2025 International Conference on Intelligent Computing
July 26-29, Ningbo, China
https://www.ic-icc.cn/2025/index.php

A Robust Image Blind Watermarking Scheme Based on Staged Adaptive Strategy

Hu Deng[1,2][0009-0002-3081-3718], Feng Chen[1,2][0000-0003-3130-9273], Pei Gan[1,2][0009-0000-3051-414X], Rongtao Liao[1,2][0009-0005-8269-4871], and XueHu Yan[1,2(✉)][0000-0001-6388-1720]

[1] College of Electronic Engineering, National University of Defense Technology, 230031 Hefei, China
[2] Anhui Province Key Laboratory of Cyberspace Security Situation Awareness and Evaluation, 230031 Hefei, China
denghu@nudt.edu.cn

Abstract. Image blind watermarking serves as a critical tool for copyright protection of digital images. Existing watermarking schemes usually perform well under a single noise condition. However, in practical applications, watermarked images are often exposed to many different types of noise. This combined noise condition significantly degrades the image visual quality and watermark extraction accuracy of existing watermarking schemes. To address these challenges, we propose a novel two-stage training strategy that enhances watermarking robustness by training the model with various noise intensities, improving performance under combined noise conditions. To further improve the imperceptibility of the watermarked image while ensuring high accuracy of watermark extraction, we propose a strength balanced watermarking optimization algorithm in the model testing phase. Furthermore, since JPEG compression has a non-differentiable operation, existing schemes can not effectively obtain satisfactory watermarking performance for JPEG compression. We introduce a differentiable fine-grained JPEG compression module to improve the robustness of existing schemes for JPEG compression. The results of the experiment indicate our proposed scheme outperforms state-of-the-art schemes under multiple noise conditions. Under noise-free condition, it achieves a 0% bit error rate and 53.55 dB PSNR, and under combined noise conditions, it still achieves an average of 2.40% bit error rate and 42.70 dB PSNR.

Keywords: Image blind watermarking, Staged training strategy, JPEG compression.

1 Introduction

With the explosive growth of digital content, protecting intellectual property rights and ensuring the integrity of information have become particularly important [1], [2]. Image blind watermarking technique serves as a key tool, offering robust copyright protection and verification for digital images with high imperceptibility [3], [4]. However, with the diversification of multimedia applications and technological advances, traditional watermarking schemes gradually show their limitations, especially in dealing with combined noise conditions and maintaining the balance between robustness and imperceptibility [5].

Traditional image watermarking schemes are usually divided into pixel and frequency domain schemes. The pixel domain-based schemes perform the watermark embedding process by adjusting the pixel data of the image, such as pixel value adjustment, pixel block shifting, pixel feature encoding [6-9]. These schemes are favored for their good imperceptibility, but exhibit low robustness against various attacks.In contrast, frequency domain-based scheme perform the watermarked embedding process by adjusting the coefficients in the transform domain of the image, such as DCT coefficient adjustment[10], [11], DWT coefficient modification [12], [13]. These schemes improve attack resistance at the expense of imperceptibility.

Rapid advances in deep learning have prompted researchers to apply it to digital watermarking, resulting in numerous innovative schemes. [14-18] These schemes mainly focus on improving models [19], training strategies [20] and noise layer design [21] to enhance model performance. However, these models tend to perform poorly under combined noise conditions. This is because various types of noise require more additional watermark residual information to maintain a high decoding rate, thereby reducing the imperceptibility of the watermark image.

In deep learning models, the noise layer is essential for enabling the model to effectively learn various noise features and withstand different types of distortions. Consequently, it is important to identify functions that accurately model real-world noise. For most of the differentiable digital distortions, researchers have proposed relevant mathematical functions to improve the robustness under such distortions, while JPEG compression presents a challenge because it involves non-differentiable operations such as quantization and cropping [22]. As a result, existing models exhibit a low watermark decoding rate under JPEG compression distortion, failing to meet the practical application standards for watermark decoding.

Following the research and analysis presented above, we can summarize the limitations and challenges faced by current image watermarking schemes: existing schemes fail to effectively balance watermark imperceptibility and robustness under combined noise conditions. Moreover, since JPEG compression involves non-differentiable operations, it restricts the model's ability to learn the noise distribution [23]. Consequently, most models exhibit poor robustness against JPEG compression, limiting their practical application. To solve the limitations discussed above, we design a novel two-stage training scheme for model training. In the testing phase, we introduce a strength balanced watermarking optimization algorithm, enabling the model to achieve an optimal trade-off between watermarking imperceptibility and robustness under combined noise conditions. To enhance the model's robustness against JPEG compression, we introduce a differentiable fine-grained JPEG compression module, closely mimicking real JPEG compression. Experimental results demonstrate superior outcomes compared to previous schemes.

In summary, the main contributions of this paper include the following three aspects:

- A novel two-stage watermarking model training strategy is proposed, which effectively ensure the robustness of watermarking under combined noise conditions.

- The strength balanced watermarking optimization algorithm is proposed to optimize the imperceptibility and robustness of watermarks in the testing phase.
- A differentiable JPEG compression module is introduced to enhance the model's robustness against JPEG compression distortions.

The rest of this paper is structured as follows: Section 2 reviews related work in the field. Section 3 presents a detailed description of the proposed method. Section 4 describes the experimental setup and evaluation criteria. Finally, Section 5 concludes the paper and discusses potential future work.

