

WAN: Watermarking Attack Network

Seung-Hun Nam¹

shnam1520@gmail.com

In-Jae Yu²

myhome98304@gmail.com

Seung-Min Mun²

qkqhd222@gmail.com

Daesik Kim¹

daesik.kim@webtooncorp.com

Wonhyuk Ahn^{†1}

whahnize@gmail.com

¹ NAVER WEBTOON Corp.

Seongnam, South Korea

² Samsung Electronics

Suwon, South Korea

Abstract

Multi-bit watermarking (MW) has been developed to improve robustness against signal processing operations and geometric distortions. To this end, benchmark tools that test robustness by applying simulated attacks on watermarked images are available. However, limitations in these general attacks exist since they cannot exploit specific characteristics of the targeted MW. In addition, these attacks are usually devised without consideration of visual quality, which rarely occurs in the real world. To address these limitations, we propose a watermarking attack network (WAN), a fully trainable watermarking benchmark tool that utilizes the weak points of the target MW and induces an inversion of the watermark bit, thereby considerably reducing the watermark extractability. To hinder the extraction of hidden information while ensuring high visual quality, we utilize a residual dense blocks-based architecture specialized in local and global feature learning. A novel watermarking attack loss is introduced to break the MW systems. We empirically demonstrate that the WAN can successfully fool various block-based MW systems. Moreover, we show that existing MW methods can be improved with the help of the WAN as an add-on module.

1 Introduction

Digital watermarking is a technique used to protect copyright by embedding identification information, referred to as watermark, into the original image [1, 2]. Unlike visible watermarking, which inserts a watermark perceptible by the human visual system (HVS), invisible watermarking is an approach that embeds imperceptible watermarks [3]. In particular, multi-bit watermarking (MW), which is a representative example of invisible watermarking, has been actively researched so that multi-bit information can be extracted from the watermarked image [4, 5]. MW inserts watermarks by considering the fundamental requirements: *Imperceptibility*, which is the degree of invisibility of the watermarked signal, and *Robustness*, which is the ability of the watermark to survive against various watermarking attacks [6, 7].

[†] Corresponding author

© 2021. The copyright of this document resides with its authors.

It may be distributed unchanged freely in print or electronic forms.

Imperceptibility is assessed using image quality assessment (IQA) metrics, which evaluate visual quality degradation caused by the watermark embedding. To assess robustness, a benchmark tool composed of various attacks, such as StirMark [29, 30] and CheckMark [27], is applied to a watermarked image. These tools assess the robustness of the watermarking system by determining how well the inserted watermark can survive these simulated attacks. However, these tools attack watermarked images in a general way without considering the context of the watermarking system, so they cannot dig into the specific weak points of the watermarking system [35]. Moreover, these attacks degrade the visual quality beyond what is acceptable for commercial usage in the process of interfering with watermark extraction by deteriorating the image.

Instead, malicious users can design effective attacks to remove the watermark by targeting the MW and without visual degradation [35], which further deepens the gap between attacks in the real-world and existing benchmark tools. In this case, the designers of the watermarking system can assume a worst-case attack, where the watermark embedding and extraction algorithms are public (i.e., Kerckhoffs’s principle [13]), to make systems more robust against adversaries. In this context, designing novel benchmark tools to create tests that are adequate for individual, specific watermarking systems to induce the false extraction of inserted information while maintaining a high quality level for the content is required.

Motivated by the need for watermarking designers, who want to analyze the vulnerability of their MW methods, we propose a watermarking attack network (WAN) that exploits the weak points of individual watermarking systems without compromising visual quality. As illustrated in Fig. 1, the proposed WAN is devised to hinder the extraction of inserted watermarks by adding interference signals to mislead the watermarking extractor. With proposed loss function, our work can both induce abnormal extraction and generate a reconstructed image with a visual quality similar to the original content. We determine that the residual dense block-based architecture’s ability to learn local and global features is suitable for analysing each MW method composed of various procedures and detailed attributes such as the watermarking domains and embedding algorithms [40]. Moreover, we show that our WAN can be used as an add-on module to further improve the performance of existing MW methods. The main contributions are listed as follows.

- To the best of our knowledge, this is the first attempt to successfully introduce a convolutional neural network (CNN)-based watermarking attack framework for interfering with the watermark extraction of MW systems.
- Compared to existing benchmark tools [27, 29, 30], the WAN induces abnormal watermark extraction while conserving perceptual quality. For specific MW methods [15, 18, 19, 24, 33], the WAN can apply subtle modification to induce the extraction of the watermark bit embedded in the image in an inverted state (e.g., $0 \rightarrow 1$ or $1 \rightarrow 0$).

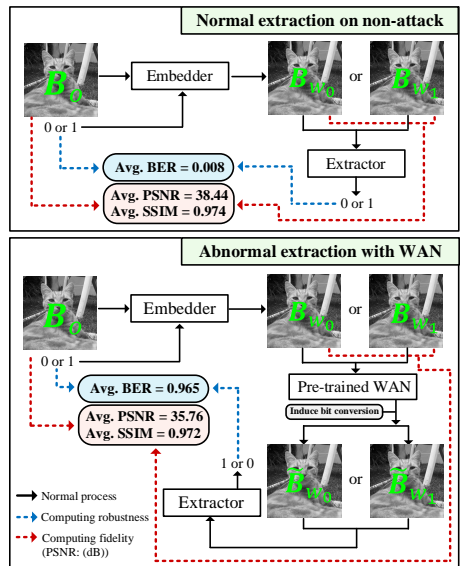


Figure 1: How the watermarking attack network works.

- We present one possible usage of WAN that pre-trained WAN can be used as an add-on module, which yields additional performance gains (imperceptibility or robustness) on the existing rule-based MW methods.

2 Background

In this section, we review the concept of MW and the limitations of existing benchmark tools for testing robustness against watermarking attacks.

