encoded to generate watermark signals, while meaningless watermarking uses structured serial numbers as the information carrier. It is worth noting that, at this stage, the mainstream research is based on randomized meaningless watermarking to build a training model [12, 13], and its technical path requires end users to convert business information into sequence numbers for embedding through preset coding rules. Although this technical architecture can realize infringement traceability and evidence fixation in increasingly complex digital forgery scenarios, its single-dimensional robust design has shown significant limitations. Relying only on the watermark traceability mechanism makes it difficult to curb the occurrence of the source of counterfeiting behavior effectively, and there is an urgent need to break through the technical framework of a single robust watermark and explore a new type of watermark authentication system combining the active traceability mechanism with the passive detection capability.

As shown in Fig. 1, our Multi-Domain Decoupled Watermarking (MDD-Watermark) realizes a double breakthrough: on the one hand, it overcomes the inherent limitations of traditional passive detection technology in terms of after-the-fact authentication; on the other hand, it effectively addresses the deficiencies of the existing source tracking watermarks in terms of detecting forged images. To ensure MDD-Watermark remains robust against deepfake attacks, we developed a deep learning-based XUNet model. Deep learning-based watermarking technology offers clear technical superiority over traditional methods and stands out as an effective alternative [14]. MDD-Watermark shows unique advantages when dealing with deep forgery attacks such as malicious face exchanges: MDD-Watermark is extracted from the original image through compression and can be regarded as the image's "ID." This watermark information occupies only a portion of the watermark capacity, leaving the remaining space available for users to embed custom identity information. The combined effect of these two information components ensures the watermark's traceability. When the original image A is subjected to deepfake attacks to generate a forged image B, differences between B and A are inevitable. If a reconstructed image B_C is created using partial information from A, significant inconsistencies will emerge between B_C and B. Conversely, reconstructing the original image A itself using the same information yields A_C, which exhibits minimal differences from A. By comparing the discrepancy levels between A and A_C, and between B and B_C, a pronounced distinction becomes apparent, revealing whether the image has been tampered with.

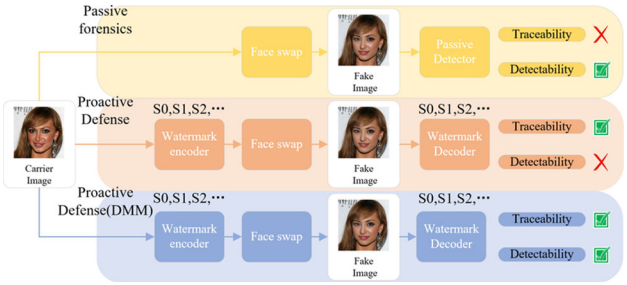

Fig. 1 Illustration of different types of Deepfake countermeasures: passive forensics focuses on the Deepfake detection task while the original face is unprotected; active watermarking for traceability focuses on post-Deepfake authentication; our active forensics focuses on traceability tracking and Deepfake detection

2 Related work

2.1 Deepfake passive forensics

UCF [15] proposes a novel decoupling framework that decomposes image information into three mutually independent components: forgery-insensitive features, method-specific forgery features, and universal forgery features. Extensive experimental evaluations demonstrate that this framework outperforms multiple state-of-the-art methods in cross-domain generalization capabilities. DeepfakeBench [16] includes 15 state-of-the-art detection methods, 9 deepfake datasets, a suite of deepfake detection evaluation protocols and analysis tools, and comprehensive evaluation results. Furthermore, based on a multi-faceted in-depth analysis of these evaluations, DeepfakeBench provides new insights. CNN-LSTM [3] proposes a facial geometry prior module (FGPM) to extract the facial geometry feature maps, which are embedded into the upsampled feature maps generated by

the CNN-LSTM network. Finally, a decoder is used to learn the mapping from low-resolution feature maps to pixels to predict the manipulation localization. Alternatively, a softmax classifier is used to predict true and false face images. Through experiments on several popular datasets, the proposed detection model demonstrates the ability to localize the manipulation at the pixel level, as well as a high prediction ability for real or fake face images.