2 Related Work

2.1 Image Blind Watermarking Based on Deep Learning

The deep-learning based image blind watermarking model was first proposed by Hamidi et al. [24] in 2017. By employing autoencoder convolutional neural networks (CNNs), it brings better imperceptibility and robustness than traditional schemes. HiDDeN [14] was the first paper introducing adversarial networks to image blind watermarking. It was the first end-to-end scheme using neural networks. The end-to-end scheme enables joint learning of the encoder model and decoder model, but training both modules makes it challenging to optimize the trade-off between watermark robustness and imperceptibility. Liu et al. [20] proposed the TSDL training strategy. The scheme consists of two phases: noise-free, end-to-end adversarial training phase, and noise-aware, decoder-focused training stage. This scheme effectively counters black-box noise. However, further improvements are required in terms of watermark robustness and imperceptibility under combined noise conditions. Motivated by these advancements, we design a novel two-stage training strategy. By using different noise strengths with loss functions for training in different stages, the robustness and imperceptibility of digital watermarked images can be significantly improved.

2.2 Strength Factor

Strength factor was explored by several scholars within the realm of deep-learning based watermarking schemes. Redmark [18] and TSDL [20] employed the strength factor to control the robustness and imperceptibility of watermarking, yet the overall enhancement was limited without significant improvements. MBRS [21] used the strength factor during testing to adjust the PSNR values for different models, enabling a clearer comparison of watermark decoding rates. However, the in-depth exploration of the strength factor remains insufficient. Later, the Adaptor scheme [25] proposed an adaptive strength factor training strategy that introduces a novel loss function during the training phase, aiming to ensure high watermark decoding rates while maximizing PSNR and SSIM values for each image. Nevertheless, this scheme necessitates retraining the model for different datasets, thereby lacking generalizability. To solve this limitation, we design a strength balanced watermarking optimization algorithm to generate watermarked images with high imperceptibility against various types and intensities of distortions.

2.3 JPEG Simulations

Among the various types of distortions faced by watermarking, existing deep learning schemes can not effectively measure watermarking performance under JPEG compression [23]. This limitation arises because the rounding and truncation operations inherent in JPEG compression are non-differentiable. Consequently, scholars have proposed several schemes. HiDDeN [14] introduced JPEG-Mask and JPEG-Drop schemes as alternatives to actual JPEG compression. The two-stage training scheme proposed by TSDL [20] represents an improvement over simulation schemes. However, since only the decoder is trained in the second stage, information loss after compression occurs, affecting final performance. MBRS [21] proposes a training strategy that combines the real and the simulated JPEG compression and has achieved significant results. However, when trained jointly with other noises, the robustness of MBRS [21] against JPEG compression decreases because adding other noises distorts the original combined training mechanism. To address these challenges, this paper introduce a refined JPEG compression function module, which is integrated into the noise layer for direct application during training, similar to other differentiable noise types.

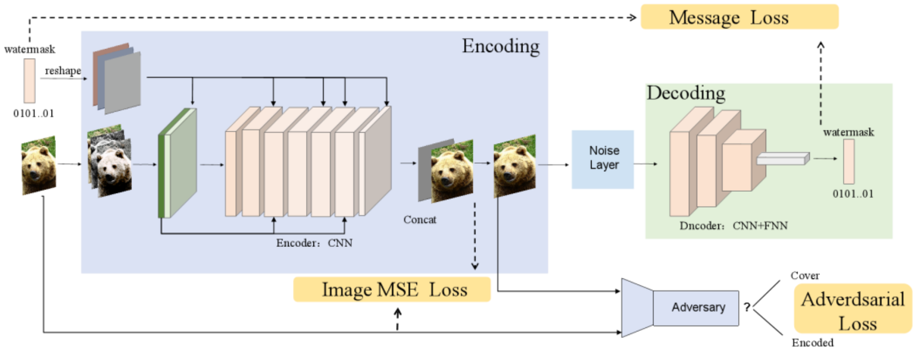

Fig. 1. Overall framework of the model.The framework has four modules: (1) Encoder, (2) Decoder, (3) Noise Layer, and (4) Adversary

3 Proposed Method

3.1 Model Architecture

Fig. **1** provides an overview of our model's comprehensive architecture. It has four parts: (1) Encoder: The encoder contains parameters θ_{en}, the inputs are the original image $I_{ori} \in R^{C\times H\times W}$ and original watermark $W_{ori} \in \{0,1\}^{L}$, and output is the residual image I_{res}, thus the watermarked image I_{enc} is obtained as follow,

$$\mathrm{I_{enc}} = \mathrm{I_{ori}} + \mathrm{I_{res}} \tag{1}$$

(2) Noise layer: a joint noise layer is constructed by fitting a mathematical function to various noises. The input to noise layer is the watermarked image I_{enc} and the output is the noise image I_{noi}. (3) Decoder: The decoder, also with parameters θ_{de}, takes the noisy image I_{noi} as input and outputs the decoded wateramrk W_{rec}. (4) Adversary: The adversary has parameters θ_{di} to distinguish original image I_{ori} and watermarked image I_{enc}. The main function of the adversary is to distinguish the difference between the watermarked image I_{enc} and the original image I_{ori}, so as to further