2.1 Multi-bit Watermarking

Rather than using zero-bit watermarking to detect the presence or the absence of a watermark, MW can be used in various applications since the n -bit-long message ($\mathbf{m} = \{0, 1\}^n$) can be inserted in the host image I_o to get

a watermarked image I_w . In particular, block-based MW methods [15, 18, 19, 24, 33], which insert a watermark bit (0 or 1) in each original block B_o , are mainly used for multi-bit information insertion rather than the keypoint-based approach [23] due to the benefits that can be achieved by utilizing the entire domain. As shown in Table 1, the attributes of block-based MW methods vary, and each attribute is determined by considering the aimed performance and media’s inherent properties¹. In general, the transform domain as discrete wavelet transform (DWT) [51], discrete cosine transform (DCT) [4], nonsubsampling contourlet transform (NSCT) [8], singular value decomposition (SVD) [12], and QR decomposition (QRD) [10] is first applied to each pre-processed block, and then watermark embedding is performed by applying an embedding algorithm such as spread spectrum (SS) [3, 9], improved spread spectrum (ISS) [20], quantization (QT) [17, 36], and embedding for causing differences between sub-groups (DIF) [19, 33] to the selected domain.

The block-based MW system consists of an embedder and extractor, and procedures for watermark embedding and extraction are performed independently for each block (see Fig. 2). The block-wise embedding function follows: $B_w = E_m(B_o, \mathbf{b}, A_d)$ where \mathbf{b} and A_d denote an assigned watermark bit and auxiliary data, respectively. By applying E_m to each B_o , constituting I_o , I_w containing message \mathbf{m} with a capacity of n can be obtained. In the case where n is set to 1, I_w and B_w are equal. In the extraction phase, the estimated message $\hat{\mathbf{m}}$ can be obtained by merging $\hat{\mathbf{b}}$, which is extracted from each B_w , and the block-wise extraction function takes the following form in the case of a blind fashion: $\hat{\mathbf{b}} = E_x(B_w, A_d)$ where the blind fashion denotes that the original image is not required [52]. In the case where n is set to 1, $\hat{\mathbf{m}}$ and $\hat{\mathbf{b}}$ are equal. The performance of MW is evaluated in terms of imperceptibility and robustness. Specifically, the visual differences between the original and watermarked images are determined using the IQA metrics, such as peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) [37], and robustness is evaluated by calculating the bit error rate (BER) between \mathbf{m} and $\hat{\mathbf{m}}$.

MW method	Watermarking domain	Embedding algorithm	Size of minimum unit	Extraction approach / Key characteristic
M1 [15]	DCT	SS	1×64 (MV)	Blind / Template-based
M2 [18]	DCT	ISS	16×16 (MV)	Blind / Multiple watermarks
M3 [19]	QRD	DIF	8×8 (MV)	Blind / Low false positive rate
M4 [33]	DWT, SVD	DIF	8×8 (MV)	Blind / Considering HVS
M5 [33]	NSCT	QT	—	Blind / Perceptual masking

† MV is abbreviation of the majority voting that aggregates results of minimum units.

Table 1: List of MW methods with attributes.

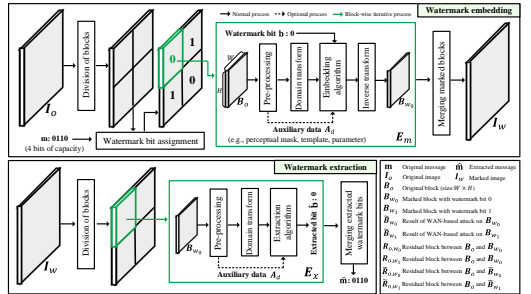


Figure 2: A general overview of the MW system.

¹More details are provided in the supplementary material.

2.2 Watermarking Attack and Motivation

Watermarking attacks are employed to evaluate the robustness of MW methods; let \tilde{B}_w be the attacked block image of B_w . By comparing the watermark bit extracted from B_w and \tilde{B}_w , a MW designer can evaluate the robustness of the MW by determining whether the hidden information survived [65]. Currently, StirMark [29, 50] and CheckMark [27] are the representative benchmark tools that provide various types of common attacks such as signal processing operations and geometric distortions. As can be seen in Fig. 3, common watermarking attacks mounted in StirMark are accompanied by visual degradation and have a limitation of not being able to model the vulnerabilities of each MW method. That is, the more that a watermarking attack utilizes the characteristics of the targeted watermarking system, the more effective the attack is possible without image quality degradations.

With the development of neural networks, CNN-based MW methods [21, 22] have been newly proposed, and they can be neutralised with adversarial attacks attempting to fool watermarking systems through malicious inputs; these are referred to as adversarial examples. However, attacking numerous handcrafted MW methods that contain non-differentiable operators with an adversarial attack is difficult. Although there is a differential evolution-based attack [32] that randomly modifies one pixel and queries the extractor, it is difficult to disable robust MW systems with only a few pixel changes. To address these issues, we propose a CNN-based watermarking attack that automatically learns and exploits the weak points of individual watermarking systems.

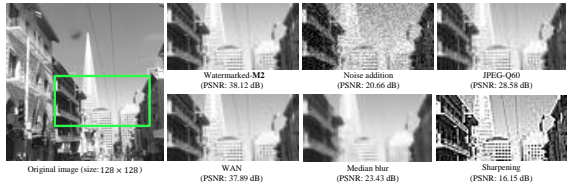


Figure 3: Comparison of fidelity between StirMark attacks causing extraction of random guessing and WAN.