2.2 Watermarking method

Deep learning-based image digital watermarking techniques show significant advantages in terms of attack robustness. Deep neural networks can better adapt to unknown image distortions and attack scenarios by virtue of their strong nonlinear fitting ability and black-box characteristics, thus significantly improving the robustness and concealment of watermarking. Balujia et al. [17] propose a deep neural network-based watermarking method, which is the first time that deep neural networks are applied to the field of image watermarking, and successfully achieves the goal of hiding one color image into another image of the same size. However, the method mainly focuses on the embedding and extraction ability of the watermark, with less consideration of robustness, and thus has limited performance in the face of attacks. HiDDeN [18] is an end-to-end deep learning watermarking framework consisting of an encoder, a discriminator, a noise layer, and a decoder. The innovation of the approach is the introduction of a noise layer to model various attacks (e.g., blurring, Gaussian noise, and cropping), which enhances the robustness of the model during the training process. Experiments show that HiDDeN exhibits better robustness against a wide range of common attacks. StegaStamp [19] also employs a deep neural network to build the encoder and decoder, but it focuses on improving robustness against photo-taking attacks. The method simulates real-world photographic distortions (e.g., lighting variations, lens distortions, etc.) to enable watermarks to be reliably extracted even after they have been photographed and printed, making it suitable for watermarking applications in physical scenarios. Sepmark [20] adopts an architecture with one encoder and two decoders for extracting robust and vulnerable watermarks, respectively. The robust watermark is used for forgery forensics, while the vulnerability watermark is used to detect forgeries. This dual watermarking strategy enables Sepmark to fulfill robustness and vulnerability requirements for complex application scenarios.

3 Method

3.1 MDD-watermark

By analyzing a large amount of experimental data, we found an important phenomenon: although images embedded with invisible watermarks are almost indistinguishable from the original images under human visual perception, significant differences can still be detected in some quantitative image evaluation metrics (e.g., PSNR [21]). This difference reflects the subtle alteration of the image's pixel-level information by the watermark embedding process. When these images with invisible watermarks are used for face-swapping counterfeiting, the difference is further amplified between the counterfeited image and the original image. This amplification effect may arise from the superimposed effect of the initial differences introduced by the watermark embedding and the secondary distortion caused by the face-swapping operation.

Algorithm 1 Watermark Construction

Input: $X_1, X_2, ...X_n$ (Carrier images, watermarked images or forged images);
Output: MDD-Watermark;
1: **for** $i \in [1, n]$ **do**
2: $X_{gi} \Leftarrow X_i$ convert to greyscale;
3: $X_{hi} \Leftarrow \mathbf{DCT}(X_{gi})[0 : H/2, 0 : W/2]$;
4: $X_{hfi} \Leftarrow X_{hi}$ normalisation, magnification 255x;
5: $LL_{2i} \Leftarrow \mathbf{2D\text{-}DCT}(X_{hfi})$;
6: $LL_{2i} \Leftarrow LL_{2i}$ normalisation;
7: $\sum_i \Leftarrow \mathbf{SVD}(LL_{2i})$;
8: **end for**

Based on this finding, we designed an innovative MDD-Watermarking framework, as shown in Fig. 2, and the code is shown in Algorithm 1. From Fig. 2, it can be seen that both watermark construction and image reconstruction are based on Algorithm 1, but their data processing strategies are different: watermark generation encodes by extracting the larger singular values in the singular value matrix, while image reconstruction uses the remaining information to complete the reconstruction. When the watermark and reconstructed image originate from the same original image, the visual difference between the reconstructed result and the original image is relatively small; If the source images are different, the difference in singular value distribution will lead to significant deviations in the reconstructed images.

3.2 Model architecture

Our XUNet model adopts an encoder-decoder architecture and combines with a Noise Pool (NP) to simulate various interferences and common forgery models in the real world, enhancing the robustness and covertness of watermarking. As