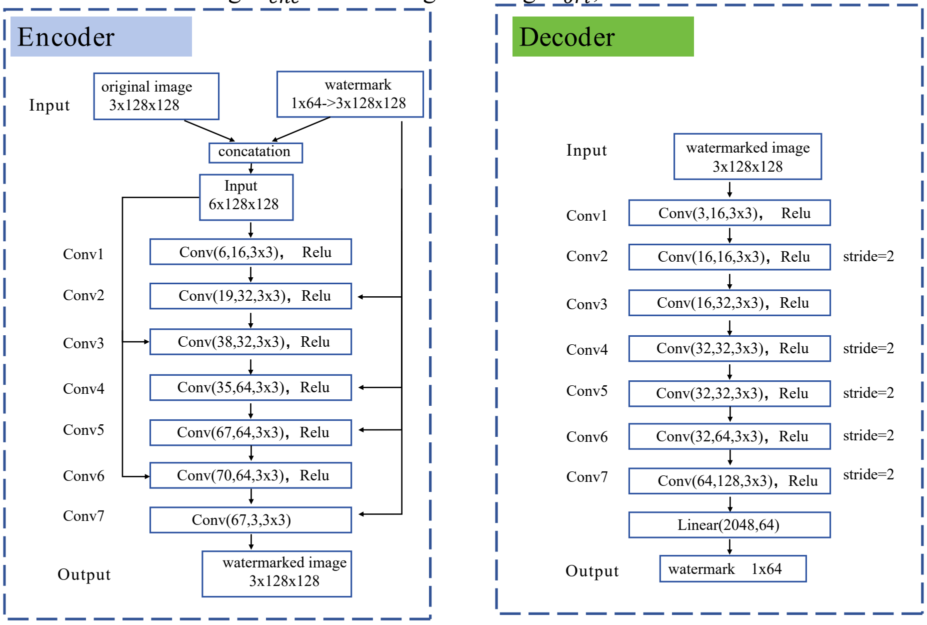

Fig. 2. Detailed structure diagram of encoder and decoder

Encoder. The encoder architecture of our model is shown in the let figure of Fig **2**. The encoder E is a network trained with parameters θ_{en} to embed the watermark into the original image. The watermark W_{ori} is expanded redundantly into a form of $3 \times H \times W$, matching the dimensions of the original image. Then, this extended watermark is concatenated with the input image to produce a residual image I_{res}. Then this residual image is added to original image to produce the watermarked image. The encoder consists of seven convolutional layer. To capture features from both the watermark and cover image at multiple levels, the outputs from layers 2, 4, 5, and 7 are concatenated with the watermarked image, respectively, while the outputs from convolutional layers 3 and 6 are spliced with the original encoded image. Utilizing these multi-level convolutional mappings allows for more efficient learning of the watermark pattern. An MSE loss function is applied during the training of encoder, aimed at minimizing the difference between original image the watermarked image.

$$L_{E_1} = MSE(I_{ori}, I_{enc}) = MSE(I_{ori}, E(\theta_{en}, I_{ori}, W_{ori})) \quad (2)$$

Decoder. The decoder D, equipped with parameters θ_{de}, is trained to extract the watermark from a noisy image $\mathrm{I}_{\mathrm{noi}}$. As shown in the right figure of Fig. **2**, the main structure of the decoder is similar to the encoder, using seven convolutional layers and one fully connected layer. To ensure the final output has the same structure as the original watermark W_{ori}, Conv1 and Conv3 employ a stride size of 1, following the default configuration, whereas the stride sizes of the remaining layers are set to 2 for reducing the feature tensor size. Following Conv7, the feature maps are flattened into a vector, which is passed through a fully connected layer to generate a one-dimensional output tensor of the same dimensionality as the watermark W_{ori}. We also applies MSE loss function to train the decoder, which to minimize the difference between the recovered watermark information $W_{rec}$and the original watermark information W_{ori}, aiming to make each bit the same.

$$L_{De} = MSE(W_{ori}, W_{rec}) = MSE(W_{ori}, E(\theta_{en}, I_{noi})) \quad (3)$$

Adversary. In our scheme, the adversary consists of three layers of 3×3 convolutional layers followed by a common pooling layer for classification. The adversary is utilized to predict whether an image contains watermark information. It is trained by minimizing the following loss function.
Update parameters θ_{di} to minimize

$$L_A = log(1 - A(\theta_{di}, E(\theta_{En}, I_{ori}, W_{ori}))) + log(A(\theta_{di}, I_{ori})) \quad (4)$$

Then update parameters θ_{En} to minimize

$$\mathrm{L}_{\mathrm{E}_2} = \log(\mathrm{A}(\theta_{\mathrm{di}}, \mathrm{I}_{\mathrm{enc}})) = \log(\mathrm{A}(\theta_{\mathrm{di}}, \mathrm{E}(\theta_{\mathrm{En}}, \mathrm{I}_{\mathrm{ori}}, \mathrm{W}_{\mathrm{ori}}))) \quad (5)$$

Noise layer. The noise layer is a crucial component of the model for achieving robustness. Common differentiable noises, such as crop, gaussian filtering (GF), gaussian noise (GN), resize, and salt&pepper(S&P) noise, can be directly modeled using existing mathematical functions. However, for non-differentiable noise types like JPEG compression, existing fitting functions are not particularly effective. To address this issue, we introduce a refined JPEG compression function module designed in Reich et al. [26]. According to our research, this represents the first instance of applying this particular scheme in the field of digital watermarking. This JPEG compression module performs differentiable processing on quantization, rounding down, truncation, and other operations, making it directly applicable for training deep learning models when highly fitting real JPEG compression. Specifically, for differentiable quantization, the JPEG compression module uses the polynomial approximation proposed by Shin et al. [27] to approximate the quantization operation using the rounding function.

