JCTA Journal of Computing Theories and Applications ISSN:3024-9104

Research Article

NAA-GAN: Noise-Aware Cross-Attention for Robust Blind Image Watermarking

Surni Erniwati [1,2,*], Jufriadif Na'am [1], and Ferda Ernawan [1,3]

[1] Informatics Doctoral Program, Universitas Nusa Mandiri, Jakarta 13620, Indonesia; e-mail: 16240023@nusamandiri.ac.id; jufriadifnaam@nusamandiri.ac.id
[2] Department of Informatics Engineering, Universitas Teknologi Mataram, Mataram 83116, Indonesia; e-mail : mentari1990@gmail.com
[3] Computing Program, Universiti Malaysia Pahang Al-Sultan Abdullah, Pekan, Pahang 26600, Malaysia; e-mail : ferda@umpsa.edu.my
* Corresponding Author : Surni Erniwati

Abstract: Robust blind image watermarking remains challenging because embedded watermark information must remain recoverable after image processing while preserving the visual quality of the watermarked image. This paper proposes NAA-GAN, a generative adversarial watermarking framework that conditions the encoder on a six-dimensional descriptor of the anticipated distortion category. The encoder integrates dense feature aggregation, noise-aware cross-attention, and residual embedding to insert a 30-bit watermark. During training, a noise subnetwork simulates image processing attacks, while the decoder recovers the watermark from the received image without requiring the distortion descriptor. Experiments on COCO, PASCAL VOC2012, ImageNet, and EUVP assess image fidelity using peak signal-to-noise ratio (PSNR) and structural similarity index measure (SSIM), and robustness using bit accuracy (BA). NAA-GAN achieves PSNR values ranging from 36.96 to 37.79 dB and average BAs above 98% across the four datasets under the five evaluated attacks. On COCO, it achieves 99.02% BA under JPEG compression and an average BA of 92.59% across distortions not encountered during training. An ablation study shows that removing the distortion descriptor reduces COCO PSNR from 37.38 to 36.73 dB and average BA from 98.07% to 96.14%. Across all four datasets, NAA-GAN achieves higher PSNR and JPEG BA than the compared models. However, ARWGAN-CN achieves higher SSIM and average BA, indicating that the proposed method improves pixel-level fidelity and JPEG robustness but does not consistently outperform competing methods across all evaluation metrics.

Keywords: Blind image watermarking; Copyright protection; Cross-attention; Deep learning; Distortion-conditioned embedding; Generative adversarial network; Noise-aware attention; Robust watermarking.

Received: August, 11th 2026
Revised: September, 30th 2026
Accepted: October, 2nd 2026
Published: October, 25th 2026

Copyright: © 2026 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) licenses (https://creativecommons.org/licenses/by/4.0/)

1. Introduction

Digital watermarking remains an important research direction in the field of information security, despite having been studied for several decades. The relevance of this topic persists due to its wide range of applications, particularly in copyright protection, ownership tracking, source identification, and content integrity authentication. In particular, image watermarking has become one of the most extensively studied areas because digital images are easy to generate, distribute, and consume through various modern multimedia platforms [1]–[4].

Image watermarking techniques are generally classified into two types: fragile watermarking and robust watermarking [5], [6]. Fragile watermarking focuses on detecting image manipulation and verifying content integrity. In contrast, robust watermarking emphasizes the ability to preserve and re-extract the embedded watermark even when the image is subjected to various processing attacks, making it suitable for copyright protection, ownership verification, and source tracking [7], [8]. Traditional robust watermarking approaches generally embed watermark information into transform-domain coefficients to improve their

ability to withstand various image processing attacks [9], [10]. Ko et al. [11] introduced a robust and blind digital image watermarking method that utilizes the Discrete Cosine Transform (DCT) domain based on inter-block correlation. Hu et al. [12] introduced a blind watermarking algorithm for color images based on Singular Value Decomposition (SVD), enhanced by integrating mixed modulation (MM) to achieve an optimal balance between imperceptibility and robustness. Zhou et al. [13] applied Support Vector Regression parameter calibration to support model transfer across spectrometers, whereas Yang et al. [14] developed a blind image quality measurement method based on data-driven transform-feature enhancement. However, conventional watermarking approaches still have limitations because they depend heavily on the designer's expertise and involve complex processing pipelines. In addition, the watermark embedding locations and strategies are typically determined based on manually extracted image features; therefore, inaccurate feature extraction may lead to a degradation in watermarking performance.

In recent years, deep learning has achieved significant progress in various computer vision tasks, including image classification and segmentation, due to its strong capability to automatically learn visual feature representations from data [15]. Compared with conventional machine learning, deep learning is capable of automatically extracting more complex image features and provides better generalization through training on large-scale datasets. Therefore, deep learning has been increasingly applied to image watermarking to improve both the watermark embedding and extraction processes. In general, existing deep network-based image watermarking methods can be classified into two main approaches: convolutional neural network (CNN)-based methods and generative adversarial network (GAN)-based methods [16]. Ahmadi et al. introduced a blind watermarking framework based on Fully Convolutional Networks (FCNs), which is trained end-to-end to perform real-time information embedding and extraction. In addition, Luo et al. [17] proposed a distortion-agnostic deep watermarking framework that does not rely on explicit modeling of image distortions during training. The robustness of this method is achieved through a combination of adversarial training and channel coding, enabling it to achieve competitive performance under known distortions while providing better generalization to distortions unseen during training.

In addition to CNNs, GANs have characteristics that are well suited for robust image watermarking, as the adversarial learning process between the generator and the discriminator can help balance embedding capacity, imperceptibility, and robustness [18]. In line with this, several recent studies have developed GAN-based image watermarking approaches to improve the visual quality of watermarked images while maintaining watermark robustness against various attacks. Zhu et al. [19] proposed HiDDeN as one of the early deep learning-based robust image watermarking frameworks. This model adopts an encoder–noise–decoder structure and leverages adversarial learning to maintain a balance between the visual quality of watermarked images and the capability of watermark extraction. The encoder generates a watermarked image that visually resembles the original image, while the decoder is trained to recover the watermark even when the image is subjected to distortion. Huang et al. [1] introduced ARWGAN, a GAN-based robust image watermarking model guided by an attention mechanism. This approach is designed to improve the quality of watermark embedding by selecting more representative image features or regions, allowing the watermark to remain imperceptible while still being extractable after the image undergoes various processing attacks. Hao et al. [20] also developed a GAN-based robust image watermarking method consisting of a generator and an adversary. The generator is used to embed and extract the watermark from distorted images, while the adversary is responsible for detecting the presence of the watermark and applying noise-based distortions. This approach is an extension of HiDDeN with the addition of a high-pass filter before the discriminator, thereby directing the watermark to be embedded more strongly in the mid-frequency components of the image. Jia et al. [21] proposed the Mini-Batch of Real and Simulated JPEG Compression (MBRS) framework to improve the robustness of image watermarking methods against JPEG compression. This approach utilizes a combination of real JPEG compression and simulated JPEG noise during the training process. By applying these two types of distortions alternately, the model can learn to handle JPEG degradation more effectively under real-world conditions.

Recent research also investigates watermarking through different representations and application objectives. The AI-assisted NEQR framework employs reversible quantum-circuit simulation and attack classification. The hybrid fragile watermarking framework addresses tamper detection, localization, and dual self-recovery.

Although these watermarking models provide a strong foundation, robustness against various trained and unseen distortions remains a challenge. Model robustness may vary across clean conditions, single trained attacks, composed attacks, and attacks that are not included during training. Therefore, this paper investigates whether the refinement of noise-aware attention can improve the balance between imperceptibility and robustness in GAN-based watermarking models, particularly under trained-noise conditions and selected unseen distortions.

NAA-GAN builds on established components, including adversarial learning, dense feature aggregation, residual embedding, and multi-head attention. Its methodological contribution centers on incorporating an explicit distortion descriptor into the encoder's feature modulation pathway. In the proposed design, a six-dimensional categorical descriptor is projected into a latent representation and supplied to a standard cross-attention module. The resulting representation is transformed into a mask that modulates message-aware features before residual watermark reconstruction. Relative to the reference ARWGAN architecture, the distinguishing design choice is the introduction of descriptor-driven conditioning during watermark insertion. The decoder remains independent of this descriptor and recovers the watermark using only the received image. The main contributions of this study are as follows.

1. We propose a distortion-conditioned blind image watermarking framework in which a six-class attack descriptor modulates message-aware image features during embedding.
2. We incorporate a descriptor-driven modulation pathway into the watermark encoder while preserving descriptor-independent decoding.
3. We evaluate the model under trained and unseen distortions, including compound distortion conditions, and assess cross-dataset generalization on multiple public datasets, in addition to the domain evaluation.