3 Watermarking Attack Network (WAN)

The proposed WAN targets block-based MW and needs one triple set of block images, of B_o , watermarked images with bit 0 B_{w_0} , and watermarked images with bit 1 B_{w_1} , in the training phase. WAN takes B_{w_0} and B_{w_1} as inputs and reconstructs each of them into attacked images \tilde{B}_{w_0} and \tilde{B}_{w_1} , respectively. Our goal is to reconstruct images that mislead the watermarking extractor to decide on the wrong bit. In other words, when \tilde{B}_{w_0} and \tilde{B}_{w_1} are considered to have been inserted 1 bit and 0 bit, we judge the attack to be successfully done. On the other hand, the attacked image should be similar to the original to minimize visual degradation. We start with in-depth descriptions of loss functions consisting of watermarking attack loss and content loss and provide detailed descriptions of the architecture of the network and the mini-batch configuration.

3.1 Loss Function

The proposed WAN is trained to reconstruct attacked images containing an inverted watermark bit while minimizing the visual quality degradation. To achieve this, we propose a customized loss as an objective function to train the WAN as follows: $\mathcal{L} = \lambda_{wa}\mathcal{L}_{wa} + \lambda_c\mathcal{L}_c$, where \mathcal{L}_{wa} and \mathcal{L}_c represent watermarking attack loss, which is devised to change an inserted bit and content loss to minimize visual degradation, respectively. λ_{wa} and λ_c indicate predefined weight terms for each loss.

3.1.1 Watermarking Attack Loss

Existing watermarking methods vary in terms of the watermarking domains and embedding algorithms, so it is difficult to theoretically model MW in a single system. Moreover, con-

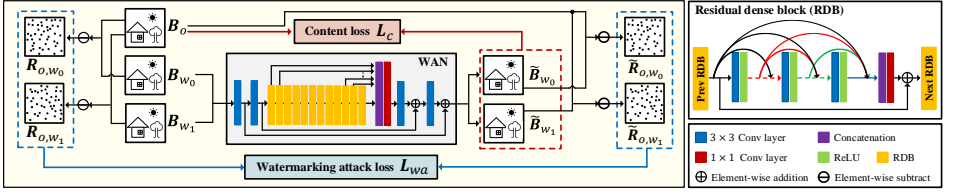


Figure 4: Schematic illustration of the proposed WAN.

ventional MW methods incorporate non-differentiable operations, so it is difficult for the neural network to learn directly from these methods even though step-by-step instructions are publicly available. We simplify this problem as the watermark signal is added to the original image in the pixel domain, and focus on the noise patterns that are decided by bit information. In other words, the residual signal arose by bit 0 insertion $R_{o,w_0} = B_o - B_{w_0}$ and the residual signal arisen by bit 1 insertion $R_{o,w_1} = B_o - B_{w_1}$, which can be identified by neural networks. We hypothesise that the neural network can remove watermark signals in images and insert opposite noise patterns, which causes wrong bit extraction at the watermarking extractor. In this case, the attacked image \tilde{B}_{w_0} on B_{w_0} would have similar noise pattern $\tilde{R}_{o,w_0} = B_o - \tilde{B}_{w_0}$ to R_{o,w_1} , for which the one with bit 0 makes. The noise pattern \tilde{R}_{o,w_1} of the attacked image \tilde{B}_{w_1} on B_{w_1} would be similar to R_{o,w_0} , in the same way.

To capture the above observation, watermarking attack loss for the image of size $W \times H$, \mathcal{L}_{wa} is defined as follows:

$$\mathcal{L}_{wa} = \frac{1}{N} \sum_{i=1}^N |R_{o,w_0}^i - \tilde{R}_{o,w_1}^i| + \frac{1}{N} \sum_{i=1}^N |R_{o,w_1}^i - \tilde{R}_{o,w_0}^i|, \quad (1)$$

where superscript i refers to pixel location and $N = W \times H$. The first term of the equation is for deriving the watermark bit 1 inserted in B_{w_1} into 0, and the second term is for deriving bit 0 inserted in B_{w_0} into bit 1. As depicted in Fig. 4, a loss is designed by pairing the residual images before and after going through the WAN according to the inserted bit and reducing the difference between the paired images. Through the \mathcal{L}_{wa} , it is possible to add a fine noise-like attack that inverts the actually inserted bit during the process of passing the watermarked images over the WAN.

3.1.2 Content Loss

In terms of watermarking attack, it is important to preserve visual quality while adding adversarial signals. To this end, content loss is adopted to reduce the visual differences between the original content B_o and its corresponding reconstructed images, including \tilde{B}_{w_0} and \tilde{B}_{w_1} attacked by the WAN (see Fig. 4). Inspired by the papers in [42] demonstrating that ℓ_1 loss can bring better visual quality than ℓ_2 loss for general restoration tasks, the content loss of \mathcal{L}_c is defined as follows: $\mathcal{L}_c = \frac{1}{N} \sum_{i=1}^N \sum_{j=0}^1 |B_o^i - \tilde{B}_{w_j}^i|$. From \mathcal{L}_c , it is possible to conduct a watermarking attack while minimizing visual quality degradations in the original content. Through the final objective function of \mathcal{L} combined with \mathcal{L}_c and \mathcal{L}_{wa} , the proposed WAN can reconstruct images in a way that adversely affects the extraction of the inserted bit while maintaining the inherent properties of the original content.

3.2 Model Architecture

We follow the network design from the residual dense network (RDN) that is used for the learning of the local and global features and the ability of image restoration [43, 44]. The

residual dense block (RDB) constituting the RDN is composed of densely connected convolution (Conv) layers and is specialized in extracting abundant local features. In the proposed WAN, the pooling layer and up-sampling are excluded, so the input and output sizes are the same ($\{B_{w_0}, B_{w_1}, \tilde{B}_{w_0}, \tilde{B}_{w_1}\} \in \mathbb{Z}^{1 \times W \times H}$). The first and second Conv layers are placed to extract shallow features and conduct global residual learning. Next, RDBs are placed consecutively, and we expect sub-components for local residual learning and local feature fusion commonly used in RDB to help our model learn fine-grained distortions caused by watermark embedding. After that, by the concatenation layer followed by 1×1 and 3×3 Conv layers, dense local features extracted from the set of RDBs are fused in a global way. The deep part of the proposed WAN is composed for global residual learning based on shallow feature maps.