$$\mathrm{x} = \lfloor \mathrm{x} \rceil + (\mathrm{x} - \lfloor \mathrm{x} \rceil)^3 \quad (6)$$

In the quantization step, a given JPEG quality Q is mapped to a scale factor s. The scaling factor s is calculated as follows,

$$s(\mathrm{Q}) = \begin{cases} \frac{5000}{\mathrm{Q}} & \mathrm{Q} < 50 \\ 200 - 2 * \mathrm{Q} & \mathrm{Q} \geq 50 \end{cases} \quad (7)$$

Since the scaling factor s and the corresponding quantization table need to be stored as integers, a differentiable scale floor function is necessary. The JPEG compression module utilizes polynomial rounding to approximate the downward rounding function, offering advantages over alternative schemes.

$$s = \lfloor s - 0.5 \rfloor + (s - 0.5 - \lfloor s - 0.5 \rfloor)^3 \quad (8)$$

For differentiable quantization table (QT) clipping, the JPEG compression module employs a differentiable clipping technique to constrain the values within an appropriate range.

$$\mathrm{clip}(x) = \begin{cases} 0 + 0.001 * (x - 0) & \text{if } x < 0 \\ x & \text{if } x \in [0, 255] \\ 255 + 0.001 * (x - 255) & \text{if } x > 255 \end{cases} \quad (9)$$

3.2 Model Training and Testing Strategy

The method employed in our scheme comprises three steps: two training stages and a testing stage for the adaptive control of watermarking performance through strength factor adjustment. This process primarily uses the principle of generating residual images robust against various types of high-intensity noises during the training phase. Subsequently, a strength balanced watermarking optimization algorithm is applied in the testing phase to produce high-quality watermark images suitable for different noise intensities. The entire process of model training and testing strategy is shown in the Fig. **3**.

In the first training stage, the model is trained under various strong noise distortion conditions. To enhance stability and accelerate training speed, only the loss function of the decoder is utilized in this phase, as represented by the following equation.

$$\mathrm{L}_1 = \lambda_{\mathrm{D}} \mathrm{L}_{\mathrm{De}} \quad (10)$$

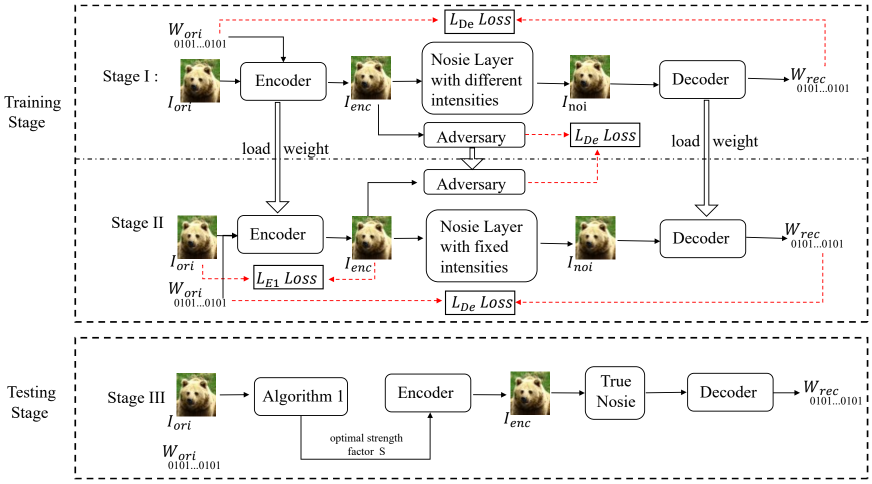

Fig. 3. The overall pipeline of model training and testing strategy.

At the same time, to improve the robustness of the model under combined noise conditions, we choose to train a single model that can withstand a variety of noises, rather than training separate models for each type of noise. The noise layer $\mathrm{N_{pool}}$ we used is shown as follows. When training the model, the watermarked image is randomly distorted by one of the noises in the noise pool $\mathrm{N_{pool}}$ before decoding, which ensures that the model has good robustness to various types of noises.

$$\begin{aligned}\mathrm{N_{pool}} = \{&\mathrm{Identity(), Crop(p_c), S\&P(p_p), Cropout(p_o), GN(\delta),}\\ &\mathrm{GF(\sigma), Resize(p_r),, Dropout(p_d), JPEG(Q)}\}\end{aligned}$$

As shown in the corresponding variables in Table **1**, during the training process, the model randomly selects different types of noises with varying intensities each time. Consequently, the model learns generalized features under diverse noise conditions. The first stage of training is designed to enable the decoder to efficiently learn robust features for various types of noise.

In the second stage, building upon the robust decoder obtained from the first stage of training, the model is further trained using all the loss functions. This ensures that the model benefits from both the initial robustness and the optimization across all parameters.

$$\mathrm{L_2 = \lambda_E L_{E_1} + \lambda_D L_{De} + \lambda_A L_{E_2}} \tag{11}$$

And the training is continued using a fixed medium intensity noise as shown in Table 1. This training strategy improves the imperceptibility of watermark while maintaining the robustness of the watermark.