The remainder of this paper is organized as follows. Section 2 reviews related work on robust image watermarking, deep learning-based watermarking models, and attention mechanisms. Section 3 presents the proposed NAA-GAN architecture and describes its main components, including the encoder, noise subnetwork, decoder, and discriminator. Section 4 describes the experimental setup and discusses the results, including comparisons with existing watermarking methods across multiple datasets and an ablation study of the distortion descriptor. Finally, Section 5 concludes the paper and outlines directions for future work.

2. Related Work

Robust image watermarking was initially developed extensively in the transform domain by exploiting the characteristics of image coefficients to balance capacity, imperceptibility, and robustness. Several classical approaches embed the watermark in the DCT domain, SVD domain, or combinations of other transforms to improve resistance against image-processing attacks [11], [12]. In addition, machine-learning-based techniques have also been employed to improve watermarking parameter estimation and geometric distortion correction, for example through SVR and SVM [13], [14], [22]. Although effective in certain scenarios, these approaches generally still rely on manual feature design and relatively complex pre-processing or post-processing pipelines.

With the advancement of deep learning, image watermarking has gradually shifted toward end-to-end models that are capable of automatically learning feature representations. Within the group of CNN-based watermarking methods, Ahmadi et al. [23] introduced ReDMark, an end-to-end residual diffusion watermarking framework that can learn watermarking in a desired transform space and model various attacks as differentiable layers. Luo et al. [17] also proposed distortion-agnostic deep watermarking (DA), which achieves robustness through adversarial training and channel coding without explicitly modeling each distortion during training. Nevertheless, most CNN-based methods still face limitations in learning features that are sufficiently rich and robust under various attack conditions.

In addition to CNNs, GAN-based approaches have also become an important direction in robust watermarking because the adversarial nature between the generator and the discriminator naturally supports the trade-off between invisibility and robustness [18]. Zhu et al. [19] introduced HiDDeN as one of the pioneering GAN-based robust watermarking methods, employing an encoder-noise-decoder structure trained in an end-to-end manner. Hao et al. [20] developed a similar approach by adding a high-pass filter before the discriminator, encouraging the watermark to be embedded in mid-frequency regions. Subsequently, Jia et al.

[21] introduced MBRS to improve robustness against JPEG compression through a combination of real and simulated JPEG noise, while Fernandez et al. [24] proposed SSLW, which utilizes a self-supervised encoder to embed the watermark in the latent space. Although these models have improved watermarking performance, many of them are still not fully capable of adaptively extracting features that are sufficiently rich, robust, and relevant for the embedding process.

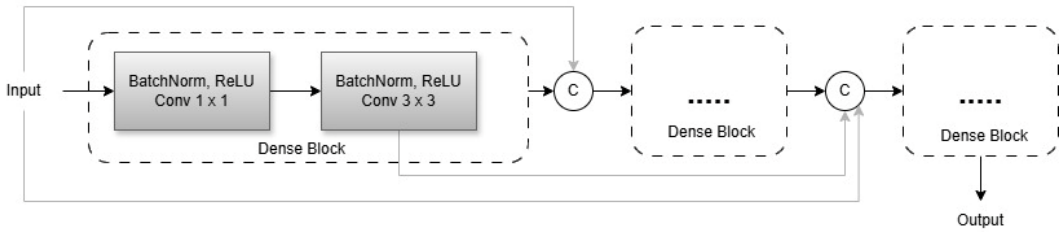


Figure 1. Structure of DenseNet

The architecture has also been strongly influenced by DenseNet and attention mechanisms. DenseNet utilizes dense connections to reuse features from preceding layers, thereby helping to alleviate the vanishing gradient problem, improve training efficiency, and enrich feature representations [25], [26]. Meanwhile, attention mechanisms have been proven effective in various computer vision tasks, including image classification, object detection, and tamper detection [27], [28]. In the context of watermarking, attention has been used to identify inconspicuous regions to improve invisibility [29], as well as to locate areas that are more robust against distortions [30], [31]. However, these approaches generally tend to emphasize only one aspect, either invisibility or robustness, and have not yet optimally improved both aspects simultaneously.

3. Proposed Watermarking Network Model

This study proposes a noise-aware robust image watermarking network built on a generative adversarial learning framework. The main objective of the proposed model is to simultaneously preserve the visual quality of the cover image and improve the robustness of watermark recovery under various distortions. To achieve this, the network integrates four main components, namely an encoder, a noiser, a decoder, and a discriminator. The encoder embeds a binary watermark message into the cover image, the noiser simulates transmission or post-processing attacks, the decoder reconstructs the hidden message from the distorted watermarked image, and the discriminator encourages the embedded image to remain visually similar to the original one.

3.1. Overview

The proposed architecture comprises an encoder-noiser-decoder pipeline coupled with an adversarial discriminator. The encoder receives the cover image, the binary watermark message, and the six-dimensional noise descriptor, and computes a residual watermark pattern that is added to the cover image to produce the watermarked image. The configurable noiser then randomly selects and applies one operation from the configured set, including an identity operation that leaves the image unchanged. The decoder estimates the embedded watermark message from the resulting image. During training, the discriminator learns to distinguish cover images from watermarked images before distortion, while the encoder is optimized to make watermarked images resemble cover images. The discriminator and encoder–decoder parameters are updated alternately. The adversarial objective, together with image fidelity losses based on MSE and SSIM, encourages visually unobtrusive embedding, while the message reconstruction loss promotes watermark recovery.

A robust watermarking model should satisfy two main properties: 1) different image attacks applied to the encoded image should have minimal impact on the embedded watermark, and 2) the watermark should be embedded by introducing imperceptible perturbations into the original image. To maintain these two properties, robust features need to be extracted and fused with the watermark, while the watermark should also be distributed in inconspicuous regions to preserve visual imperceptibility.

Fig. 2 illustrates the architecture of NAA-GAN, which consists of an encoder, a noiser, a decoder, and a discriminator. During the training iterations, these components are continuously updated to achieve high watermarking invisibility and strong robustness.

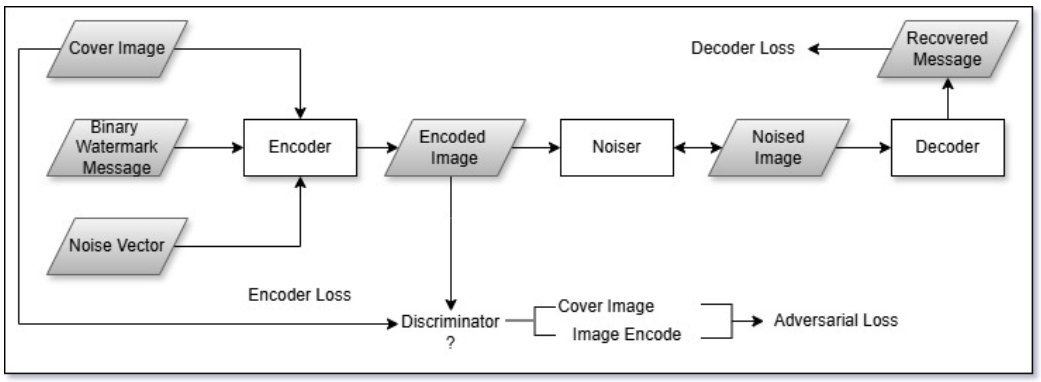


Figure 2. Architecture of NAA-GAN

Given a cover image $I \in [-1,1]^{3\times H\times W}$ and a binary watermark message $m \in \{0,1\}^L$, with L=30 bits and $n \in \mathbb{R}^6$ denote a distortion descriptor. The overall process of the proposed network can be formulated as

$$\begin{aligned} I_w &= E(I, m, n), \\ I_a &= N(I_w, I), \\ \hat{m} &= D(I_a), \end{aligned} \tag{1}$$

where $E(\cdot)$ is the encoder, $N(\cdot)$ is the noiser, $D(\cdot)$ is the decoder, I_w is the watermarked image, I_a is the attacked image, and $\hat{m}$ is the recovered watermark message. In contrast to conventional watermarking models that embed the message uniformly, the proposed method introduces a noise-aware attention mechanism so that the embedding process can adapt to the expected distortion characteristics

3.2. Encoder

The encoder is designed to embed the watermark message by combining image-dependent feature extraction with distortion-conditioned feature modulation. The cover image is first processed by convolutional layers to extract shallow visual features. Meanwhile, the binary watermark message is spatially expanded and concatenated with intermediate feature maps. Several dense bottleneck blocks are then employed to promote feature reuse, stabilize gradient flow, and improve the expressive capacity of the embedding network.

