



Performance Evaluation of Associative Watermarking Using Statistical Neurodynamics

Ryoto Kanegae and Masaki Kawamura*

Graduate School of Sciences and Technology for Innovation, Yamaguchi University, Yamaguchi 753-8512, Japan

(Received February 6, 2024; revised August 26, 2024; accepted September 11, 2024; published online October 16, 2024)

We theoretically evaluated the performance of our proposed associative watermarking method in which the watermark is not embedded directly into the image. We previously proposed a watermarking method that extends the zero-watermarking model by applying associative memory models. In this model, the hetero-associative memory model is introduced to the mapping process between image features and watermarks, and the auto-associative memory model is applied to correct watermark errors. We herein show that the associative watermarking model outperforms the zero-watermarking model through computer simulations using actual images. In this paper, we describe how we derive the macroscopic state equation for the associative watermarking model using the Okada theory. The theoretical results obtained by the fourth-order theory were in good agreement with those obtained by computer simulations. Furthermore, the performance of the associative watermarking model was evaluated using the bit error rate of the watermark, both theoretically and using computer simulations.

1. Introduction

Illegal use of digital content in social media is a serious problem. In particular, unauthorized use and tampering with images and videos have been frequently detected. The digital watermarking method^{1–3)} is an effective method to solve such problems. In this kind of method, information such as copyrights and licenses is imperceptibly embedded in digital content. This prevents unauthorized use of the content and identifies the authorized owner. The information embedded in the content is called a watermark. Embedding methods include pixel substitution,⁴⁾ which adds the watermark to a pixel, and embedding in the frequency domain obtained by discrete cosine transformations (DCT)^{5,6)} and other methods.⁷⁾ These embedding methods place the watermark directly into the content, causing distortion in the image.

The zero-watermarking method^{8–10)} was proposed as a method of undistorted image watermarking. In this method, instead of embedding a watermark in an image, a secret key is generated from the watermark and the unique features extracted from the image. The validity of the image can be determined by retrieving the watermark from the features and the stored secret key. Because this method never embeds a watermark, the image is never distorted. However, the watermark cannot be retrieved correctly if the original image is attacked because there is no error correction capability. Furthermore, if the zero-watermarking method were applied to a large number of images, a large number of secret keys would be generated, requiring that the pair between the secret keys and features be managed.

The hetero-associative memory model (HMM)¹¹⁾ can store pairs of key patterns and associative patterns, and the corresponding associative patterns can be recalled if a key pattern is input. Therefore, the HMM can be applied to achieve the zero-watermarking method by considering the features extracted from the images as key patterns and by considering the watermarks as associative patterns. Applying the HMM has the following advantages. First, the sizes of the features and watermark can be made different. Second, if the features have errors, the HMM can correct the errors in the

watermark. Third, pair management is no longer necessary because multiple pairs can be stored in the HMM. The auto-associative memory model (AMM)^{12–14)} stores multiple patterns, and the most similar pattern among the stored ones is recalled when a pattern is input. Therefore, watermark errors can be corrected by storing watermarks in the AMM.

We proposed an associative watermarking method (AWM) based on the HMM and AMM to enable a zero-watermarking method.¹⁵⁾ The AWM is composed of a HMM in the first layer and an AMM in the second layer, resulting in the same network structure as the human associative processor (HASP).^{16,17)} Owing to the error correction capability of the associative memory models, we showed using computer simulations that the watermark is retrieved without errors even when the image is attacked. In this paper, we present an evaluation of the error correction capability of the AWM through theory and computer simulations. Because AWM consists of the associative memory models, its recall process of the watermark and memory capacity can be obtained by using statistical neurodynamics, in particular, Okada theory.^{13,14)} The macroscopic state equations for the HASP have been derived.¹¹⁾ Since the structure of the proposed model is the same as that of HASP,¹¹⁾ its macroscopic state equations were derived. The recall process for one-to-many associations was analyzed in the previous study.¹¹⁾ In this paper, we present new results to characterize it as the AWM. We evaluated the bit error rate of the watermark by using the equations.

The paper is organized as follows. Section 2 explains the zero-watermarking method. Section 3 discusses associative memory models. Section 4 describes the associative watermarking method. Section 5 describes computer simulations that were conducted to evaluate the theory. Section 6 concludes the paper.