3.3 Mini-batch Configuration

Since invisible MW is the approach of inserting a watermark so that it is unnoticeable by HVS, mini-batch configuration suitable for fine signal learning is required instead of the standard mini-batch used in high-level computer vision. The authors in [15] presented paired mini-batch training, which is efficient for learning noise-like signals such as multimedia forensics [1, 18] and steganalysis [19]. To aid in learning the discriminative features between watermarked results more effectively, paired mini-batch training is employed. That is, B_{w_0} and B_{w_1} generated for the same original image B_o are allocated in a single batch, which allows the proposed WAN to learn fine signals due to the differences in the fine signals caused by the watermark bit. In detail, when the batch size is b_s , $\frac{b_s}{2} B_{w_0}$ images are selected first, and then $\frac{b_s}{2} B_{w_1}$ images corresponding to B_{w_0} are assigned to be in the same batch. The entire dataset is shuffled every epoch.

4 Experiments

4.1 Datasets

BOSSbase [5] and BOWS [4] datasets are used to generate 20,000 original grey-scale images with a size of 512×512 . We resize them to 64×64 (i.e., $W = H = 64$) using the default settings in MATLAB R2018a, the resized images are divided into three sets for training, validation, and test (with a 14 : 1 : 5 ratio). The block-based MW methods [15, 18, 19, 24, 33] are used to generate watermarked images, and the images are generated by embedding watermark bits (0 or 1) into the original images given for each method listed in Table 2. These methods perform watermark bit extraction in blind fashion. For further quantitative and qualitative evaluation on watermark capacity, we additionally generate test images sized 128×128 and 256×256 for the test set. Watermarked images with resolutions of 128×128 and 256×256 have a watermark capacity of 4 bits and 16 bits, respectively. In experiments, the WAN-based attacks and watermark bit extraction proceeds for each 64×64 patch ².

4.2 Implementation Details and Training Settings

The number of RDB, Conv layer per RDB, feature-maps, and the growth rate are set to 12, 6, 32, and 16, respectively. We build our network using PyTorch and run the experiments on NVIDIA GeForce GTX 1080 Ti. The size of mini-batch b_s is set to 32, and each mini-batch

²We provide details of datasets and additional experimental results in the supplementary material.






MW method	1 bit of watermark capacity						4 bits of watermark capacity						16 bits of watermark capacity					
	Non-attack			WAN			Non-attack			WAN			Non-attack			WAN		
	PSNR	SSIM	BER	PSNR	SSIM	BER	PSNR	SSIM	BER	PSNR	SSIM	BER	PSNR	SSIM	BER	PSNR	SSIM	BER
M1 	35.55	0.938	0.026	34.04	0.956	0.893	35.53	0.938	0.049	34.79	0.963	0.905	37.73	0.957	0.043	35.98	0.971	0.882
M2 	41.86	0.988	0	37.47	0.979	0.996	42.62	0.987	0	37.44	0.980	0.993	43.11	0.985	0	38.30	0.982	0.990
M3 	36.59	0.974	0	33.05	0.96	1.000	37.55	0.973	0	33.64	0.962	0.998	38.33	0.973	0	35.31	0.976	0.994
M4 	38.98	0.986	0.002	37.70	0.985	0.988	39.71	0.985	0.002	38.09	0.985	0.990	39.88	0.979	0.003	38.44	0.980	0.991
M5 	39.21	0.987	0.013	36.54	0.980	0.947	40.64	0.987	0.041	37.22	0.982	0.885	41.63	0.989	0.032	38.48	0.985	0.851
Average	38.44	0.974	0.008	35.76	0.972	0.965	39.21	0.974	0.018	36.24	0.974	0.954	40.14	0.977	0.015	37.30	0.978	0.942

Table 2: Quantitative evaluation results of the proposed WAN on the test set with 1 bit, 4 bits, and 16 bits watermark capacities.

is configured for paired mini-batch training [25]. We use the Adam optimizer with a learning rate of 10^{-4} and momentum coefficients $\beta_1 = 0.9$, $\beta_2 = 0.999$. The proposed WAN is trained with the hyperparameters $\lambda_c = 0.4$ and $\lambda_{wa} = 0.3$ during 30 epochs, and the best model is selected as the one that maximizes BER on the validation set for each MW method.

4.3 Quantitative Evaluation of WAN

First, a quantitative evaluation of the WAN is conducted in terms of watermark extraction interference and the visual quality of attacked images. We use IQA metrics, PSNR (dB) and SSIM [57], to determine the imperceptibility and BER to evaluate attacks to get quantitative results. The left part of Table 2 shows the performance results of our work on the test set with 1 bit capacity generated through each MW method, which are composed of various attributes. In non-attack situations, each method has a low BER value of 0.026 or less, while the average BER value increases dramatically to 0.965 after WAN is applied. In particular, for MW methods in [18, 19, 53], the BER value of methods rise to 0.988 or more, which means that the WAN has learned a fine signal generated during the watermark embedding and successfully performs bit inversion. In general, making the extraction performance at a random guessing level is considered a very fatal attack [54], and it is validated that the proposed \mathcal{L}_{wa} successfully leads to abnormal extraction of watermark bits.

In addition, minimizing the visual damage caused by watermarking attacks is an important issue in our work. To do this, we introduce \mathcal{L}_c , and the gain of visual quality obtained from the loss can be analyzed through PSNR and SSIM values with original content in Table 2. In case of 1 bit, the average PSNR and SSIM values in non-attack situation are 38.44 dB and 0.974, respectively. After the WAN attacks images, average PSNR decreases by 2.68 dB, and SSIM remained similar to that before the attack. We would like to note that existing benchmark tools [27, 50] have to degrade images up to 20.66 dB for noise addition to get a random guessed results, as shown in Fig. 3. Meanwhile, our model is capable of inducing the drastic reversal of the watermark bit with acceptable small loss of perceptual quality.