The six-dimensional distortion descriptor is introduced to provide the encoder with explicit information about the anticipated distortion category when constructing the watermark embedding. Its dimensionality follows the five distortion types considered in the training configuration (dropout, Gaussian blur, JPEG compression, resizing, and random cropping) plus an identity condition. These distortion types represent different degradation mechanisms relevant to watermark recovery, including dispersed pixel replacement, local smoothing, lossy compression, spatial resampling, and localized content removal. The identity condition explicitly represents the absence of an anticipated distortion, allowing the descriptor to distinguish clean conditions from the five distortion categories. A six-dimensional one-hot representation therefore provides a separate indicator for each configured condition without imposing an ordinal relationship between distortion types. The descriptor encodes the distortion category, while severity is controlled separately through the parameters of the corresponding noise operator. By projecting this categorical information into the encoder's latent feature space, the proposed design provides a conditioning signal for modulating message-aware embedding features according to the expected degradation mechanism.

During evaluation, the distortion to be simulated after watermark embedding is specified as the test condition. Before embedding, the corresponding distortion descriptor is generated automatically from this configured attack condition and represented as a six-dimensional one-hot vector corresponding to identity, dropout, Gaussian blur, JPEG compression, resizing,

or cropping. The descriptor is then supplied to the encoder during watermark embedding; thus, it is not independently selected by the evaluator and is not inferred from the received image. The descriptor represents only the distortion category and does not encode its severity or parameter values. For distortions that cannot be mapped to any of the six predefined categories, the identity descriptor is used, meaning that the encoder receives no explicit category-specific information about these unseen distortions. For combined distortion conditions, the current descriptor representation remains single-label and does not explicitly encode multiple constituent distortion categories simultaneously. The decoder operates independently of the descriptor and reconstructs the embedded message directly from the received image.

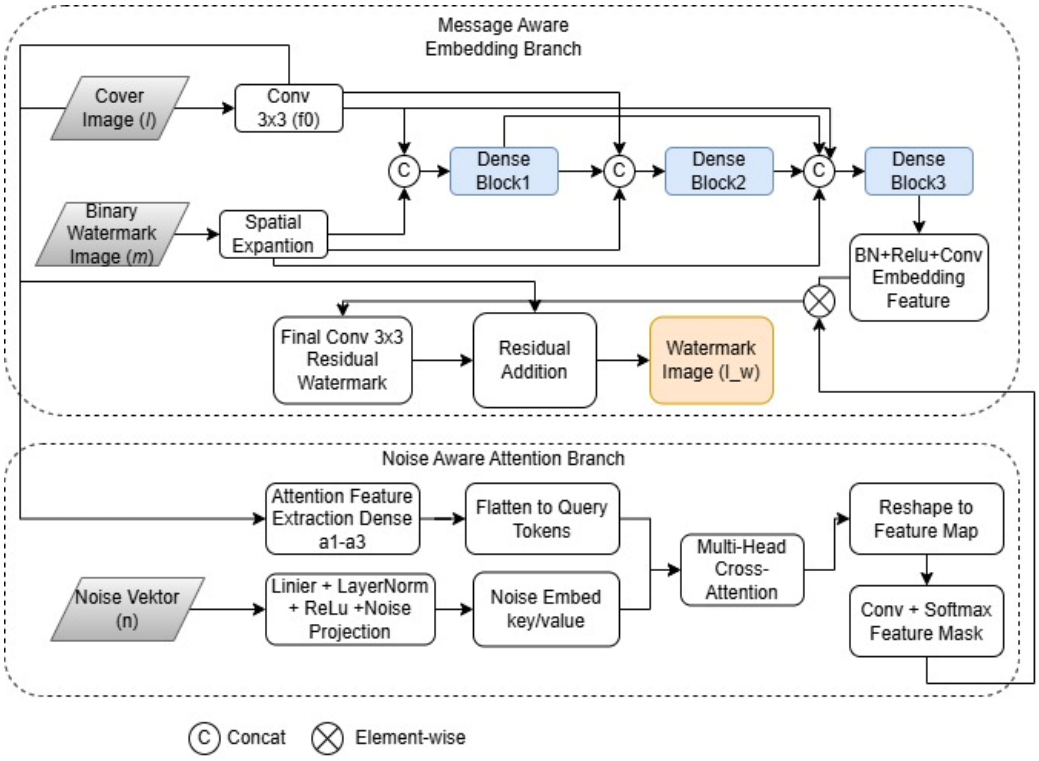


Figure 3. Structure of encoder

As illustrated in Fig. 3, the encoder first extracts shallow visual features from the input image through an initial convolutional layer, producing a feature representation f_0. In parallel, the binary watermark message is expanded from a one-dimensional vector into a spatial tensor with the same height and width as the input image. This spatial expansion enables each watermark bit to interact with image features across the entire embedding region rather than being treated only as a global descriptor. The expanded message tensor is then concatenated with the image feature maps and progressively processed by a series of dense bottleneck blocks. This design promotes feature reuse, improves gradient propagation, and enhances the expressive power of the embedding branch.

The main embedding branch generates message-aware features by repeatedly combining the shallow image feature f_0, the expanded message representation, and intermediate dense features. This iterative fusion allows the encoder to jointly model local image structure and watermark semantics. The resulting feature representation is subsequently refined by additional convolutional layers to produce a compact embedding feature map F_e which serves as the primary carrier of the hidden message. In this way, the watermark is not inserted directly into raw pixels but rather encoded within a learned latent representation that is optimized for both visual fidelity and recoverability.

To incorporate distortion information, the encoder introduces a noise-aware attention branch. The distortion descriptor is first projected into the latent feature space through a linear transformation followed by layer normalization and nonlinear activation, yielding a noise-aware embedding. Meanwhile, image-derived features are extracted from the shallow visual representation and reshaped into query tokens. A multi-head cross-attention operation

is then performed using the image tokens as queries and the projected distortion descriptor as a single key–value token. Since only one key–value token is used, the attention weights are constant across spatial positions and do not perform spatially selective attention. Instead, the module provides a descriptor-dependent conditioning signal that is subsequently transformed into a feature mask and applied to the message-aware embedding features.

The output of the cross-attention module is reshaped back into a feature map and transformed into a feature mask that modulates the message-aware embedding features. Let F_a denote the attention-guided feature map and M_a the resulting mask. The final distortion-aware embedding feature is obtained by element-wise modulation.

$$M_a = \mathcal{G}(F_a), \qquad \hat{F} = F_e \odot M_a \tag{2}$$

where $\odot$ denotes element-wise multiplication. This operation applies distortion-dependent modulation to the message-aware embedding features through the feature mask. Because the cross-attention module uses a single key–value token, it does not independently determine spatially varying attention weights or explicitly select image regions based on their resistance to distortion. Instead, the descriptor provides a conditioning signal, while spatial variations in the final embedding representation arise from the image-dependent features processed by the convolutional embedding branch. Thus, the proposed mechanism is more appropriately characterized as distortion-conditioned feature modulation rather than spatially selective cross-attention.

Finally, the modulated feature map is passed through a convolutional reconstruction layer to generate a residual watermark component R. Instead of synthesizing a completely new image, the proposed encoder adopts a residual embedding strategy, and the final watermarked image is obtained as

$$I_w = I + R \tag{3}$$

This residual formulation is beneficial for preserving the perceptual appearance of the cover image, since the network only learns the minimal modification necessary for reliable watermark embedding. Therefore, the proposed noise-aware encoder simultaneously addresses two key requirements of robust watermarking: imperceptibility, by limiting visible distortion through residual embedding, and robustness, by adapting the embedding pattern to distortion characteristics through noise-guided cross-attention.

3.3. Noise Subnetwork

In general, watermarks can be extracted more effectively if the watermark information is embedded in parts of the image or features that are relatively resistant to distortion. Therefore, various types of disturbances can be simulated during the training process to encourage the network to learn robust watermark embedding patterns. In this approach, the noise subnetwork N is used to represent various types of image attacks as part of the training process. Since the noise transformation is integrated into the network workflow, the effect of distortion on the watermark extraction process can be taken into account when the network parameters are updated via backpropagation.

In the noise subnetwork, the input consists of images that have been watermarked I_{w} and the original image I. At each training iteration t, one of the noise operators N_t is selected at random and applied to a single batch. The process of generating images that have been distorted I_{a} can be expressed as

$$I_a = N_{\mathrm{net}}(I_w, I; N_t) \tag{4}$$

with N_{net} defines the noise subnetwork function, whereas N_t is the disturbance operator selected in the t-th training iteration. Semicolon notation in N_t is used to emphasize that N_t shows the selection of transformation operators, not the network's input image. The operators are selected at random, and the selected operators are then applied to the data in the batch currently being processed.