2. Zero-Watermarking Method

We explain the mechanism of the zero-watermarking method using DCT coefficients as features.⁸⁾ Let $\xi = (\xi_1, \xi_2, \dots, \xi_K)^\top$ be a K -bit watermark, where each element of the watermark ξ_i takes the value ± 1 .



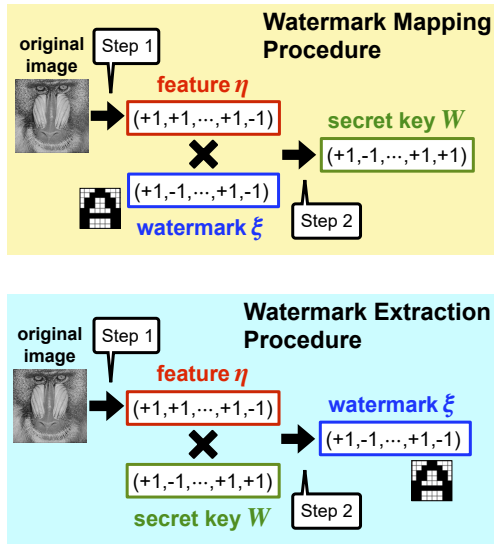


Fig. 1. (Color online) Mechanism of zero-watermarking method.

2.1 Mapping procedure from the features and watermark to the secret key

As shown in the upper part of Fig. 1, mapping from features and a watermark to a secret key involves two steps. **Step 1:** Extract features from the original image.

Two-dimensional DCT is performed on the original image. Low-frequency components are chosen as features because the changes in the DCT coefficients are small when the image is degraded. K low-frequency components $\mathbf{d} = (d_1, d_2, \dots, d_K)^T$ are extracted from the DCT coefficients, excluding the DC component. The feature $\boldsymbol{\eta} = (\eta_1, \eta_2, \dots, \eta_K)^T$ is obtained by binarizing these extracted DCT coefficients \mathbf{d} , that is, given by

$$\eta_i = \text{sgn}(d_i), \quad i = 1, 2, \dots, K, \quad (1)$$

where the sign function $\text{sgn}(\cdot)$ is defined as

$$\text{sgn}(x) = \begin{cases} +1, & x \geq 0 \\ -1, & x < 0 \end{cases}. \quad (2)$$

Step 2: Generate a secret key.

The secret key \mathbf{W} is generated by the Hadamard product of the feature $\boldsymbol{\eta}$ and the watermark $\boldsymbol{\xi}$. That is, it is calculated in Step 1. That is,

$$W_i = \eta_i \xi_i, \quad i = 1, 2, \dots, K. \quad (3)$$

Note that the bit length of the feature, K , must be the same as the bit length of the watermark. Finally, the generated secret key is stored.

2.2 Watermark extraction procedure

As shown in the lower part of Fig. 1, the procedure for obtaining a watermark from a feature and a secret key consists of two steps.

Step 1: Extract features from the original image.

This step is identical to step 1 in the previous subsection.

Step 2: Extract a watermark.

A watermark $\boldsymbol{\xi}$ can be obtained by the Hadamard product of a feature $\boldsymbol{\eta}$ and a secret key \mathbf{W} . That is, it is calculated by

$$\xi_i = W_i \eta_i, \quad i = 1, 2, \dots, K. \quad (4)$$

Thus, the zero-watermarking method can map a watermark to a feature derived from the original image by generating a secret key. Because no watermark is embedded in the image, i.e., the image itself is not modified, no degradation in image quality occurs. However, the zero-watermarking method cannot extract the watermark if the original image has been degraded. Therefore, if compression, noise addition, filtering, or other attacks are applied to the image, the extracted feature might not coincide with the original, and errors might occur in the watermark. In other words, the method lacks robustness against attacks. In addition, because the Hadamard product is used to generate the secret key, there is a restriction that the bit length of the features and the watermark must be the same. Furthermore, as the number of images increases, the number of secret keys also increases. Therefore, it becomes difficult to manage the pairing between the features and the secret keys.