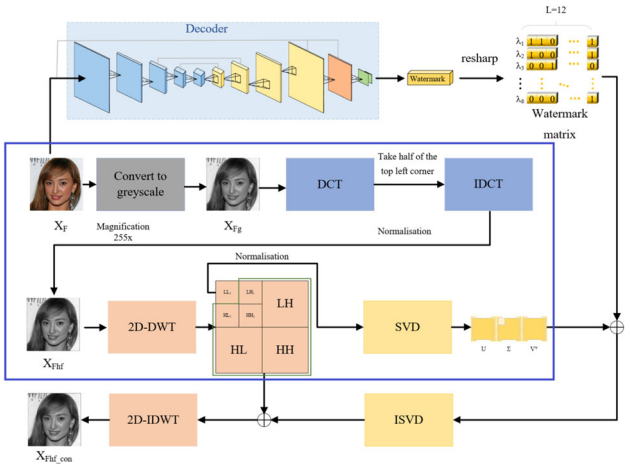

Fig. 2 Illustration of Watermark Construction and reconstructing an image using MDD-Watermark under a forged image. The blue area represents the process of building MDD-Watermark, which does not require U and V, where only half of Σ is selected. U and V are unitary matrices after singular value decomposition, and Σ is the singular value

shown in Fig. 3, the core components of the model include an Encoder (En), a Decoder (De), and a Noise Pool (NP) with the following workflow.

3.2.1 Encoder

Encoder is based on U-Net [22] and Xception [23] architectures. The carrier image X_O is first passed through the Entry Flow module of the Xception architecture for initial feature extraction and downsampling. This module gradually reduces the spatial dimensions of the image through a series of convolutional and deconvolutional operations while extracting high-level semantic features. These feature mappings will be used for subsequent watermark embedding and feature fusion. Meanwhile, the watermark information M is extended to $L \times L$ by the original information length L through multiple diffusion modules to increase the redundancy of the watermark. In the Middle Flow module, the watermark information M is embedded into the hidden space of the image for the first time. This step realizes the watermark embedding at the deep semantic level of the feature map. To alleviate the gradient vanishing problem, the feature maps of the Middle Flow module are jump-connected to the feature maps of the Entry Flow module. The SENet [24] mechanism is introduced to adaptively adjust the channel weights of the feature maps to enhance the expressiveness of important features, thus improving the model performance. During the up-sampling process, the watermark information is redundantly embedded twice after one up-sampling and at the completion of up-sampling, respectively. First embedding: after one up-sampling, the watermark information is embedded into the medium resolution feature map to realize the watermark embedding at the medium semantic level. Second embedding: after up-sampling, the watermark information is embedded into the high-resolution feature map to realize the watermark embedding in the spatial domain. Both embeddings are concatenated with the encoded features corresponding to the spatial dimension in the down-sampling stage, and SENet adjusts the channel weights of the spliced features. Finally, the feature maps after upsampling and watermark embedding are concatenated with the original carrier image X_O, and the concatenated features are fused using 1×1 convolution to generate the final coded image X_W.

To ensure the invisibility of the watermark, the Mean Square Error Loss (MSE) between the encoded image X_W and the original carrier image X_O is used to constrain the model training.The MSE Loss achieves the level of invisibility perceived by the human eye by minimizing the pixel-level differences that make the encoded image visually almost indistinguishable from the original image. The loss for the encoder is as follows:

$$L_{\text{en}} = \frac{1}{n}\sum_{i=1}^{n}(E_n(X_{O_i}, \sigma, M_i), X_{O_i}) \tag{1}$$

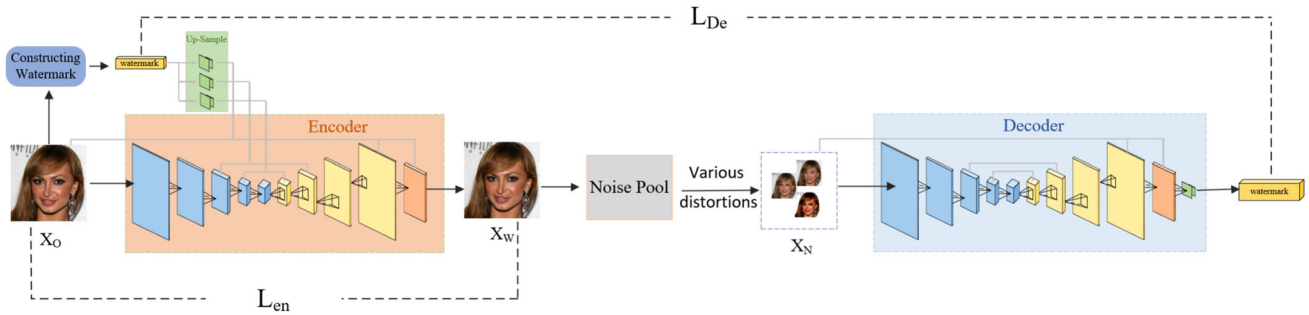

Fig. 3 Architecture diagram of XUNet. The subsampling in the Encoder section adopts the Entry Flow and Middle Flow structures from the Xception network. The overall architecture resembles U-Net, featuring three skip connections between the subsampling and upscaling paths. The Decoder employs the same model design as the Encoder