The noise subnetwork consists of several types of transformations, namely dropout, Gaussian blur, JPEG compression, resize, and random crop. Each transformation simulates a different type of distortion so that the network experiences varying noise conditions during

the training process. From among these operators, one is selected at random at each training iteration to generate I_{a}.

Unlike some other transformations, the dropout operation explicitly requires I_{w} and I. In this operation, a portion of the pixels from the watermarked image is retained based on a binary mask M, while the pixels that are not retained are replaced by pixels at the same positions in the original image. This operation is formulated as

$$I_a = M \odot I_w + (1 - M) \odot I \tag{5}$$

with M is a binary mask and $\odot$ shows element-by-element multiplication. When $M = 1$, pixels come from I_{w}, whereas when $M = 0$, those pixels are replaced by the corresponding values from I. This explains why the original image is still provided as one of the inputs to the noise subnetwork.

In the Gaussian blur transformation, the watermarked image is blurred using a Gaussian filter with a convolution kernel of size 3×3, while the level of blurring is controlled by the variance parameter δ. The JPEG operator is used to simulate degradation caused by compression with a quality factor Q. Next, the resize operation simulates a change in image size based on a scale factor R, whereas random crop simulates the loss of part of the image area due to the cropping process. Selecting one of these operators at random during training aims to introduce variation in distortion so that the model can learn to embed and extract watermarks that are more resistant to various image manipulations.

3.4. Decoder

The watermark decoder is responsible for recovering the embedded message from the attacked watermarked image. $I_a \in \mathbb{R}^{3\times H\times W}$ denotes the distorted image produced by the noiser. The decoder estimates the hidden watermark sequence as

$$\hat{m} = D(I_a) \tag{6}$$

where $D(\cdot)$ represents the decoding network and $\hat{m} \in \mathbb{R}^L$ is the recovered message vector of length L. In contrast to the encoder, which is explicitly conditioned on the expected distortion type, the decoder operates only on the attacked image. This design is practically advantageous because message extraction does not require prior knowledge of the attack category at inference time.

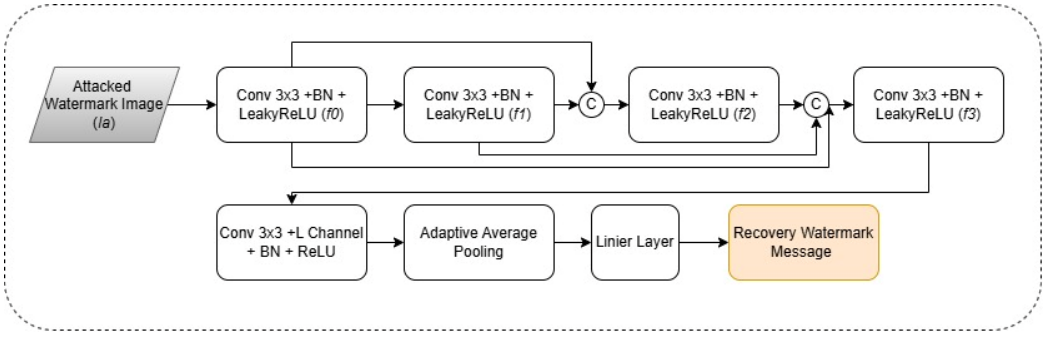


Figure 4. Structure of decoder

As shown in Fig. 4, the decoder adopts a hierarchical convolutional architecture with progressive feature aggregation. The input attacked image is first processed by a 3×3 convolutional layer followed by batch normalization and LeakyReLU activation to produce the initial feature map f_0. A second convolutional block further refines this representation and yields f_1. Rather than relying on a strictly sequential feature transformation, the decoder progressively concatenates intermediate feature maps so that both low-level and high-level information can be jointly exploited during message reconstruction. Specifically, the third stage receives the concatenation of f_0 and f_1, while the fourth stage receives the concatenation of f_0, f_1 and f_2. This progressive aggregation strategy preserves complementary structural cues that may still carry watermark traces after distortion. The intermediate decoding process can be expressed as

$$
\begin{aligned}
f_0 &= \phi_1(I_a), \\
f_1 &= \phi_2(f_0), \\
f_2 &= \phi_3([f_0, f_1]), \\
f_3 &= \phi_4([f_0, f_1, f_2]),
\end{aligned} \tag{7}
$$

where $[\cdot]$ denotes channel-wise concatenation and $\phi_i(\cdot)$ denotes a convolutional transformation followed by normalization and nonlinear activation. Through this design, the decoder is encouraged to combine shallow texture evidence and deeper semantic patterns when inferring the hidden message, which is particularly important when watermark-carrying regions have been partially corrupted by blur, compression, cropping, or additive noise.

After hierarchical feature extraction, the decoder maps the final feature representation into an L-channel message-oriented tensor through another convolutional projection layer. This tensor is then compressed by adaptive average pooling to a 1x1 spatial resolution, producing a global descriptor that summarizes the watermark evidence across the entire image. The pooled descriptor is finally passed through a fully connected layer to generate the recovered watermark vector. This stage can be written as

$$
\begin{aligned}
z &= \phi_5(f_3), \\
g &= \mathrm{GAP}(z), \\
\hat{m} &= Wg + b,
\end{aligned} \tag{8}
$$

where z is the message-oriented feature map, $\mathrm{GAP}(\cdot)$denotes global average pooling, and $W$and b are the learnable parameters of the final linear layer. The use of global pooling is important because the embedded watermark is spatially distributed over the image, and successful extraction depends on aggregating weak but correlated evidence from multiple regions rather than relying on a single local patch.

3.5. Discriminator

To preserve the visual quality of the watermarked image, the proposed framework incorporates a discriminator and trains the encoder-decoder branch in an adversarial manner. While the encoder is responsible for embedding the watermark and the decoder is responsible for message recovery, the discriminator serves as a visual critic that distinguishes between the original cover image and the watermarked image. Through this competition, the embedding network is encouraged to hide the watermark in a way that minimizes perceptual distortion and makes the watermarked image appear as natural as possible.

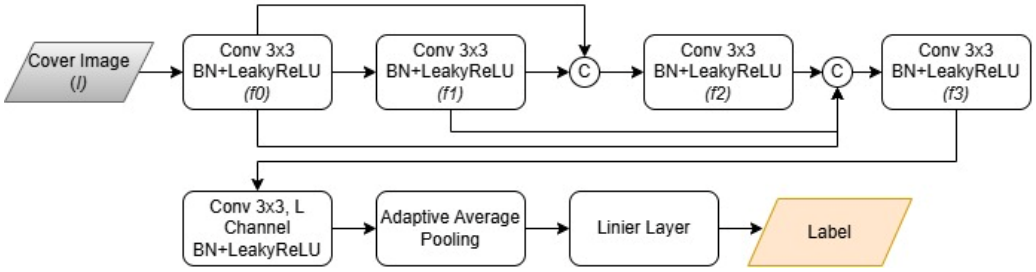


Figure 5. Structure of discriminator

Let I denote the original cover image and I_w denote the watermarked image generated by the encoder. The discriminator, denoted by $C(\cdot)$, produces a scalar output indicating whether the input image is a real cover image or an embedded one. In the proposed framework, the discriminator is trained to classify I as real and I_w as fake. Unlike the decoder, which operates on attacked images after the noiser, the discriminator evaluates the watermarked image before attack simulation. This design choice is intentional, since the role of the discriminator is to constrain embedding invisibility, whereas robustness against distortions is handled by the encoder-noiser-decoder pipeline.

Architecturally, the discriminator follows a hierarchical convolutional design similar to the decoder. The input image is first processed by a sequence of convolutional layers with batch normalization and LeakyReLU activation. Intermediate features are progressively aggregated through channel-wise concatenation, allowing the discriminator to jointly analyze low-level texture inconsistencies and higher-level structural artifacts introduced by watermark embedding. The final feature map is then compressed by adaptive average pooling and mapped to a single logit through a fully connected layer. Formally, the discriminator output can be written as

$$y = C(x), \quad x \in \{I, I_w\} \tag{9}$$

where y is the predicted realism score of the input image. Since the discriminator is optimized as a binary classifier, its output is trained using binary cross-entropy loss.