In the testing phase, to further improve the quality of the watermarked image, we designed a strength balanced watermarking optimization algorithm to measure the quality of the watermarked image and the decoding accuracy, the algorithm process is shown in Algorithm 1. The core principle behind our algorithm is that since the

Table 1. The parameter settings of noise pool in different training stage.

Parameter	p_c	p_p	p_o	δ
First stage value	(0,50%]	(0,10%]	(0,50%]	(0,10%]
Second stage value	3.5%	10%	10%	10%
Parameter	σ	p_r	p_d	Q
First stage value	[2,25]	(0,1]	(0,50%]	[10,50]
Second stage value	2	50%	30%	10

watermarked image I_{enc} is derived from the direct addition of the original image I_{ori} and the residual image I_{res}, the robustness and imperceptibility of the watermark are mainly influenced by the residual image. Therefore, we use the strength factor S to adjust the weight of the residual image during the testing phase. The relationship can be represented as follows:

$$I_{enc} = I_{ori} + I_{res} * S \tag{12}$$

By adjusting S, we aim to identify an optimal value in different noise conditions that ensures watermark imperceptibility while maintaining an acceptable decoding loss rate.

4 Experiments

To validate the effectiveness of our scheme, this section presents a =introduction to the experimental details and results. The content encompasses five main parts: First, we describe the datasets and explain how we chose the evaluation metrics. Second, we detail the experimental steps, including model training and testing setups. Third, we analyze the scheme's performance about watermark imperceptibility and robustness. Fourth, we compare our scheme with state-of-the-art schemes to show its advantages. Finally, ablation experiments verify the effectiveness of our refined JPEG compression module and explain why our model is simple yet effective.

4.1 Datasets and Evaluation Metrics

Datasets. Similar to other schemes, we randomly selected 10,000 natural images from the COCO dataset as the training set and an additional 1,000 images as the validation set. We used 5000 COCO datasets different from the training and validation sets as the test set. To better demonstrate the scheme's generalization ability, we randomly selected 1000 images from the Mini-ImageNet dataset of different categories as the test set. The evaluation will be based on the average performance metrics reported on test set.

Metrics. To evaluate the performance of our scheme, we use a series of quantitative metrics. The robustness of watermarks measured by Bit Error ratio (BER), which refers to the error between original watermark M_{ori} and extracted watermark M_{rec}. The imperceptibility of Watermarked images I_{enc} measured by Signal-to-Noise Ratio (PSNR) and Structural Similarity (SSIM). The formula for metric is shown below.

$$\mathrm{BER}(\%) = (\tfrac{1}{L} \times \textstyle\sum_{k=1}^{L} |M_{ori} - M_{rec}|) \times 100\% \tag{13}$$

$$\mathrm{PSNR}(I_{ori}, I_{enc}) = 20 \times \log_{10} \frac{\mathrm{MAX}(I_{ori}, I_{enc}) - 1}{\mathrm{MSE}(I_{ori}, I_{enc})} \tag{14}$$

$$\mathrm{SSIM}(I_{ori}, I_{enc}) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{15}$$

Algorithm 1 Method of the Strength Balanced Watermarking Optimization

```
Input: Original images Iori, watermarks Wori, error_rate
Output: BER, PSNR, SSIM, optimal strength factor S
Initialize parameters:
    $S \leftarrow 0.01$
    $step\_size \leftarrow 0.02$
    $max\_attempts \leftarrow 50$
    $optimal\_S \leftarrow S$
    $attempt \leftarrow 0$
while $attempt < max\_attempts$ do
    Encode $W_{ori}$ into $I_{ori}$ with strength $S \rightarrow$ Generate $I_{enc}$ and $W_{rec}$
    Compute PSNR and SSIM between $I_{ori}$ and $I_{enc}$
    Compute BER between $W_{ori}$ and $W_{rec}$
    if $BER < error\_rate$ then
    $optimal\_S \leftarrow S$
    $S \leftarrow S + step\_size$
    $attempt \leftarrow attempt + 1$
    else
        Break loop
    end if
end while
Output optimal_S, corresponding BER, P SNR, SSIM
```

Baseline. Our baselines for comparison are HiDDeN [14], MBRS [21], Adaptor[25] and MCFN [28]. Their works all use convolutional neural networks as the model framework and use strength factors to control the experimental results. HiDDeN [14] and MBRS [21] opened the source code of their model. We are unable to achieve the best performance they reported. Adaptor [25] and MCFN [28] are not yet open-source. To ensure fair comparison with their reported results, we directly adopt the results published in MCFN [28].

4.2 Implementation Details

The model training in this scheme is divided into two stages. During training and testing stage, all images are uniformly resized to 128 × 128. To maintain consistency with other schemes, we randomly embed L=64 bits of information into each image. The entire training process is conducted on NVIDIA GeForce RTX 3090.

In the first stage, the initial learning rate is set to 1e-3, with a batch size 32, weight factor λ_D is 10. Additionally, the learning rate is automatically halved every 20 epochs to ensure rapid convergence in the early training stages and more precise learning of watermark robustness features. In the second stage, the initial learning rate is set to 1e-4 and a reduced batch size 8, weight factor λ_E is 6.18, λ_D is 10 and λ_A is 0.001.