where σ adjusts the watermark information multiplier parameter, E_n is the Encoder M_i is the watermark information, and X_{O_i} is the carrier image.

3.2.2 Decoder

Decoder adopts the Xception architecture as the front-end framework. It uses its Depthwise Separable Convolution to effectively remove the influence of the image color channels and focus on extracting the watermark information from the grayscale channels. This design allows the Decoder to decouple the main information of the watermark from the encoded image X_W while reducing the interference of the color channel on the watermark extraction. The output of the Decoder is first passed through a diffusion module that integrates the extracted redundant watermark information. Subsequently, the redundant watermark information is mapped back to the length of the original watermark information through a linear layer to recover the complete watermark content. The Decoder and Encoder have the same overall structure, but the model parameters are not shared.

The loss L_{de} of the Decoder uses the mean square error loss, which is used to measure the difference between the extracted watermarked information M_{de} and the original watermarked information M. The formula is expressed as:

$$L_{\mathrm{de}} = \frac{1}{n}\sum_{i=1}^{n}\left(\mathrm{D_e}(X_{\mathrm{W_i}}), M_i\right)_{\omega} \tag{2}$$

where M is the watermark information, D_e is the Decoder and X_{W_i} is the encoded image.

4 Experiments

4.1 Experimental settings

Our experimental design is based on the CelebA-HQ dataset, using smaller validation and test sets due to limited computational resources while ensuring that the training set is large enough to guarantee the generalization ability of the model. Training set: 15,107 images, validation set: 1,889 images, and test set: 1,500 images.SimSwap is used as a typical DeepFakes model for end-to-end training and testing. GANimation [25] is used for generalizability testing to verify the robustness of the MDD watermarking framework under different counterfeiting methods. The regular perturbation sets are Identity, JpegTest, Resize, GaussianBlur, MedianBlur, Brightness, Contrast, Saturation, Hue, Dropout, Saltpepper, GaussianNoise. Cropping is not included in the set. The cropping operation is difficult to regard as a common deformation of the whole face and, therefore, not included in the noise pool. We use PyTorch to implement MDD-Watermarking. Training and testing are performed on NVIDIA RTX 4090. The training period is 120 epochs, and the batch size is 16. The Adam [26] optimizer is used with an initial learning rate of 0.002 and a weight decay of 0.00001. The watermark amplification factor σ is set to 0.1. The Encoder loss factor λ_{en} is 10, and the Decoder loss factor λ_{de} is 1.

In our experiments, we use a variety of evaluation metrics to comprehensively measure the performance of MDD-Watermarking and XUNet, including the visual quality of encoded images, the visual quality of reconstructed images, and the robustness of watermark extraction. We use PSNR, SSIM, and LPIPS [27] to evaluate the visual quality of coded images and reconstructed images.PSNR is evaluated using the whole test dataset. SSIM and LPIPS use only 50 random test samples due to their small values and the difficulty of visualizing the classification effect with too many test samples. The average bit error rate (BER) is used to measure the accuracy of watermark extraction. The original watermark information M and the extracted watermark information M_{de} are converted to binary form by $B()$ operation. Then, the difference between the two is calculated using logical difference or operation (XOR). The specific formula is as follows:

$$BER = \frac{1}{B}\frac{1}{L}\sum_{i=1}^{B}\sum_{j=1}^{L}\left(XOR\left(B(M^{i\times j}), B(M^{F^{i\times j}})\right)\right) \times 100\% \tag{3}$$

$$B(M) = \begin{cases} 1, & M > 0 \\ 0, & M \le 0 \end{cases} \tag{4}$$

To ensure the accuracy and fairness of the experiments, we selected a variety of existing watermarking methods as baseline models and compared them with the MDD-Watermarking framework. These baseline models include MBRS [28], CIN [29], PIMoG [30], and SepMark [20], where SepMark uses its robust watermarking module. In order to fully evaluate the performance of different models in different scenarios, we used two independent experimental setups for training and testing for image sizes of 128 × 128 and 256 × 256.