3.6. Evaluation Metrics

Watermark robustness is evaluated using bit accuracy (BA), while image fidelity is assessed using peak signal-to-noise ratio (PSNR) and structural similarity index measure (SSIM). For each attacked watermarked image I_a, the decoder D produces a recovered watermark message $\hat{m}$. After rounding and clipping the decoder output to $[0,1]$, the bit accuracy for a watermark of length L is calculated as

$$\mathrm{BA}_i = \frac{1}{L}\sum_{j=1}^{L} \mathbb{1}\left[m_{i,j} = \mathrm{clip}\left(\mathrm{round}\left(\hat{m}_{i,j}\right), 0, 1\right)\right] \tag{10}$$

where m is the original watermark message and $L = 30$ in all experiments. PSNR measures pixel-level fidelity between the cover image I and the corresponding watermarked image I_w. Since the images are represented in the range $[-1,1]$, with a dynamic range of 2, PSNR is calculated as

$$\mathrm{PSNR}_i = 10\ \log_{10}\left(\frac{4}{\mathrm{MSE}\left(I_i,\ I_{w,i}\right)}\right) \tag{11}$$

SSIM measures local structural similarity between I and I_w and is defined as:

$$\mathrm{SSIM}(x,y) = \frac{\left(2\mu_x\mu_y + C_1\right)\left(2\sigma_{xy} + C_2\right)}{\left(\mu_x^2 + \mu_y^2 + C_1\right)\left(\sigma_x^2 + \sigma_y^2 + C_2\right)} \tag{12}$$

where μ, σ^2 and σ_{xy} denote the local mean, variance, and covariance, respectively. An 11×11 Gaussian window is used, with $C_1 = 0.01^2$ and $C_2 = 0.03^2$. For each evaluation condition, BA, PSNR, and SSIM are first obtained at the image level and then averaged over all N images in the corresponding test dataset. When BA is summarized across multiple distortion conditions, the reported value is the macro-average of the corresponding condition-level BA values.

4. Results and Discussion

The proposed watermarking model was implemented using PyTorch and executed on an NVIDIA GeForce RTX 4060 GPU. The Microsoft Common Objects in Context (COCO) dataset [32] was used for training and testing. The Pattern Analysis, Statistical Modelling and Computational Learning Visual Object Classes 2012 (PASCAL VOC2012) [33], ImageNet [34], and Enhancing Underwater Visual Perception (EUVP) dataset [35] were included as additional evaluation datasets to assess cross-dataset generalization. The training run comprised 100 epochs, using a batch size of 8 and a watermark length of $L = 30\ bits$. The checkpoint from epoch 96 was used for evaluation. During both training and validation, a new binary watermark message was independently generated for each image in every batch, with each bit sampled from a Bernoulli distribution with probability 0.5. Training images were randomly cropped to 128×128 pixels, with padding applied when necessary, whereas validation images were center-cropped to the same spatial size. The training script did not explicitly set a global random seed. For testing, a single 30-bit watermark message was generated using NumPy with a random seed of 42 and reused for all test images. Test images were resized to

128×128 pixels before evaluation. All supported image files in each designated evaluation directory were included without random subsampling, comprising 1,250 images each for COCO, PASCAL VOC2012, and ImageNet, and 515 images for EUVP. Image paths were sorted before processing to ensure a consistent evaluation order. The loss-function weights were set to λ_1=0.7, λ_2=0.1, λ_3=1.5, λ_4=1×10^{-3}. For Adam gradient descent optimization, the learning rate was set to $\mathrm{lr} = 1 \times 10^{-3}$.

To investigate the effectiveness of noise training, NAA-GAN was trained using five types of noise, namely dropout (30%), Gaussian blur (2.0), JPEG compression (quality factor of 50), resizing (0.8), and random cropping (3.5%). For contextual comparison, ARWGAN was adopted as the baseline model, together with ReDMark [23], DA [17], TSDL [36], MBRS [21], and SSLW [24]. For methods that were not re-evaluated within our implementation, the results reported in the corresponding publications were used as comparative references. The watermark length was restricted to L=30 bits where comparable results were available. However, because the reported results may differ in implementation details, training procedures, and evaluation protocols, these comparisons should be interpreted as indicative rather than as a fully controlled benchmark.

In this section, the evaluation focuses primarily on watermark recovery robustness under trained, unseen, and compound distortions, using bit accuracy (BA) as the main robustness metric. We first examine recovery under distortions not used during training, including compound conditions, and then compare the proposed method with existing watermarking models across COCO, PASCAL VOC2012, ImageNet, and EUVP. An ablation study is subsequently used to assess the contribution of the distortion descriptor to recovery accuracy. PSNR and SSIM are reported as complementary measures to verify that robustness is achieved without excessive visual degradation.

4.1. Watermarking Invisibility Evaluation

Table 1 compares the imperceptibility performance of different watermarking models using the Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM). Higher values for both metrics indicate lower visual distortion and greater similarity between the original and watermarked images. ARWGAN-NN achieves the best overall performance, obtaining the highest PSNR of 41.48 dB and the highest SSIM of 0.9900. These results indicate that ARWGAN-NN can embed watermark information while introducing minimal perceptual degradation.

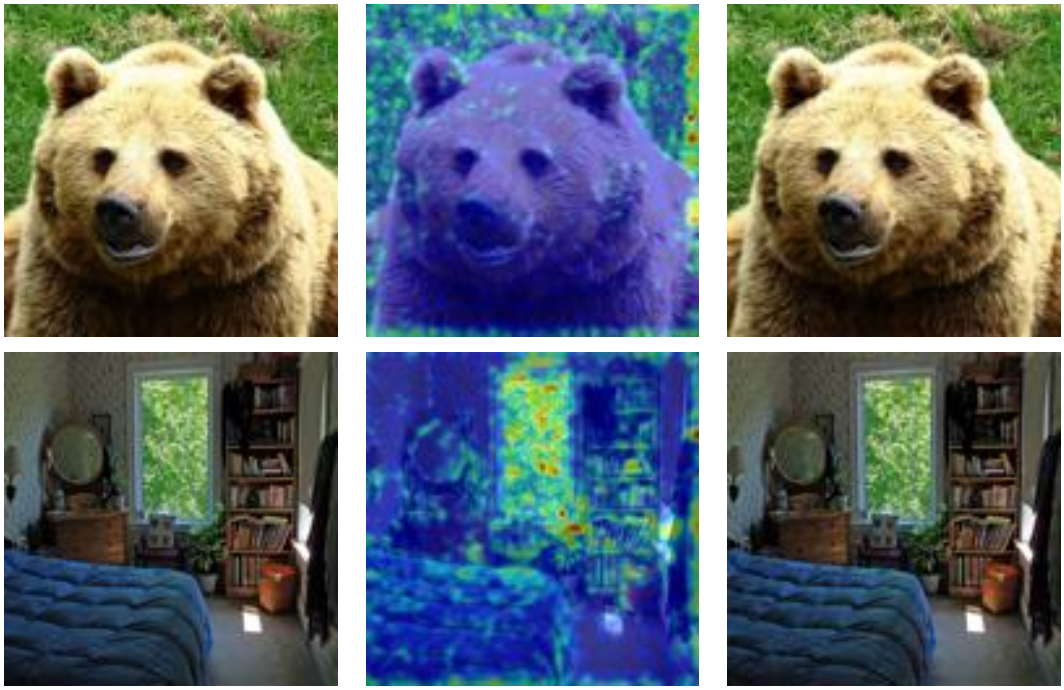
NAA-GAN achieves the second-highest PSNR of 37.38 dB, outperforming Identity, ARWGAN-CN, and HiDDeN in terms of pixel-level fidelity. However, its SSIM of 0.9605 is lower than those of Identity and ARWGAN-CN, indicating that higher PSNR does not necessarily correspond to better preservation of structural similarity. Identity achieves a PSNR of 36.72 dB and an SSIM of 0.9727, while ARWGAN-CN records a PSNR of 35.87 dB and an SSIM of 0.9688. In contrast, HiDDeN obtains the lowest PSNR and SSIM, at 32.20 dB and 0.9227, respectively. Overall, ARWGAN-NN provides the strongest imperceptibility performance among the evaluated models, whereas NAA-GAN offers competitive pixel-level fidelity.

Table 1. Invisibility Of Different Watermarking Models

Model	Invisibility	
	PSNR	**SSIM**
Identity [19]	36.72	0.9727
ARWGAN-NN [1]	41.48	0.9900
HiDDeN [19]	32.20	0.9227
ARWGAN-CN [1]	35.87	0.9688
NAA-GAN	37.38	0.9605

Figure 6 provides a qualitative illustration of NAA-GAN's watermarking performance by showing the cover image, the corresponding attention mask, and the encoded image. As shown in Fig. 6(a) and Fig. 6(c), no obvious visual differences are apparent between the original and watermarked images, supporting the quantitative evaluation of image fidelity. Figure

6(b) shows the attention masks generated for different images, with color variations representing different attention levels.