We further test for 4 and 16 bits of watermark capacity scenarios with the trained WAN model with stride 64. As listed in the middle side of Table 2, for 4 bits capacity, the average PSNR, SSIM, and BER values for the attacked images over the WAN are 36.24 dB, 0.974, and 0.954, respectively. For the results of the 16 bits capacity, the average BER value is 0.942 while achieving the improved visual quality. Compared with the results for 1 bit capacity, we can confirm that the WAN’s overall performance is maintained even when the watermark capacity is increased. Overall, the results of quantitative evaluation show that the proposed WAN is suitable for testing MW methods as a benchmark tool in terms of interference of watermark extraction, maintenance of visual quality, and scalability according to watermark capacity.

4.4 Qualitative Evaluation of WAN

Next, we perform qualitative evaluation in terms of perceptual quality. As shown in Fig. 5, the types of low-level distortion caused by watermark embedding vary by MW method while having similar high-level features (i.e., inherent content). For MW systems [14, 18, 19, 24, 33], the WAN hardly causes visual degradation in the process of inverting watermark bits (see magnified sub-figures in Fig. 5). The proposed WAN with \mathcal{L}_{wa} and \mathcal{L}_c can hinder watermark extraction by learning these fine feature and induces the attacked image to visually follow the original content.

Next, we acquire four types of residual images ($\{R_{o,w_0}, R_{o,w_1}, \tilde{R}_{o,w_1}, \tilde{R}_{o,w_0}\}$) and analyze WAN-based attacks by comparing the differences between residual images. As described in Sec. 3.1.1, the WAN is guided to reduce the difference between the paired residual images during the training phase, thereby applying an adversarial signal to B_w that causes abnormal extraction. High similarity between R_{o,w_0} and \tilde{R}_{o,w_1} located in the 2nd and 4th columns in Fig. 6 is observed, indicating that the WAN successfully attacks the watermarked image containing watermark bit 1 (i.e., $1 \rightarrow 0$). Similarly, the characteristics of residual images in the 3rd and 5th columns are co-related. Furthermore, since MW methods with various attributes are used in this study, the distribution and characteristics of each residual data vary depending on the methods. In summary, we can confirm that WAN can adaptively learn the individual characteristics of the MW method, such as watermarking domain and embedding algorithm.

Fig. 3 compares results of the our model and StirMark [24, 30] consisting signal processing operations and geometric distortions. For fairness in comparison, attacked images generated through attack parameters of StirMark that cause random guessing of bit extraction (e.g., BER = 0.5) are compared. As mentioned above, the StirMark is not an approach of attacking by modeling the vulnerability of the MW method or considering inherent content, so it is accompanied by unwanted visual degradation in the attack process (see magnified sub-figures in Fig. 3). In contrast, our WAN can adversely affect the extraction of the inserted bit while maintaining the inherent properties of the original content. From the results of qualitative evaluation, it is confirmed that the CNN architecture specialized for image restoration and the proposed loss function are effective in generating natural attacked images.

4.5 Ablation Study

To investigate how loss function \mathcal{L} contributes to the overall performance of our WAN, we conduct an ablation study on the test set with 1 bit of watermark capacity. The results are summarized in Table 3. For \mathcal{L}_c only, each MW shows outstanding performance in terms of fidelity, but the effectiveness of WAN decreases to the level of random guessing. \mathcal{L}_c induces the reconstructed image to follow the distribution of the original content. When training \mathcal{L}_{wa}

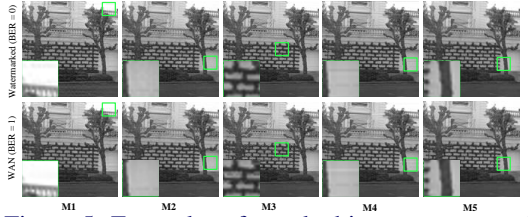


Figure 5: Examples of attacked images generated from the WAN applied to the watermarked image with 4 bits of capacity.

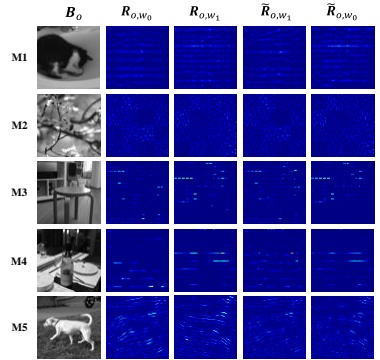


Figure 6: Visualization of residual images. \tilde{R}_{o,w_1} and \tilde{R}_{o,w_0} are reconstructed from R_{o,w_0} and R_{o,w_1} , respectively.

Loss	M1			M2			M3			M4			M5		
	PSNR	SSIM	BER	PSNR	SSIM	BER	PSNR	SSIM	BER	PSNR	SSIM	BER	PSNR	SSIM	BER
\mathcal{L}_c	38.49	0.983	0.526	43.94	0.995	0.497	41.79	0.992	0.513	42.5	0.994	0.525	41.93	0.991	0.364
\mathcal{L}_{wa}	30.38	0.905	0.914	35.55	0.964	0.996	31.48	0.947	1	34.88	0.972	0.997	34.17	0.969	0.972
\mathcal{L}	34.04	0.956	0.893	37.47	0.979	0.996	33.05	0.96	1	37.7	0.985	0.988	36.54	0.98	0.947

Table 3: Ablation study results of the proposed WAN on loss function.

only, all MW methods show a high BER value (of 0.914 or higher), but the visual quality is greatly reduced during the attack process. In particular, \mathcal{L}_{wa} -based WAN shows a tendency to rapidly improve attack performance with a BER value of 0.85 or more before 10 epochs. As described in Section 3.1, the proposed objective function \mathcal{L} is designed considering the advantages of \mathcal{L}_c and \mathcal{L}_{wa} jointly. From the results in Table 3, we can conclude that \mathcal{L} helps the model to leverage the advantages of \mathcal{L}_c and \mathcal{L}_{wa} simultaneously.