4.2 Experimental results

4.2.1 Watermarking robustness

Table 1 shows the results of the MDD-Watermarking framework compared with the baseline models (MBRS, CIN, PIMoG, SepMark) on the visual quality assessment metrics (PSNR, SSIM, LPIPS). The MDD-Watermarking framework outperforms the best-performing CIN model on visual quality assessment metrics on 128×128 sized images, close to the best-performing CIN model, while significantly outper-

Table 1 Image quality evaluation metrics

Model Image Size	MBRS [28] 128×128	CIN [29] 128×128	PIMOG [30] 128×128	SepMark [20] 128×128	MDD 128×128	SepMark [20] 256×256	MDD 256×256
PSNR ↑	33.0456	42.4135	37.7271	38.5112	40.9913	38.5646	43.7579
SSIM ↑	0.8106	0.9628	0.9470	0.9588	0.9538	0.9328	0.9558
LPIPS ↓	0.0141	0.0006	0.0086	0.0028	0.0046	0.0080	0.0098

Table 2 Watermark robustness test results under regular perturbation

	128×128					256×256	
Distortion	MBRS [28]	CIN [29]	PIMoG [30]	SepMark [20]	MDD	SepMark [20]	MDD
Identity	0.0000%	0.0000%	0.0366%	0.0000%	0.0000%	0.0000%	0.0000%
JpegTest $Q = 50$	0.2597%	2.7514%	19.5562%	0.2136%	2.3215%	0.0075%	0.0659%
Resize $p = 50\%$	0.0000%	0.0000%	0.0767%	0.0059%	0.0000%	0.0000%	0.0000%
GaussianBlur $k = 3, \sigma = 2$	0.0000%	22.7786%	0.1169%	0.0024%	0.0000%	0.0000%	0.0000%
MedianBlur $k = 3$	0.0000%	0.0307%	0.0992%	0.0012%	0.0020%	0.0000%	0.0000%
Brightness $f = 0.5$	0.0000%	0.0000%	1.3443%	0.0059%	0.0133%	0.0017%	0.0180%
Contrast $f = 0.5$	0.0000%	0.0000%	0.8121%	0.0012%	0.0020%	0.0000%	0.0041%
Saturation $f = 0.5$	0.0000%	0.0000%	0.0803%	0.0000%	0.0000%	0.0000%	0.0000%
Hue $f = 0.1$	0.0000%	0.0000%	0.1523%	0.0000%	0.0000%	0.0000%	0.0000%
Dropout $p = 50\%$	0.0000%	0.0000%	0.4828%	0.0000%	0.0000%	0.0058%	0.0000%
SaltPepper $p = 10\%$	0.0000%	0.0378%	2.3667%	0.0413%	0.0000%	0.0008%	0.0007%
GaussianNoise $\sigma = 0.1$	0.0000%	0.0000%	12.7396%	0.7460%	0.0685%	0.0578%	0.0021%

forming MBRS and PIMoG. Its superior performance on PSNR and SSIM metrics indicates high visual quality at the pixel level and structural similarity, while its performance on LPIPS metrics verifies the superiority in perceptual quality. The MDD-Watermarking framework significantly outperforms SepMark in PSNR and SSIM metrics at 256 × 256 image size, indicating its superior performance in visual quality. Although it is slightly inferior to SepMark in LPIPS metrics, the difference is small, and the overall performance is still excellent. These results fully demonstrate the competitiveness of the MDD-Watermarking framework in terms of visual quality.