(a) (b) (c)

Figure 6. Watermarking performance of NAA-GAN: (**a**) Cover image; (**b**) Attention mask; (**c**) Encoded image

Although preserving image quality is essential, imperceptibility alone does not establish whether the embedded watermark can be reliably recovered after image processing. In practical applications, watermarked images may undergo distortions that differ from those encountered during model training. Robustness to such unseen distortions is therefore important for evaluating the generalization capability of a watermarking model. To investigate this aspect, Table 2 compares NAA-GAN with Identity, ARWGAN-NN, HiDDeN, and ARWGAN-CN under individual and compound distortions not included during training. The evaluation covers noise, filtering, brightness and contrast adjustments, hue and saturation changes, cropping, cropout, and combinations of these operations. The results allow the robustness of the models to be compared both overall and across different distortion types, revealing where NAA-GAN maintains its advantages and where its performance remains limited.

Table 2 evaluates the robustness of NAA-GAN against four comparison models under distortions not encountered during training, including individual and compound attacks. The results indicate that NAA-GAN achieves the strongest overall performance across the evaluated conditions, suggesting that its distortion-aware design helps preserve watermark recoverability beyond the training distribution. Its robustness is particularly evident under Gaussian noise, median filtering, and several compound attacks, where it consistently maintains high bit accuracy.

However, the results also reveal that this advantage is not uniform across all distortion types. ARWGAN-CN performs better under certain spatial cropping operations, particularly grid crop and cropout, indicating that NAA-GAN remains vulnerable to some forms of spatial information loss. This contrast highlights the importance of distinguishing between distortions that modify image intensities and those that remove or rearrange spatial content, as their effects on watermark recovery differ. Moreover, NAA-GAN's strong performance under several unseen and compound attacks suggests that its robustness extends beyond isolated distortions, although the effectiveness of this generalization depends on the specific attack characteristics. NAA-GAN achieves an overall average bit accuracy of 92.59%, compared with 91.82% for ARWGAN-CN. This difference is descriptive only, as repeated experiments

and uncertainty estimates were not available; therefore, statistical superiority cannot be established.

Table 2. Robustness to Untrained Noises

Noises	Identity	ARWGAN -NN	HiDDeN	ARWGAN -CN	NAA-GAN
Gaussian Noise (0.06)	84.85	71.01	92.38	92.84	**99.47**
Gaussian Noise (0.08)	79.14	66.19	90.39	89.60	**99.24**
Gaussian Noise (0.10)	74.85	63.42	87.89	86.61	**98.88**
Salt&Pepper (0.05)	94.00	96.42	93.57	**99.41**	96.26
Salt&Pepper (0.10)	76.61	85.33	83.02	**96.34**	87.65
Salt&Pepper (0.15)	65.16	73.37	73.21	**90.73**	81.14
Median filter (3x3)	45.27	57.74	84.66	99.71	**99.76**
Median Filter (5x5)	41.83	57.10	70.23	91.76	**93.41**
Median filter (7x7)	39.54	58.12	63.70	86.47	**87.18**
Adjust Brightness (1.1)	91.33	89.82	93.97	**98.28**	94.31
Adjust Brightness (1.2)	88.68	86.53	93.00	**94.41**	92.90
Adjust Brightness (1.3)	85.39	82.98	91.58	**94.28**	91.21
Adjust Contrast (1.0)	81.19	83.93	90.02	94.97	**95.33**
Adjust Contrast (1.5)	73.10	77.49	87.77	93.09	**93.52**
Adjust Contrast (2.0)	67.38	72.49	85.04	90.95	**91.46**
Adjust Hue (0.2)	88.96	93.41	90.69	**93.89**	**99.78**
Adjust Hue (0.3)	83.28	86.30	82.77	84.94	**99.62**
Adjust Hue (0.4)	79.59	79.02	79.91	78.13	**98.89**
Adjust Saturation (15)	77.85	78.21	85.25	**95.41**	90.36
Adjust Saturation (20)	76.86	77.74	84.98	**93.82**	90.09
Adjust Saturation (25)	76.23	77.51	83.77	**93.41**	89.91
Grid Crop (50%)	96.32	94.07	94.25	**99.82**	95.93
Grid crop (60%)	90.58	86.59	90.39	**99.34**	93.87
Grid crop (70%)	78.55	76.63	81.87	**97.60**	91.20
Cropout (30%)	77.63	86.67	93.82	**99.98**	92.16
Cropout (20%)	69.23	79.22	89.57	**98.08**	88.15
Cropout (10%)	58.40	67.31	76.44	**88.13**	81.86
Gaussian Noise (0.08) + Salt&Pepper(0.10)	61.55	61.75	76.96	82.48	**91.01**
Adjust Brightness (1.2) +Adjust Contrast (1.5)	73.98	83.40	86.75	90.99	**91.35**
Cropout(20%) +Grid Crop (60%)	53.63	59.79	54.94	73.49	**89.01**
Grid Crop(50%)+Adjust Brightness (1.1) +Median filter(3x3)	52.97	54.94	68.53	77.62	**85.38**
Average	73.68	76.27	83.91	91.82	**92.59**

4.2. Comparison with Existing Watermarking Models

To place the performance of NAA-GAN in context, its image fidelity and watermark robustness are compared with the reported results of several existing watermarking methods. As some comparison values are obtained from previously published studies, they are treated as reference points rather than results from a uniformly controlled re-evaluation. Table 3 presents the comparison on COCO.

As shown in Table 3, NAA-GAN achieves the highest PSNR and JPEG BA among the compared methods on COCO. Its SSIM is also competitive, although ARWGAN-CN obtains a higher value. However, NAA-GAN performs less favorably under cropout, where its BA is substantially lower than those of ARWGAN-CN and MBRS. Consequently, its average

BA of 98.07% remains slightly below the 98.39% achieved by ARWGAN-CN. These results indicate that NAA-GAN offers advantages in image fidelity and selected robustness conditions, but does not achieve the highest overall robustness across all evaluated attacks. To examine whether this performance pattern extends to other image collections, NAA-GAN was further evaluated on PASCAL VOC2012, ImageNet, and EUVP, with the results presented in Tables 4, 5, and 6, respectively.

Table 3. Comparison Of Different Models on COCO

Model	Invisibility		Robustness					
	PSNR	SSIM	JPEG (50)	Cropout (30%)	Dropout (30%)	Crop (3.5%)	Gaussian (2.0%)	Average
ReDMark [23]	35.93	0.9660	74.60	92.50	92.00	**100.00**	50.00	81.82
DA [17]	33.70	-	81.70	-	97.90	93.50	60.00	83.28
TSDL [36]	33.50	-	76.20	97.30	97.40	89.00	98.60	91.70
MBRS [21]	35.84	0.8899	91.97	**99.98**	99.96	92.68	**100.00**	96.92
SSLW [24]	34.00	0.8725	83.01	79.66	88.11	50.73	98.96	80.09
ARWGAN-CN [1]	35.87	**0.9688**	93.98	99.82	**100.00**	98.17	99.99	**98.39**
NAA-GAN	**37.38**	0.9605	**99.02**	92.16	99.79	99.70	99.72	98.07

Table 4. Comparison of Different Models on PASCAL VOC2012

Model	Invisibility		Robustness					
	PSNR	SSIM	JPEG (50)	Cropout (30%)	Dropout (30%)	Crop (3.5%)	Gaussian (2.0%)	Average
HiDDeN [19]	33.37	0.9333	77.58	96.05	95.76	96.02	93.40	91.76
MBRS [21]	36.64	0.8938	97.99	100.00	100.00	93.21	100.00	98.24
SSLW [24]	33.90	0.8613	77.33	77.65	87.88	50.65	99.40	78.90
ARWGAN-CN [1]	36.83	0.9698	94.45	100.00	100.00	99.99	100.00	98.89
NAA-GAN	37.79	0.9615	99.22	92.49	99.90	99.83	99.88	98.26

Table 5. Comparison of Different Models on ImageNet

Model	Invisibility		Robustness					
	PSNR	SSIM	JPEG (50)	Cropout (30%)	Dropout (30%)	Crop (3.5%)	Gaussian (2.0%)	Average
HiDDeN [19]	33.26	0.9322	77.30	95.98	95.67	96.02	92.89	91.57
MBRS [21]	36.44	0.8919	92.64	99.99	99.98	92.93	100.00	97.11
SSLW [24]	33.50	0.8412	77.48	80.22	86.88	50.52	98.91	79.13
ARWGAN-CN [1]	36.66	**0.9685**	94.08	**99.99**	**100.00**	**99.95**	**99.98**	**98.80**
NAA-GAN	**37.59**	0.9607	**99.05**	92.75	99.77	99.72	99.68	98.19

As reported in Table 4, NAA-GAN achieves the highest PSNR and JPEG BA among the compared methods on PASCAL VOC2012. Its BA under cropping is also high, demonstrating robustness to this spatial distortion. Nevertheless, ARWGAN-CN performs better under cropout and dropout and achieves the highest SSIM and average BA. The average BA of NAA-GAN is close to that of MBRS but remains below that of ARWGAN-CN. Thus, the PASCAL VOC2012 results reinforce the pattern observed on COCO: NAA-GAN performs

favorably in image fidelity and JPEG robustness, whereas ARWGAN-CN retains an advantage in aggregate robustness.