MW method	Color image					
	Non-attack			WAN		
	PSNR	SSIM	BER	PSNR	SSIM	BER
M1 	42.63	0.968	0.048	40.14	0.966	0.805
M2 	43.14	0.979	0	39.51	0.971	0.942
M3 	40.89	0.972	0	38.19	0.963	0.987
M4 	41.03	0.969	0	38.41	0.965	0.977
M5 	43.07	0.988	0.037	41.97	0.981	0.802
Average	42.15	0.975	0.017	39.64	0.969	0.903

Table 4: Quantitative evaluation results of the proposed WAN on color images with 1 bit of capacity.

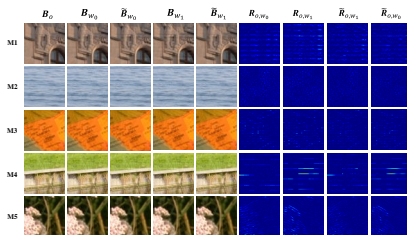


Figure 7: Qualitative evaluation results of the proposed WAN on color images with 1 bit of capacity.

4.6 Evaluation Results on Color Images

In this subsection, we present quantitative and qualitative results of the WAN on color images (see Table 4 and Fig. 7). To this end, we exploited the color image dataset to generate watermarked color images for training, validation, and test of WAN [9, 10, 26]. In detail, after converting the RGB domain to the YCbCr domain, watermarks are inserted into the Y-channel of a given image for each MW method. In non-attack situation, each MW method has a low BER value (of 0.048 or less), while the average BER value increases dramatically to 0.903 after WAN is applied. For average PSNR and SSIM, only minor degradations of 2.51 dB and 0.006, respectively, are shown after WAN-based attack. Furthermore, as shown in attacked and residual images in Fig. 7, it can be observed that WAN hardly causes visual degradation. It is expected that the performance difference from the results on gray-scale image is caused because the WAN should target the watermark signals inserted in a specific channel from the given RGB image.

4.7 Application of WAN

Watermarking designers need to adjust their watermarking methods according to the requirements of content characteristics and distribution environments. Professional photographers may prefer the invisibility of embedded watermarks for robustness against various signal processing attacks. On the other hand, confidential documents have to be very robust against not only general processing but also malicious editing, so visual degradation to some extent may be acceptable. We present Add-on Watermarking (AoW), one application of WAN, that adjusts between the imperceptibility and robustness of pre-defined watermarking methods. We introduce AoW-min and AoW-max, which are defined as follows:

WM method	Watermarked			AoW-max			AoW-min		
	PSNR	SSIM	BER	PSNR	SSIM	BER	PSNR	SSIM	BER
M1	35.55	0.938	0.026	31.67	0.914	0.020	40.01	0.979	0.085
M2	41.86	0.988	0	36.70	0.973	0	43.49	0.992	0
M3	36.59	0.974	0	32.35	0.954	0	38.26	0.980	0
M4	38.98	0.986	0.002	36.01	0.978	0.002	41.87	0.991	0.009
M5	39.21	0.987	0.013	35.18	0.973	0.018	41.97	0.993	0.057
Average	38.43	0.974	0.008	34.38	0.958	0.008	41.12	0.987	0.030

Table 5: Quantitative evaluation results of the proposed AoW on fidelity and robustness.

$$B_{AoW-min}^{i,j} = \begin{cases} B_o^i + R_{o,w_j}^i & \text{if } |R_{o,w_j}^i| < |\tilde{R}_{o,w_{1-j}}^i| \\ B_o^i + \tilde{R}_{o,w_{1-j}}^i & \text{if } |R_{o,w_j}^i| \geq |\tilde{R}_{o,w_{1-j}}^i| \end{cases},$$

WM method	Watermarked			AoW-max			AoW-min		
	JPEG	MB	NA	JPEG	MB	NA	JPEG	MB	NA
M1	0.115	0.133	0.114	0.066	0.096	0.076	0.197	0.255	0.231
M2	0	0.174	0.003	0	0.148	0	0.002	0.183	0.013
M3	0.004	0.316	0.014	0.003	0.312	0.004	0.008	0.319	0.010
M4	0.024	0.324	0.007	0.021	0.302	0.004	0.061	0.355	0.033
M5	0.069	0.220	0.090	0.065	0.210	0.087	0.142	0.294	0.158
Average	0.042	0.233	0.045	0.031	0.213	0.034	0.082	0.281	0.089

Table 6: Comparison of robustness of the AoW against signal processing attacks.

$$B_{AoW-max}^{i,j} = \begin{cases} B_o^i + \tilde{R}_{o,w_{1-j}}^i & \text{if } |R_{o,w_j}^i| < |\tilde{R}_{o,w_{1-j}}^i| \\ B_o^i + R_{o,w_j}^i & \text{if } |R_{o,w_j}^i| \geq |\tilde{R}_{o,w_{1-j}}^i| \end{cases} \quad (2)$$

where i and j denote pixel location and bit information. AoW-min and AoW-max embed the watermark by adding R_{o,w_j}^i or $\tilde{R}_{o,w_{1-j}}^i$ selectively. AoW-min selects the residual with the minimum absolute value, and AoW-max selects with the maximum absolute value to get additional imperceptibility and robustness, respectively.