Both stages involve training for 100 epochs and employ the Adam optimizer with default hyperparameters. For the noise layer, we use a randomly mixed high-intensity noise layer, as described earlier. This scheme aims to enhance the model's resistance to various distortions by exposing it to different noises, thereby improving the robustness in practical applications.

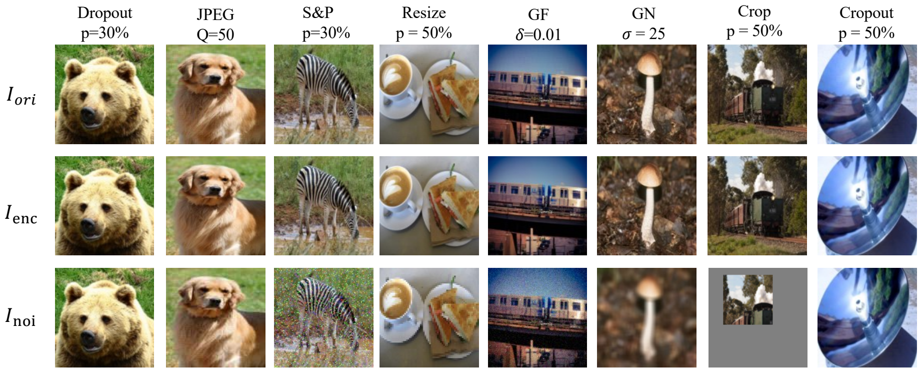

Fig. 4. Qualitative results of our scheme under different noises. A separate type of noise is represented by each column. Top: original image $\mathrm{I_{ori}}$; Middle: encoded image $\mathrm{I_{enc}}$; Bottom: noisy image $\mathrm{I_{noi}}$.

4.3 Imperceptibility and Robustness Tests

We evaluate the results of our scheme using a model trained under combined noise conditions. Extensive tests were conducted with varying noise intensities, and Fig. **4** shows the visual quality results of each noise. Each column shows the test results for each specific noise, and the name of the noise and the noise intensity are shown in the first row. The next two rows are the original image $\mathrm{I_{ori}}$, and the watermarked image $\mathrm{I_{enc}}$. We can find that the images $\mathrm{I_{ori}}$ and $\mathrm{I_{enc}}$ are almost visually indistinguishable, which shows that our scheme has good invisibility. The bottom row shows the noise images $\mathrm{I_{noi}}$ under different noises.

Table 2 lists the detailed quantitative experimental results corresponding to Fig. **4**. The SSIM and PSNR values of the watermarked images under each type of noise are not the same because different types and intensities of noise affect the watermarked images differently. In the Identity case, our scheme achieves a PSNR value of 53.55 and a BER of less than 0.01%, which demonstrates the high imperceptibility and robustness of the scheme. Table 3 lists the detailed quantitative experimental results under two types of distortions.The PSNR values of the watermarked images obtained under other noises also reach above 30 when high decoding rates are guaranteed. It shows that there is basically no gap visually with the original image.

Table 3. The parameter settings of noise pool in different training stage.

Noise	Factor	combined			
		BEE(%)	PSNR(dB)	SSIM	S
Identity	-	0.0	53.55	0.995	0.064
Dropout	p = 30%	0.0	44.14	0.968	0.192
GF	δ = 0.01	0.0	43.72	0.966	0.186
Resize	p = 50%	0.0	48.15	0.985	0.121
S&P	p = 10%	0.0	39.13	0.921	0.317
GN	σ = 25	4.16	41.49	0.947	0.247
Cropout	p = 50%	7.80	45.50	0.973	0.169
Crop	p = 50%	7.26	45.90	0.975	0.154
RealJPEG	Q = 50	0.01	36.61	0.888	0.434
Average	-	2.40	42.70	0.953	0.227

Table 3. The robustness and imperceptibility results of two types of combined noise across 1000 images in the Mini ImageNet dataset. The meaning of ‘SP (0.1)+Crop (0.5)’ is that the watermarked image is first processed by SP(0.1) and then processed by Crop(0.5) to obtain a noisy image containing two types of distortions.

Noise（Factor）	combined			
	BEE(%)	PSNR(dB)	SSIM	S
SP(0.1)+Crop(0.5)	3.92	33.14	0.808	0.638
SP(0.1)+RealJpeg(50)	4.11	32.62	0.80	0.68
RealJpeg(50)+GF(2)	5.00	31.69	0.78	0.758
RealJpeg(50)+Crop(0.5)	4.45	32.82	0.805	0.667
RealJpeg(50)+Resize(0.5)	4.29	35.33	0.87	0.508
GN(0.01)+Crop(0.5)	3.56	39.89	0.931	0.291

4.4 Comparison with the State-of-the-arts

To fairly evaluate the performance of our scheme compared with other schemes, we use the same Dataset as MCFN [28] and randomly selected 5000 images as the test set for the experiment. Table 4 shows the values of each metric for each scheme under different noises, and our scheme not only achieves a higher PSNR than other schemes but also excels in robustness tests.HiDDeN [14], MBRS [21], Adaptor[25] and MCFN [28]

To further illustrate the performance of our scheme, a comprehensive analysis is conducted which is shown in Table 4. Under Dropout (p=70%), our scheme achieves a PSNR of 50.80 dB, which is higher than HiDDeN[14] (35.04 dB), MBRS[21] (41.6 dB), Adaptor[25] (44.53 dB) and MCFN [28] (49.75dB). Similarly, under Crop (p =70%), our scheme maintains a high PSNR of 48.46 dB, outperforming all other schemes. In terms of JPEG compression (Q = 50), our scheme achieves a PSNR of 36.31 dB, which is significantly higher than other schemes. Notably, our scheme also achieves the lowest Bit Error Rate (BER) across all noise types. This comprehensive evaluation confirms that our scheme enhances the imperceptibility and robust of watermark under various types of noise conditions.