The results in Table 5 show a similar pattern on ImageNet. NAA-GAN records the highest PSNR and JPEG BA among the reported methods, while its BA under cropping remains close to that of ARWGAN-CN. However, the difference in cropout performance persists, with ARWGAN-CN achieving substantially higher BA. This difference contributes to its higher average BA, despite NAA-GAN's advantages in image fidelity and JPEG robustness. ARWGAN-CN also achieves a higher SSIM, further illustrating that higher PSNR does not necessarily correspond to the highest structural similarity.

Table 6. Comparison of Different Models on EUVP

Model	Invisibility		Robustness					
	PSNR	SSIM	JPEG (50)	Cropout (30%)	Dropout (30%)	Crop (3.5%)	Gaussian (2.0%)	Avg
HiDDeN [19]	33.53	0.9272	79.98	95.49	95.34	95.07	93.76	91.93
MBRS [21]	36.19	0.9009	91.88	99.98	99.96	93.11	**100.00**	96.99
SSLW [24]	34.20	0.8531	83.62	77.69	88.22	50.13	99.27	79.79
ARWGAN-CN [1]	36.39	0.9623	93.63	**99.99**	**100.00**	**99.97**	99.97	**98.72**
NAA-GAN	**36.96**	0.9563	**99.36**	92.31	99.93	99.85	99.83	98.25

As shown in Table 6, the comparison on EUVP, which extends the evaluation to underwater images, yields consistent findings. NAA-GAN achieves the highest PSNR and JPEG BA among the compared methods and maintains high BA under dropout, cropping, and the reported Gaussian distortion. However, its cropout BA remains lower than that of ARWGAN-CN. Accordingly, ARWGAN-CN achieves a higher average BA and SSIM, while NAA-GAN retains advantages in PSNR and JPEG robustness. The EUVP results therefore support the performance pattern observed across the other datasets rather than indicating uniform superiority across all evaluation criteria.

Overall, the comparisons in Tables 3–6 reveal a consistent distinction between image fidelity and robustness. Across all four datasets, NAA-GAN achieves the highest PSNR and JPEG BA among the listed comparison methods, while its average BA remains above 98%. ARWGAN-CN, however, achieves the highest SSIM and average BA on each dataset. The recurring performance gap under cropout is an important contributor to this difference and suggests that robustness to spatial information loss remains a limitation of NAA-GAN. Therefore, the results support the effectiveness of NAA-GAN in improving image fidelity and JPEG robustness, but do not establish universal superiority over existing methods. Its performance is better characterized as a favorable balance across selected evaluation criteria, with further scope for improving robustness under cropout attacks.

4.3. Ablation Experiments and Analysis

To examine the contribution of the six-dimensional distortion descriptor, an ablation variant of NAA-GAN was trained and evaluated without the descriptor. The complete NAA-GAN and its ablation variant were evaluated on COCO using the same five attacks. ARWGAN-CN is included as an external reference for interpreting their performance. Table 7 presents the corresponding invisibility and robustness results.

Table 7. Comparison and Ablation of the Distortion Descriptor

Model	Invisibility		Robustness					
	PSNR	SSIM	JPEG (50)	Cropout (30%)	Dropout (30%)	Crop (3.5%)	Gaussian (2.0%)	Average
ARWGAN-CN [1]	35.87	**0.9688**	93.98	99.82	**100.00**	98.17	99.99	**98.39**
NAA-GAN without distortion descriptor	36.73	0.9521	97.00	90.57	97.54	97.94	97.64	96.14
NAA-GAN	**37.38**	0.9605	**99.02**	92.16	99.79	99.70	99.72	98.07

As shown in Table 7, removing the distortion descriptor reduces the PSNR from 37.38 to 36.73 dB and the SSIM from 0.9605 to 0.9521. The average BA also decreases from 98.07% to 96.14%, a difference of 1.93 percentage points. This reduction is observed under every evaluated attack. In particular, the complete NAA-GAN improves BA by 2.02 percentage points under JPEG compression, 1.59 points under cropout, 2.25 points under dropout, 1.76 points under cropping, and 2.08 points under Gaussian blur relative to the variant without the descriptor. These results indicate that the descriptor contributes to both watermark invisibility and extraction accuracy under the tested distortions.

Compared with ARWGAN-CN, the complete NAA-GAN achieves a higher PSNR (37.38 versus 35.87 dB) and higher BAs under JPEG compression (99.02% versus 93.98%) and cropping (99.70% versus 98.17%). ARWGAN-CN, however, retains a higher SSIM (0.9688 versus 0.9605) and average BA (98.39% versus 98.07%). Its advantage under cropout is particularly pronounced, with a BA of 99.82% compared with 92.16% for NAA-GAN. Overall, the comparison between the two NAA-GAN variants supports the contribution of the distortion descriptor, while the ARWGAN-CN results show that further improvement is possible, especially under cropout.

5. Conclusions

This study introduced NAA-GAN, a blind image watermarking framework designed to improve watermark embedding by adapting the encoder to anticipated distortion characteristics. Its main contribution is the integration of a six-dimensional distortion descriptor into a noise-aware cross-attention mechanism, enabling the encoder to modulate image features during watermark insertion while keeping the decoder independent of the descriptor. This design aims to improve the adaptability of watermark embedding without requiring additional distortion information during watermark extraction.

The experimental findings demonstrate the potential of distortion-conditioned embedding to improve pixel-level image fidelity and robustness to JPEG compression across different image datasets. The evaluation under unseen and compound distortions further indicates that the proposed framework can maintain watermark recoverability beyond the distortion conditions encountered during training. However, the results also show that these improvements do not translate into uniform superiority across all evaluation criteria. In particular, the comparatively weaker performance under cropout highlights the continuing challenge of preserving watermark information when substantial spatial content is removed. The ablation study further supports the contribution of the distortion descriptor to the overall performance of NAA-GAN.

Future work will investigate more effective strategies for preserving watermark information under spatial information loss and diverse compound attacks. Further evaluation using repeated experiments and uncertainty estimates would also help establish the consistency and statistical reliability of the observed performance differences. These directions may improve the robustness and generalizability of distortion-aware watermarking while retaining the image fidelity achieved by the proposed framework.

Author Contributions: Conceptualization: S.E. and F.E.; Methodology: S.E. and F.E.; Software: S.E.; Validation: S.E., J.N., and F.E.; Formal analysis: S.E. and F.E.; Investigation: S.E. and F.E.; Resources: S.E.; Data curation: S.E.; Writing, original draft preparation: S.E.; Writing, review and editing: S.E., J.N., and F.E.; Visualization: S.E. and F.E.; Supervision: J.N. and F.E.; Project administration: S.E. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The datasets used in this study are publicly available. The experiments were conducted using the Microsoft COCO (MS COCO), PASCAL VOC2012, ImageNet, and EUVP datasets, which can be obtained from their respective public repositories. No new dataset was created specifically for this study. Additional data supporting the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgments: The authors acknowledge the use of generative AI tools to improve the clarity and grammar of the manuscript. All technical content was written and verified by the authors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

[1] J. Huang, T. Luo, L. Li, G. Yang, H. Xu, and C.-C. Chang, "ARWGAN: Attention-Guided Robust Image Watermarking Model Based on GAN," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–17, 2023, doi: 10.1109/TIM.2023.3285981.

[2] L. Xiong, X. Han, C.-N. Yang, and Y.-Q. Shi, "Robust Reversible Watermarking in Encrypted Image With Secure Multi-Party Based on Lightweight Cryptography," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 1, pp. 75–91, Jan. 2022, doi: 10.1109/TCSVT.2021.3055072.

[3] I. C. Dragoi and D. Coltuc, "On the Security of Reversible Data Hiding in Encrypted Images by MSB Prediction," *IEEE Trans. Inf. Forensics Secur.*, vol. 16, pp. 187–189, 2021, doi: 10.1109/TIFS.2020.3006382.