Table 5 reports the PSNR, SSIM, and BER of the original MW methods and their AoW-max and AoW-min versions. First, AoW-min achieves 41.12 dB in average PSNR, which is improved by 2.69 dB compared to the original MW methods, but the robustness of MW methods is slightly sacrificed. As shown in Fig. 8, the signal intensity of residual images caused by AoW-min is relatively small compared to the others. Through these, it is observed that AoW-min helps to improve the invisibility of watermarking. Next, to evaluate the advantage of AoW-max, the robustness evaluation against signal processing attacks (i.e., JPEG, median blur (MB), and noise addition (NA)) of StirMark is performed (see Table 6). Here, parameters of JPEG, MB, and NA are set to [60, 70, 80], [2, 3, 4], and [1, 2, 3], respectively. Although the average PSNR of AoW-max is 34.38 dB, which is degraded compared to original MW methods, the robustness against signal processing attacks is improved in terms of BER. Furthermore, we test the robustness against geometric distortion in Table 7. **M5** is tested only because it is designed to be robust to geometric attacks. We can find that AoW-max enhances the capability of **M5** to withstand three types of geometric attacks. We would like to note that the proposed AoW is applicable to assist in improving imperceptibility or robustness of MW methods, so watermarking designers can choose between AoW-max and AoW-min before redesigning watermarking systems to meet new requirements.

M5	Rotation	Center crop	Rescaling
Watermarked	0.106	0.055	0.047
AoW-max	0.092	0.046	0.038
AoW-min	0.185	0.086	0.093

† Rotation para.: [3, 4, 5], Center crop para.: [85, 90, 95]

† Rescaling para.: [80, 90, 110, 120]

Table 7: Robustness evaluation of **M5** with AoW against geometric distortions.

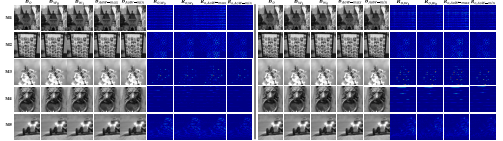


Figure 8: Qualitative evaluation results of AoW-min and AoW-max.

5 Conclusion

In this paper, we propose a novel CNN-based benchmark tool for block-based MW methods that exploits vulnerability of the targeted watermarking method and attacks watermarked images to mislead the watermarking extractor with minimal visual degradation. To achieve this goal, we design customized losses of a watermarking attack loss for abnormal bit extraction and a content loss to maintain visual quality. Through quantitative and qualitative experiments with a variety of MW methods, we demonstrate that the WAN performs more effective attacks than existing benchmark tools in terms of maintaining visual quality and interfering with watermark extraction. Furthermore, we show our WAN can be an add-on module for existing MW methods to get additional performance gains.

Acknowledgements

This research was developed with supporting from NAVER WEBTOON Corp. The authors would like to highly acknowledge colleagues from Webtoon AI of NAVER WEBTOON Corp. for their insightful advice for this research. We particularly thank Prof. Heung-Kyu Lee, Choong-Hyun Seo, and Ji-Hyeon Kang for their support.

References

- [1] Nasir Ahmed, T Natarajan, and Kamisetty R Rao. Discrete cosine transform. *IEEE transactions on Computers*, 100(1):90–93, 1974.
- [2] Woogeun Bae, Seung-Hun Nam, In-Jae Yu, Myung-Joon Kwon, Minseok Yoon, and Heung-Kyu Lee. Dual-path convolutional neural network for classifying fine-grained manipulations in h. 264 videos. *Multimedia Tools and Applications*, pages 1–28, 2021.
- [3] Mauro Barni, Franco Bartolini, Vito Cappellini, and Alessandro Piva. A dct-domain system for robust image watermarking. *Signal processing*, 66(3):357–372, 1998.
- [4] Patrick Bas and Teddy Furon. Bows-2, 2007.
- [5] Patrick Bas, Tomáš Filler, and Tomáš Pevný. Break our steganographic system: the ins and outs of organizing a boss. In *International workshop on information hiding*, pages 59–70. Springer, 2011.
- [6] Ingemar Cox, Matthew Miller, Jeffrey Bloom, Jessica Fridrich, and Ton Kalker. *Digital watermarking and steganography*. Morgan kaufmann, 2007.
- [7] Ingemar J Cox, Joe Kilian, F Thomson Leighton, and Talal Shamon. Secure spread spectrum watermarking for multimedia. *IEEE transactions on image processing*, 6(12): 1673–1687, 1997.
- [8] Arthur L Da Cunha, Jianping Zhou, and Minh N Do. The nonsubsampling contourlet transform: theory, design, and applications. *IEEE transactions on image processing*, 15(10):3089–3101, 2006.
- [9] Duc-Tien Dang-Nguyen, Cecilia Pasquini, Valentina Conotter, and Giulia Boato. Raise: A raw images dataset for digital image forensics. In *Proceedings of the 6th ACM multimedia systems conference*, pages 219–224, 2015.
- [10] Bart De Moor and Paul Van Dooren. Generalizations of the singular value and qr-decompositions. *SIAM journal on matrix analysis and applications*, 13(4):993–1014, 1992.
- [11] Thomas Gloe and Rainer Böhme. The ‘dresden image database’ for benchmarking digital image forensics. In *Proceedings of the 2010 ACM Symposium on Applied Computing*, pages 1584–1590, 2010.
- [12] Gene H Golub and Christian Reinsch. Singular value decomposition and least squares solutions. In *Linear Algebra*, pages 134–151. Springer, 1971.