Table 4. Results of robustness and imperceptibility compared with different models under 5000 images in COCO dataset.

Models	Metrics	Noise					
		GN	GF	S&P	Dropout	Crop	JPEG
		δ=0.05	σ=2	p_p=1%	p_d=70%	p_c=70%	Q=50
HiDDeN	PSNR(dB)	29.92	33.49	34.90	35.04	38.68	30.99
	SSIM	0.902	0.919	0.943	0.930	0.965	0.858
	BER(%)	22.0	12.0	28.0	22.0	21.0	35.2
MBRS	PSNR(dB)	33.19	42.78	42.73	41.6	39.59	31.08
	SSIM	0.832	0.973	0.973	0.975	0.956	0.832
	BER(%)	0.063	0.0035	0.0016	0.013	0.98	0.0022
Adaptor	PSNR(dB)	34.8	45.37	44.15	44.53	41.92	31.15
	SSIM	0.873	0.984	0.981	0.982	0.972	0.841
	BER(%)	0.0034	0.0030	0.0016	0.011	0.86	0.0024
MCFN	PSNR(dB)	35.15	**47.67**	**48.27**	49.75	47.79	31.56
	SSIM	**0.919**	**0.991**	**0.986**	0.990	**0.993**	0.848
	BER(%)	0.0022	0.0024	0.00014	0.00062	0.047	0.001
Ours	PSNR(dB)	**37.55**	45.14	46.78	**50.80**	**48.46**	**36.31**
	SSIM	0.902	0.974	0.982	**0.992**	0.987	**0.884**
	BER(%)	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.00006**

4.5 Ablation Experiments

To verify the advantage of the two-stage training strategy proposed in our scheme, the model is trained with L_2 total loss in only one stage under strong mix noise. The experimental results show that the decoding rate of watermarks in the training stage is around 50%. This is because the model can not learn to be robust to each strong noise under the action of multiple loss functions, which leads to poor robustness. We also verify the advantage of the model we used. We replace our model with the model used by MBRS [21]. During the training stage, MBRS [21] model performs well on the training set with a BER of 0, while the BER on the validation set is as high as 20%, indicating that the model is not able to effectively learn the mixed noise features, leading to overfitting of the model. Excessive model complexity may be the cause of the problem. MBRS [21] contains a large number of attention modules which tend to focus more on feature extraction from the original image. In contrast, our scheme use only a combination of convolutional and watermarked embeddings, allowing the model to focus more on the learning of the residual image, which can help the model converge and achieve better results.

In addition, to verify the advantage of the JPEG differentiable compression layer used in our scheme, we replace its original mixed training scheme with our noise layer based on MBRS [21] scheme. We also compare with the scheme JPEG-SS used in model HiDDeN [14], and the experimental results obtained are shown in Table 5. The experimental results show that our scheme can be closer to the real JPEG compression and get better results in the robustness test.

Table 5. Comparison among MBRS, JPEG-SS, and our method. The result shows that our method is more robust.

Methods	Q=90		Q=50		Q=30	
	BER(%)	SSIM	BER(%)	SSIM	BER(%)	SSIM
Ours	**0.00**	31.71	**0.00**	31.93	**2.48**	31.92
MBRS	0.01	**33.28**	2.75	**36.89**	12.46	**36.90**
JPEG-SS	2.11	31.98	0.55	31.96	7.32	31.97

5 Conclusion

In this paper, in order to further improve the robustness and imperceptibility of watermarking in combined noise conditions, we propose a two-phase training strategy aimed at obtaining a model with high robustness to multiple types of noise. In the testing phase, we employ a strength balanced watermarking optimization algorithm to further improve the imperceptibility of the watermarking while ensuring high robustness. Moreover, to enhance the robustness of the existing scheme to JPEG compression, we introduce a fine-grained JPEG compression module. Experimental results show that our scheme outperforms other existing schemes and demonstrates significant practical value. Specifically, the scheme allows the selection of optimal strength factors based on the application scenarios to achieve the best performance. In future studies, we will continue to optimize the algorithm to improve its efficiency and extend the application scenarios of this scheme so that it can work effectively in more complex distortion conditions, such as print-and-scan or print-and-shoot scenarios.

Acknowledgments. This work is funded by the National Natural Science Foundation of China under Grant 62271496.

References

1. Kun Hu, Mingpei Wang, Xiaohui Ma, Jia Chen, Xiaochao Wang, and Xingjun Wang. Learning-based image steganography and watermarking: A survey. Expert Systems with Applications, 249:123715, 2024.
2. Khalid M Hosny, Amal Magdi, Osama ElKomy, and Hanaa M Hamza. Digital image watermarking using deep learning: A survey. Computer Science Review, 53:100662, 2024.
3. Wolfram Szepanski. A signal theoretic method for creating forgery-proof docu- ments for automatic verification. In Carnahan Conf. on Crime Countermeasures, Lexington, KY, pages 101–109, 1979.
4. I Cox, M Miller, J Bloom, J Fridrich, and T Kalker. Digital watermarking and steganography morgan kaufmann publishers. Amsterdam/Boston, 2008.