2.3 Amount of information required to store

Let us calculate the amount of information required to store the secret keys, C_z , in the zero-watermarking method. The K -bit secret key must be stored. If P secret keys are to be stored, PK bits of information must be stored. In addition, information about the mapping of images to secret keys must be stored. Note that the need for this mapping information is not considered in conventional zero-watermarking methods.^{8–10} We assume that an index of $L = \log_2 P$ bits is appended to both the image and the secret key to manage them. The index allows us to find the secret key corresponding to the image. This management information requires $2LP$ bits. Thus, the total information to be stored is $C_z = PK + 2LP = P(K + 2 \log_2 P)$ bits. However, if the use of the index is allowed, the watermark can be indexed directly. It can be easily mapped to the image. As a result, it is no longer necessary to implement the zero-watermarking method. The zero-watermarking method does not support the management of multiple images.

3. Associative Memory Models

There are two types of associative memory models: the hetero-associative memory model¹¹ and the auto-associative memory model.^{12–14}

3.1 Hetero-associative memory model

The HMM stores mappings from key patterns to associative patterns. Therefore, when a key pattern is given to the input layer, the corresponding associative pattern is recalled in the output layer. Let $\boldsymbol{\eta}^\mu$ be the μ -th key pattern and let $\boldsymbol{\xi}^\mu$, $\mu = 1, 2, \dots, P$ be the μ -th associative pattern, where P is the number of patterns. The bit length of the key pattern is K bits, and that of the associative pattern is N bits. That is, the key pattern and the associative pattern are represented by $\boldsymbol{\eta}^\mu = (\eta_1^\mu, \eta_2^\mu, \dots, \eta_K^\mu)^T$ and $\boldsymbol{\xi}^\mu = (\xi_1^\mu, \xi_2^\mu, \dots, \xi_N^\mu)^T$, respectively. We assume that each component of the key and associative pattern takes the value ± 1 with equal probability and is given by

$$\text{Prob}[\eta_i^\mu = \pm 1] = \frac{1}{2}, \quad (5)$$

$$\text{Prob}[\xi_i^\mu = \pm 1] = \frac{1}{2}. \quad (6)$$

The synaptic weight W_{ik}^h of the HMM is given by

$$W_{ik}^h = \frac{1}{N} \sum_{\mu=1}^P \xi_i^\mu \eta_k^\mu. \quad (7)$$

As an input $\mathbf{y} = (y_1, y_2, \dots, y_K)^\top$ is given to the HMM, the output of the HMM, $\mathbf{x}^0 = (x_1^0, x_2^0, \dots, x_N^0)^\top$, is given by

$$x_i^0 = \text{sgn}(h_i), \quad (8)$$

where h_i is the internal state of the neuron and is defined as

$$h_i = \sum_{k=1}^K W_{ik}^h y_k. \quad (9)$$

Here, we define the similarity between the input \mathbf{y} and the ν -th key pattern $\boldsymbol{\eta}^\nu$ as overlap given by

$$m_*^\nu = \frac{1}{K} \sum_{k=1}^K \eta_k^\nu y_k. \quad (10)$$

The internal state given by (9) can be expressed as

$$h_i = \gamma m_*^\nu \xi_i^\nu + z_i^*, \quad (11)$$

where $\gamma = \frac{K}{N}$ and z_i^* are the crosstalk noise term, given by

$$z_i^* = \frac{1}{N} \sum_{k=1}^K \sum_{\mu \neq \nu}^P \xi_i^\mu \eta_k^\mu y_k. \quad (12)$$

The crosstalk noise z_i^* is subject to a Gaussian distribution with mean 0 and variance σ_*^2 when the large system limit $N \rightarrow \infty$, $K \rightarrow \infty$, $P \rightarrow \infty$ hold and both the loading rate $\alpha = \frac{P}{N}$ and γ are finite.¹¹⁾ The variance σ_*^2 given by

$$\sigma_*^2 = E[(z_i^*)^2] = \alpha \gamma. \quad (13)$$

The overlap m_0^μ between the output \mathbf{x}^0 of the HMM and the μ -th associative pattern $\boldsymbol{\xi}^\mu$ is defined by

$$m_0^\mu = \frac{1}{N} \sum_{i=1}^N \xi_i^\mu x_i^0. \quad (14)$$

In the large system limit, the overlap m_0^μ is given by

$$m_0^\mu = \text{erf}\left(\frac{\gamma m_*^\mu}{\sqrt{2\sigma_*^2}}\right), \quad (15)$$

where $\text{erf}(x)$ is the error function. See Appendix A.1 for the derivation of (15).