- [13] Auguste Kerckhoffs. *La cryptographie militaire, ou, Des chiffres usités en temps de guerre: avec un nouveau procédé de déchiffrement applicable aux systèmes à double clef*. Librairie militaire de L. Baudoin, 1883.
- [14] W-H Kim, S-H Nam, and H-K Lee. Blind curvelet watermarking method for high-quality images. *Electronics Letters*, 53(19):1302–1304, 2017.
- [15] Wook-Hyung Kim, Jong-Uk Hou, Han-Ul Jang, and Heung-Kyu Lee. Robust template-based watermarking for dibr 3d images. *Applied Sciences*, 8(6):911, 2018.
- [16] Wook-Hyung Kim, Seung-Hun Nam, Ji-Hyeon Kang, and Heung-Kyu Lee. Robust watermarking in curvelet domain for preserving cleanness of high-quality images. *Multimedia Tools and Applications*, 78(12):16887–16906, 2019.
- [17] Deepa Kundur and Dimitrios Hatzinakos. Digital watermarking using multiresolution wavelet decomposition. In *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*, volume 5, pages 2969–2972. IEEE, 1998.
- [18] Yu-Hsun Lin and Ja-Ling Wu. A digital blind watermarking for depth-image-based rendering 3d images. *IEEE transactions on Broadcasting*, 57(2):602–611, 2011.
- [19] Nasrin M Makbol, Bee Ee Khoo, and Taha H Rassem. Block-based discrete wavelet transform-singular value decomposition image watermarking scheme using human visual system characteristics. *IET Image processing*, 10(1):34–52, 2016.
- [20] Henrique S Malvar and Dinei AF Florêncio. Improved spread spectrum: A new modulation technique for robust watermarking. *IEEE transactions on signal processing*, 51(4):898–905, 2003.
- [21] Seung-Min Mun, Seung-Hun Nam, Han-Ul Jang, Dongkyu Kim, and Heung-Kyu Lee. A robust blind watermarking using convolutional neural network. *arXiv preprint arXiv:1704.03248*, 2017.
- [22] Seung-Min Mun, Seung-Hun Nam, Haneol Jang, Dongkyu Kim, and Heung-Kyu Lee. Finding robust domain from attacks: A learning framework for blind watermarking. *Neurocomputing*, 337:191–202, 2019.
- [23] Seung-Hun Nam, Wook-Hyoung Kim, Seung-Min Mun, Jong-Uk Hou, Sunghee Choi, and Heung-Kyu Lee. A sift features based blind watermarking for dibr 3d images. *Multimedia Tools and Applications*, 77(7):7811–7850, 2018.
- [24] Seung-Hun Nam, Seung-Min Mun, Wonhyuk Ahn, Dongkyu Kim, In-Jae Yu, Wook-Hyung Kim, and Heung-Kyu Lee. Nscf-based robust and perceptual watermarking for dibr 3d images. *IEEE Access*, 8:93760–93781, 2020.
- [25] Jin-Seok Park, Hyeon-Gi Kim, Do-Guk Kim, In-Jae Yu, and Heung-Kyu Lee. Paired mini-batch training: A new deep network training for image forensics and steganalysis. *Signal Processing: Image Communication*, 67:132–139, 2018.

- [26] Jinseok Park, Donghyeon Cho, Wonhyuk Ahn, and Heung-Kyu Lee. Double jpeg detection in mixed jpeg quality factors using deep convolutional neural network. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 636–652, 2018.
- [27] Shelby Pereira, Sviatoslav Voloshynovskiy, Maribel Madueno, Stéphan Marchand-Maillet, and Thierry Pun. Second generation benchmarking and application oriented evaluation. In *International workshop on information hiding*, pages 340–353. Springer, 2001.
- [28] Luis Pérez-Freire and Fernando Pérez-González. Spread-spectrum watermarking security. *IEEE Transactions on Information Forensics and Security*, 4(1):2–24, 2009.
- [29] Fabien AP Petitcolas. Watermarking schemes evaluation. *IEEE signal processing magazine*, 17(5):58–64, 2000.
- [30] Fabien AP Petitcolas, Ross J Anderson, and Markus G Kuhn. Attacks on copyright marking systems. In *International workshop on information hiding*, pages 218–238. Springer, 1998.
- [31] Mark J Shensa. The discrete wavelet transform: wedding the a trous and mallat algorithms. *IEEE Transactions on signal processing*, 40(10):2464–2482, 1992.
- [32] Jiawei Su, Danilo Vasconcellos Vargas, and Kouichi Sakurai. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, 23(5):828–841, 2019.
- [33] Qingtang Su, Gang Wang, Xiaofeng Zhang, Gaohuan Lv, and Beijing Chen. An improved color image watermarking algorithm based on qr decomposition. *Multimedia Tools and Applications*, 76(1):707–729, 2017.
- [34] Anastasios Tefas, Nikos Nikolaidis, and Ioannis Pitas. Watermarking techniques for image authentication and copyright protection. In *Handbook of Image and Video Processing*, pages 1083–1109. Elsevier, 2005.
- [35] Sviatoslav Voloshynovskiy, Shelby Pereira, Victor Iquise, and Thierry Pun. Attack modelling: towards a second generation watermarking benchmark. *Signal processing*, 81(6):1177–1214, 2001.
- [36] Shih-Hao Wang and Yuan-Pei Lin. Wavelet tree quantization for copyright protection watermarking. *IEEE Transactions on Image Processing*, 13(2):154–165, 2004.
- [37] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.
- [38] Minseok Yoon, Seung-Hun Nam, In-Jae Yu, Wonhyuk Ahn, Myung-Joon Kwon, and Heung-Kyu Lee. Frame-rate up-conversion detection based on convolutional neural network for learning spatiotemporal features. *arXiv preprint arXiv:2103.13674*, 2021.
- [39] I-J Yu, Wonhyuk Ahn, S-H Nam, and H-K Lee. Bitmix: data augmentation for image steganalysis. *Electronics Letters*, 56(24):1311–1314, 2020.

- [40] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2472–2481, 2018.
- [41] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.
- [42] Hang Zhao, Orazio Gallo, Iuri Frosio, and Jan Kautz. Loss functions for image restoration with neural networks. *IEEE Transactions on computational imaging*, 3(1):47–57, 2016.