Table 2 shows the average BER of MDD-Watermarking with baseline models (MBRS, CIN, PIMoG, SepMark) under different regular perturbation scenarios. At 128×128 image size, the average BER of MDD is significantly better than PIMoG and SepMark in most of the perturbation scenarios, and is equal or close to MBRS and CIN. The average BER of the MDD-Watermarking framework is equal to or lower than that of SepMark under most perturbation scenarios at 256×256 image size, indicating that it performs well in terms of robustness. In particular, under Dropout and GaussianNoise perturbations, the BER of MDD is significantly lower than that of SepMark, verifying its superiority in these scenarios. These results further demonstrate the reliability and superiority of MDD-Watermarking in practical applications.

Table 3 Watermark robustness test results under malicious perturbation

Distortion		SimSwap [31]	GANimation [25]
128×128	MBRS [28]	19.3744%	0.0000%
	CIN [29]	48.5068%	0.0000%
	PIMoG [30]	8.6745%	0.4802%
	SepMark [20]	13.8255%	0.0000%
	MDD	11.8817%	0.0000%
256×256	SepMark [20]	7.9068%	0.0020%
	MDD	1.7209%	0.0000%

Under the malicious distortion shown in Table 3, the MDD-Watermarking framework outperforms the BER under both SimSwap and GANimation deep forgery models. In particular, under the GANimation perturbation, the BER of MDD is 0.0000%, which is on par with MBRS, CIN, and SepMark and significantly better than PIMoG. Under the SimSwap perturbation, the BER of MDD is significantly lower than that of CIN and MBRS, slightly lower than that of SepMark, and close to that of PIMoG. These results verify the MDD-Watermarking framework under the robustness and adaptability in deep forgery scenarios.

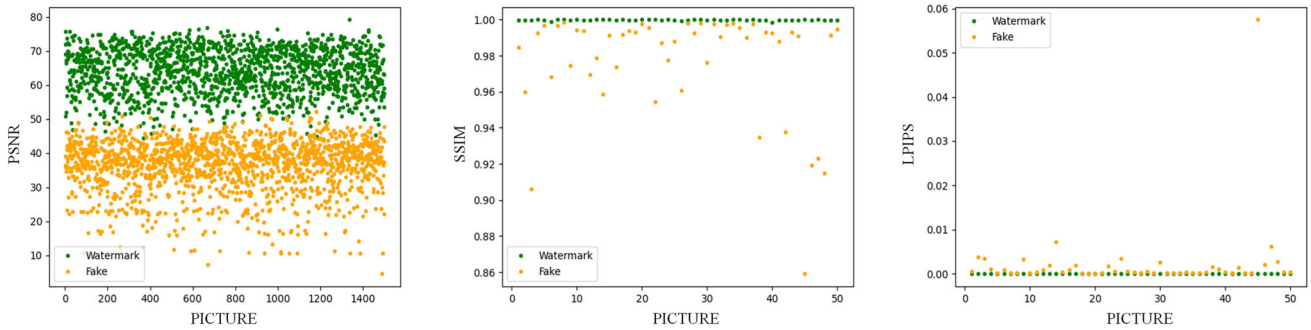

Fig. 4 Distribution of image evaluation metrics for the reconstructed image of the watermarked image and the reconstructed image after SimSwap forgery for 256×256 image size, green dots are the watermarked image, orange dots are the forged image, PSNR evaluation metrics are used on the left side, SSIM evaluation metrics are used on the center, and LPIPS evaluation metrics are used on the right side

4.2.2 Face forgery detection

In this section, we discuss the effectiveness of MDD-Watermarking for detecting face forgery images. We test it under the SimSwap forgery model and the GANimation forgery model, respectively. It is worth mentioning that our MDD-Watermarking is designed based on 256×256 size images. Although we also conducted experiments at 128×128, the detection of the realized watermark is far less effective than the 256×256 size image.

Figure 4 shows the test results under the two forgery models, SimSwap. From the PSNR metrics, the watermarked images are generally concentrated in the higher image quality region, while the forged images are mainly distributed in the lower image quality region, which creates an obvious stratification between the two. In testing with 1,500 image pairs, 32 watermarked images exhibited a PSNR below 50 compared to their reconstructed counterparts, while 11 forged images showed a PSNR above 50 against their reconstructions. Using PSNR exceeding 50 as the authenticity criterion, this method achieves an accuracy rate (ACC) of 99.97%. As clearly demonstrated by the SSIM and LPIPS metrics in Fig. 4, significant gaps exist between watermarked images and forged images when compared against their respective reconstructions. We propose that when both genuine and forged image samples are available, directly comparing the quality metric differences between each image and its reconstruction enables more intuitive and reliable authenticity discrimination. Detection results for 256×256 sized images under GANimation attacks, along with detection performance for 128×128 sized images under both SimSwap and GANimation attacks, can be found in the supplementary materials. Current experimental results demonstrate that the MDD-watermarking technique exhibits a pronounced layered effect in image quality assessment metrics, providing a reliable theoretical foundation and data support for subsequent deep learning detection. Future research may explore passive detection methods integrating deep learning models based on this approach to further enhance detection accuracy.