[4] L.-C. Kuo and C.-C. Tai, "Robust Image-Based Water-Level Estimation Using Single-Camera Monitoring," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–11, 2022, doi: 10.1109/TIM.2022.3161691.

[5] J. Molina-Garcia, B. P. Garcia-Salgado, V. Ponomaryov, R. Reyes-Reyes, S. Sadovnychiy, and C. Cruz-Ramos, "An effective fragile watermarking scheme for color image tampering detection and self-recovery," *Signal Process. Image Commun.*, vol. 81, no. October 2019, p. 115725, Feb. 2020, doi: 10.1016/j.image.2019.115725.

[6] P. Lefèvre, P. Carré, C. Fontaine, P. Gaborit, and J. Huang, "Efficient image tampering localization using semi-fragile watermarking and error control codes," *Signal Processing*, vol. 190, p. 108342, Jan. 2022, doi: 10.1016/j.sigpro.2021.108342.

[7] X. Wang, X. Li, and Q. Pei, "Independent Embedding Domain Based Two-Stage Robust Reversible Watermarking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 8, pp. 2406–2417, Aug. 2020, doi: 10.1109/TCSVT.2019.2915116.

[8] A. Winarno, D. R. I. M. Setiadi, A. A. Arrasyid, C. A. Sari, and E. H. Rachmawanto, "Image watermarking using low wavelet subband based on 8×8 sub-block DCT," in *2017 International Seminar on Application for Technology of Information and Communication (iSemantic)*, Oct. 2017, pp. 11–15. doi: 10.1109/ISEMANTIC.2017.8251835.

[9] T. Jin and W. Zhang, "A Novel Interpolated DFT Synchrophasor Estimation Algorithm With an Optimized Combined Cosine Self-Convolution Window," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–10, 2021, doi: 10.1109/TIM.2020.3033073.

[10] S. Shi, T. Luo, J. Huang, and M. Du, "A Novel HDR Image Zero-Watermarking Based on Shift-Invariant Shearlet Transform," *Secur. Commun. Networks*, vol. 2021, pp. 1–12, Mar. 2021, doi: 10.1155/2021/6662700.

[11] H. J. Ko, C. T. Huang, G. Horng, and S. J. WANG, "Robust and blind image watermarking in DCT domain using inter-block coefficient correlation," *Inf. Sci. (Ny).*, vol. 517, pp. 128–147, 2020, doi: 10.1016/j.ins.2019.11.005.

[12] H. T. Hu, L. Y. Hsu, and H. H. Chou, "An improved SVD-based blind color image watermarking algorithm with mixed modulation incorporated," *Inf. Sci. (Ny).*, vol. 519, pp. 161–182, May 2020, doi: 10.1016/j.ins.2020.01.019.

[13] C. Zhou *et al.*, "A New Model Transfer Strategy Among Spectrometers Based on SVR Parameter Calibrating," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–13, 2021, doi: 10.1109/TIM.2021.3119129.

[14] C. Yang, P. An, and L. Shen, "Blind Image Quality Measurement via Data-Driven Transform-Based Feature Enhancement," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–12, 2022, doi: 10.1109/TIM.2022.3191661.

[15] A. Lin, B. Chen, J. Xu, Z. Zhang, G. Lu, and D. Zhang, "DS-TransUNet: Dual Swin Transformer U-Net for Medical Image Segmentation," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–15, 2022, doi: 10.1109/TIM.2022.3178991.

[16] B. Chen, Z. Zhang, Y. Li, G. Lu, and D. Zhang, "Multi-Label Chest X-Ray Image Classification via Semantic Similarity Graph Embedding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 4, pp. 2455–2468, Apr. 2022, doi: 10.1109/TCSVT.2021.3079900.

[17] X. Luo, R. Zhan, H. Chang, F. Yang, and P. Milanfar, "Distortion Agnostic Deep Watermarking," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2020, pp. 13545–13554. doi: 10.1109/CVPR42600.2020.01356.

[18] Z. Huang, J. Zhang, Y. Zhang, and H. Shan, "DU-GAN: Generative Adversarial Networks With Dual-Domain U-Net-Based Discriminators for Low-Dose CT Denoising," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–12, 2022, doi: 10.1109/TIM.2021.3128703.

[19] J. Zhu, R. Kaplan, J. Johnson, and L. Fei-Fei, "HiDDeN: Hiding Data With Deep Networks," in *Computer Vision – ECCV 2018*, 2018, pp. 682–697. doi: 10.1007/978-3-030-01267-0_40.

[20] K. Hao, G. Feng, and X. Zhang, "Robust image watermarking based on generative adversarial network," *China Commun.*, vol. 17, no. 11, pp. 131–140, Nov. 2020, doi: 10.23919/JCC.2020.11.012.

[21] Z. Jia, H. Fang, and W. 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*, Oct. 2021, pp. 41–49. doi: 10.1145/3474085.3475324.

[22] H. Chen and L. Zhang, "A Novel Digital watermarking Approach under the DCT and SVM," in *2021 IEEE 4th International Conference on Automation, Electronics and Electrical Engineering (AUTEEE)*, Nov. 2021, pp. 154–156. doi: 10.1109/AUTEEE52864.2021.9668682.

[23] M. Ahmadi, A. Norouzi, N. Karimi, S. Samavi, and A. Emami, "ReDMark: Framework for residual diffusion watermarking based on deep networks," *Expert Syst. Appl.*, vol. 146, p. 113157, May 2020, doi: 10.1016/j.eswa.2019.113157.

[24] P. Fernandez, A. Sablayrolles, T. Furon, H. Jegou, and M. Douze, "Watermarking Images in Self-Supervised Latent Spaces," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2022, pp. 3054–3058. doi: 10.1109/ICASSP43922.2022.9746058.

[25] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 2261–2269. doi: 10.1109/CVPR.2017.243.

[26] Z. Wu, J. Hai, L. Zhang, J. Chen, G. Cheng, and B. Yan, "Cascaded Fully Convolutional DenseNet for Automatic Kidney Segmentation in Ultrasound Images," in *2019 2nd International Conference on Artificial Intelligence and Big Data (ICAIBD)*, May 2019, pp. 384–388. doi: 10.1109/ICAIBD.2019.8836994.

[27] C.-F. R. Chen, Q. Fan, and R. Panda, "CrossViT: Cross-Attention Multi-Scale Vision Transformer for Image Classification," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2021, pp. 347–356. doi: 10.1109/ICCV48922.2021.00041.

[28] B. Chen, W. Tan, G. Coatrieux, Y. Zheng, and Y.-Q. Shi, "A Serial Image Copy-Move Forgery Localization Scheme With Source/Target Distinguishment," *IEEE Trans. Multimed.*, vol. 23, pp. 3506–3517, 2021, doi: 10.1109/TMM.2020.3026868.

[29] C. Yu, "Attention Based Data Hiding with Generative Adversarial Networks," *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 01, pp. 1120–1128, Apr. 2020, doi: 10.1609/aaai.v34i01.5463.

[30] H. Zhang, H. Wang, Y. Cao, C. Shen, and Y. Li, "Robust Data Hiding Using Inverse Gradient Attention," *Arxiv preprint*. Oct. 17, 2022. [Online]. Available: http://arxiv.org/abs/2011.10850

[31] K. A. Zhang, L. Xu, A. Cuesta-Infante, and K. Veeramachaneni, "Robust Invisible Video Watermarking with Attention," *Arxiv preprint*. Sep. 03, 2019. doi: 10.48550/arXiv.1909.01285.

[32] T.-Y. Lin *et al.*, "Microsoft COCO: Common Objects in Context," in *Computer Vision – ECCV 2014*, 2014, pp. 740–755. doi: 10.1007/978-3-319-10602-1_48.

[33] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jun. 2010, doi: 10.1007/s11263-009-0275-4.

[34] J. Deng, W. Dong, R. Socher, L.-J. Li, Kai Li, and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2009, pp. 248–255. doi: 10.1109/CVPR.2009.5206848.

[35] M. J. Islam, Y. Xia, and J. Sattar, "Fast Underwater Image Enhancement for Improved Visual Perception," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 3227–3234, Apr. 2020, doi: 10.1109/LRA.2020.2974710.

[36] Y. Liu, M. Guo, J. Zhang, Y. Zhu, and X. Xie, "A Novel Two-stage Separable Deep Learning Framework for Practical Blind Watermarking," in *Proceedings of the 27th ACM International Conference on Multimedia*, Oct. 2019, pp. 1509–1517. doi: 10.1145/3343031.3351025.