5. Om Prakash Singh, Amit Kumar Singh, Gautam Srivastava, and Neeraj Kumar. Image watermarking using soft computing techniques: A comprehensive survey. Multimedia Tools and Applications, 80(20):30367–30398, 2021.
6. Dipti Prasad Mukherjee, Subhamoy Maitra, and Scott T Acton. Spatial domain digital watermarking of multimedia objects for buyer authentication. IEEE Trans- actions on multimedia, 6(1):1–15, 2004.
7. Irene G Karybali and Kostas Berberidis. Efficient spatial image watermarking via new perceptual masking and blind detection schemes. IEEE Transactions on Information Forensics and security, 1(2):256–274, 2006.
8. Guang Hua, Yong Xiang, and Leo Yu Zhang. Informed histogram-based water- marking. IEEE Signal Processing Letters, 27:236–240, 2020.
9. Nan-Run Zhou, Long-Long Hu, Zhi-Wen Huang, Meng-Meng Wang, and Guang- Sheng Luo. Novel multiple color images encryption and decryption scheme based on a bit-level extension algorithm. Expert Systems with Applications, 238:122052, 2024.
10. Mohamed Hamidi, Mohamed El Haziti, Hocine Cherifi, and Mohammed El Has- souni. Hybrid blind robust image watermarking technique based on dft-dct and arnold transform. Multimedia Tools and Applications, 77:27181–27214, 2018.
11. Nan-Run Zhou, Jia-Wen Wu, Ming-Xuan Chen, and Meng-Meng Wang. A quan- tum image encryption and watermarking algorithm based on qdct and baker map. International Journal of Theoretical Physics, 63(4):100, 2024.
12. Shabir A Parah, Javaid A Sheikh, Jahangir A Akhoon, Nazir A Loan, and Ghu- lam M Bhat. Information hiding in edges: A high capacity information hiding tech- nique using hybrid edge detection. Multimedia Tools and Applications, 77:185–207, 2018.
13. Tamer Rabie and Ibrahim Kamel. Toward optimal embedding capacity for trans- form domain steganography: a quad-tree adaptive-region approach. Multimedia Tools and Applications, 76:8627–8650, 2017.
14. J Zhu. Hidden: hiding data with deep networks. arXiv preprint arXiv:1807.09937, 2018.
15. Bingyang Wen and Sergul Aydore. Romark: A robust watermarking system using adversarial training. arXiv preprint arXiv:1910.01221, 2019.
16. Xiyang Luo, Ruohan Zhan, Huiwen Chang, Feng Yang, and Peyman Milanfar. Dis- tortion agnostic deep watermarking. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 13548–13557, 2020.
17. Seung-Min Mun, Seung-Hun Nam, Haneol Jang, Dongkyu Kim, and Heung-Kyu Lee. Finding robust domain from attacks: A learning framework for blind water- marking. Neurocomputing, 337:191–202, 2019.
18. Mahdi Ahmadi, Alireza Norouzi, Nader Karimi, Shadrokh Samavi, and Ali Emami. Redmark: Framework for residual diffusion watermarking based on deep networks. Expert Systems with Applications, 146:113157, 2020.
19. Chaoning Zhang, Philipp Benz, Adil Karjauv, Geng Sun, and In So Kweon. Udh: Universal deep hiding for steganography, watermarking, and light field messaging. Advances in Neural Information Processing Systems, 33:10223–10234, 2020.
20. Yang Liu, Mengxi Guo, Jian Zhang, Yuesheng Zhu, and Xiaodong Xie. A novel two-stage separable deep learning framework for practical blind watermarking. In Proceedings of the 27th ACM International conference on multimedia, pages 1509– 1517, 2019.
21. Zhaoyang Jia, Han Fang, and Weiming Zhang. Mbrs: Enhancing robustness of dnn-based watermarking by mini-batch of real and simulated jpeg compression. In Proceedings of the 29th ACM international conference on multimedia, pages 41–49, 2021.
22. Gregory K Wallace. The jpeg still picture compression standard. Communications of the ACM, 34(4):30–44, 1991.

23. David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning repre- sentations by back-propagating errors. nature, 323(6088):533–536, 1986.
24. Haribabu Kandi, Deepak Mishra, and Subrahmanyam RK Sai Gorthi. Exploring the learning capabilities of convolutional neural networks for robust image water- marking. Computers & Security, 65:247–268, 2017.
25. Baowei Wang, Yufeng Wu, and Guiling Wang. Adaptor: Improving the robustness and imperceptibility of watermarking by the adaptive strength factor. IEEE Trans- actions on Circuits and Systems for Video Technology, 33(11):6260–6272, 2023.
26. Christoph Reich, Biplob Debnath, Deep Patel, and Srimat Chakradhar. Differen- tiable jpeg: The devil is in the details. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pages 4126–4135, 2024.
27. Richard Shin and Dawn Song. Jpeg-resistant adversarial images. In NIPS 2017 workshop on machine learning and computer security, volume 1, page 8, 2017.
28. Yufeng Wu, Baowei Wang, Guiling Wang, and Xin Liao. MCFN: Multi-scale crossover feed-forward network for high performance watermarking. Neurocom-puting, 622:129282, 2025.