4.3 Ablation study

In order to improve the performance of the coded images in terms of the human eye perception effect, instead of focusing only on the pixel-level accuracy, we introduce the discriminator of GAN in XUNet. Specifically, the discriminator discriminates the carrier image and the coded image, and the loss of the discriminator is back-propagated into the loss function of the encoder to optimize the coding process. However, this approach faces some challenges: first, the discriminator loss itself is volatile, resulting in a less stable training process; second, since the watermark information embedded in different images varies and the perturbations introduced by the coded image after NP vary, these factors together lead to large fluctuations in the decoder loss and the encoder loss. This volatility affects the convergence and final performance of the model.

5 Conclusion

In this paper, we propose a watermarking method with detection and traceability functions, called MDD-Watermarking, and innovatively apply it to the field of active Deepfake forensics, realizing a unified framework for source tracking and Deepfake detection. In order to support the efficient embedding and extraction of MDD-Watermarking, we design a specialized watermarking Encoder and Decoder named XUNet. XUNet can fully use the characteristics of MDD-Watermarking, efficiently embed the watermark into the carrier image, and accurately extract the watermark information under various interference conditions. The experimental results show that XUNet is significantly robust to multiple distortions, while the reconstructed image based on MDD

watermarking can effectively distinguish the forged image from the encoded image, which further verifies the practicality and reliability of MDD-Watermarking in Deepfake forensics. This research result provides a new solution for digital image authenticity verification and forgery traceability.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s11760-025-04832-y.

Author Contributions M.Z. (Mengya Zhang): Conceptualization, Writing—original draft, Writing—review & editing. Q.Z. (Qinghui Zhang): Conceptualization, Writing—review & editing. X.W. (Xiaohan Wang): Software, Validation, Writing—original draft, Writing—review & editing. H.W. (Huaming Wu): Writing—review & editing. W.L. (Wenjuan Li): Formal, Analysis, Data curation.

Funding This work is funded in part by the National Natural Science Foundation of China (No. 62066011, No. 62401198), in part by the National Key Research and Development Program of China (2023YFC3321501), in part by the Science and Technology Project of Henan Province (No. 252102321028), and the grant 22ZZRDZX41, KFJJ2023013.

Data Availability No datasets were generated or analysed during the current study.

Declarations

Competing interests The authors declare no competing interests.

References

1. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial networks. Commun. ACM **63**(11), 139–144 (2020)
2. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Adv. Neural. Inf. Process. Syst. **33**, 6840–6851 (2020)
3. Liang, P., Liu, G., Xiong, Z., Fan, H., Zhu, H., Zhang, X.: A facial geometry based detection model for face manipulation using cnn-lstm architecture. Inf. Sci. **633**, 370–383 (2023)
4. AlDuaij, N.: Veracos: an operating system extension for the veracity of files. Comput. Secur. **157**, 104565 (2025)
5. Chen, Z., Xie, L., Pang, S., He, Y., Zhang, B.: Magdr: Mask-guided detection and reconstruction for defending deepfakes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9010–9019 (2021)
6. Rossler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., Nießner, M.: Faceforensics++: Learning to detect manipulated facial images. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 1–11 (2019)
7. Durall, R., Keuper, M., Keuper, J.: Watch your up-convolution: Cnn based generative deep neural networks are failing to reproduce spectral distributions. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7890–7899 (2020)
8. Wang, G., Jiang, Q., Jin, X., Li, W., Cui, X.: Mc-lcr: multimodal contrastive classification by locally correlated representations for effective face forgery detection. Knowl.-Based Syst. **250**, 109114 (2022)
9. Liu, D., Dang, Z., Peng, C., Zheng, Y., Li, S., Wang, N., Gao, X.: Fedforgery: generalized face forgery detection with residual federated learning. IEEE Trans. Inf. Forensics Secur. **18**, 4272–4284 (2023)
10. Ahmed, K.: 2dots-multi-bit-encoding for robust and imperceptible image watermarking. Multimed. Tools Appl. **80**(2), 2395–2411 (2021).
11. Rabeah, N., Ahmed, K., Aaliya, S.: High performance and energy efficient image watermarking for video using a mobile device. Wirel. Pers. Commun. **104**, 1535–1551 (2019).
12. Yu, N., Skripniuk, V., Abdelnabi, S., Fritz, M.: Artificial fingerprinting for generative models: Rooting deepfake attribution in training data. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 14448–14457 (2021)
13. Wang, R., Juefei-Xu, F., Luo, M., Liu, Y., Wang, L.: Faketagger: Robust safeguards against deepfake dissemination via provenance tracking. In: Proceedings of the ACM International Conference on Multimedia, pp. 3546–3555 (2021)
14. Fernandez, P., Sablayrolles, A., Furon, T., Jégou, H., Douze, M.: Watermarking Images in Self-Supervised Latent Spaces (2022). https://arxiv.org/abs/2112.09581
15. Yan, Z., Zhang, Y., Fan, Y., Wu, B.: Ucf: Uncovering common features for generalizable deepfake detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 22355–22366 (2023).
16. Yan, Z., Zhang, Y., Yuan, X., Lyu, S., Wu, B.: Deepfakebench: a comprehensive benchmark of deepfake detection. In: In Proceedings of the International Conference on Neural Information Processing Systems, pp. 4534–4565 (2023)
17. Baluja, S.: Hiding images in plain sight: Deep steganography. Advances in Neural Information Processing Systems **30**, (2017)
18. Zhu, J., Kaplan, R., Johnson, J., Fei-Fei, L.: Hidden: Hiding data with deep networks. In: Proceedings of the European Conference on Computer Vision, pp. 657–672 (2018)
19. Tancik, M., Mildenhall, B., Ng, R.: Stegastamp: Invisible hyperlinks in physical photographs. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2117–2126 (2020)
20. Wu, X., Liao, X., Ou, B.: Sepmark: Deep separable watermarking for unified source tracing and deepfake detection. In: Proceedings of the ACM International Conference on Multimedia, pp. 1190–1201 (2023)
21. Poobathy, D., Chezian, R.M.: Edge detection operators: peak signal to noise ratio based comparison. IJ Image, Graph. Signal Process. **10**, 55–61 (2014)
22. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 234–241 (2015)
23. Chollet, F.: Xception: Deep learning with depthwise separable convolutions. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1251–1258 (2017)
24. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7132–7141 (2018)
25. Pumarola, A., Agudo, A., Martinez, A.M., Sanfeliu, A., Moreno-Noguer, F.: Ganimation: Anatomically-aware facial animation from a single image. In: Proceedings of the European Conference on Computer Vision, pp. 818–833 (2018)
26. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
27. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 586–595 (2018)

28. Jia, Z., Fang, H., Zhang, W.: Mbrs: Enhancing robustness of dnn-based watermarking by mini-batch of real and simulated jpeg compression. In: Proceedings of the ACM International Conference on Multimedia, pp. 41–49 (2021)
29. Ma, R., Guo, M., Hou, Y., Yang, F., Li, Y., Jia, H., Xie, X.: Towards blind watermarking: Combining invertible and non-invertible mechanisms. In: Proceedings of the ACM International Conference on Multimedia, pp. 1532–1542 (2022)
30. Fang, H., Jia, Z., Ma, Z., Chang, E.-C., Zhang, W.: Pimog: An effective screen-shooting noise-layer simulation for deep-learning-based watermarking network. In: Proceedings of the ACM International Conference on Multimedia, pp. 2267–2275 (2022)
31. Chen, R., Chen, X., Ni, B., Ge, Y.: Simswap: An efficient framework for high fidelity face swapping. In: Proceedings of the ACM International Conference on Multimedia, pp. 2003–2011 (2020)