3.2 Auto-associative memory model

The AMM stores multiple patterns and can recall the most similar pattern when a pattern is input to the network. If the AMM stores P patterns $\boldsymbol{\xi}^\mu = (\xi_1^\mu, \xi_2^\mu, \dots, \xi_N^\mu)^\top$, $\mu = 1, 2, \dots, P$, the synaptic weight W_{ij}^a is given by

$$W_{ij}^a = \frac{1}{N} \sum_{\mu=1}^P \xi_i^\mu \xi_j^\mu. \quad (16)$$

The state x_i^{t+1} of the i -th neuron at time $t+1$ is given by

$$x_i^{t+1} = \text{sgn}(h_i^t), \quad (17)$$

where h_i^t is the internal state of the neuron at time t , given by

$$h_i^t = \sum_{j \neq i}^N W_{ij}^a x_j^t. \quad (18)$$

The overlap m_t^μ between the state \mathbf{x}^t of the neurons in the AMM at time $t = 0, 1, 2, \dots$ and the μ -th pattern $\boldsymbol{\xi}^\mu$ is defined as,

$$m_t^\mu = \frac{1}{N} \sum_{i=1}^N \xi_i^\mu x_i^t. \quad (19)$$

The overlap at time $t = 0$ is the same as (14) and is called the initial overlap. Now, assuming that the μ -th pattern is recalled at time t , from Eq. (19), the internal state (18) can be expressed as,

$$h_i^t = m_t^\mu \xi_i^\mu + z_i^t, \quad (20)$$

where the crosstalk noise term z_i^t is given by

$$z_i^t = \frac{1}{N} \sum_{j \neq i}^N \sum_{\nu \neq \mu}^P \xi_i^\nu \xi_j^\nu x_j^t. \quad (21)$$

In the large system limit, the crosstalk noise term z_i^t follows a Gaussian distribution with mean 0 and variance σ_t^2 , that is,

$$\sigma_t^2 = E[(z_i^t)^2]. \quad (22)$$

The overlap m_{t+1}^μ and the variance σ_{t+1}^2 are given by

$$m_{t+1}^\mu = \text{erf}\left(\frac{m_t^\mu}{\sqrt{2\sigma_t^2}}\right), \quad (23)$$

$$\sigma_{t+1}^2 = \alpha + U_{t+1}^2 \sigma_t^2 + 2\alpha \sum_{\tau=t-n+1}^t q_{t+1,\tau} \prod_{k=\tau+1}^{t+1} U_k. \quad (24)$$

Because AMM is a recurrent neural network, the variance of the crosstalk noise has temporal correlations.^{11,14)} See Appendix A.2 for the derivation of (23) and (24).

4. Associative Watermarking Method

The AWM¹⁵⁾ introduces HMM for mapping features to watermarks in the zero-watermarking method and AMM for watermark error correction. When mapping multiple images and watermarks, the μ -th feature $\boldsymbol{\eta}^\mu$ and watermark $\boldsymbol{\xi}^\mu$ in the zero-watermarking method correspond to the key pattern $\boldsymbol{\eta}^\mu$ and the associative pattern $\boldsymbol{\xi}^\mu$ in the HMM, respectively. Also, all secret keys \mathbf{W}^μ are represented by a single synaptic weight \mathbf{W}^h . Therefore, the AWM has several advantages compared to the zero-watermarking method. First, it does not need to manage the mapping between many secret keys and features—only the synaptic weight. Second, errors in the watermark can be corrected even if a feature is degraded. Third, the bit length of the feature $\boldsymbol{\eta}$ and that of the watermark $\boldsymbol{\xi}$ may be different. Fourth, the AWM has the same structure as the HASP,^{11,16)} as shown in Fig. 2, and the retrieval process and memory capacity can be analyzed by using statistical neurodynamics.¹¹⁾

4.1 Proposed model

Let $t = -1$ be the time when the input \mathbf{y} is given to the HMM, and let $t = 0$ be the time when the output \mathbf{x}^0 of the HMM is input to the AMM.

Suppose that there are P images and that the N -bit watermark $\boldsymbol{\xi}^\mu = (\xi_1^\mu, \xi_2^\mu, \dots, \xi_N^\mu)^\top$ is associated with the μ -th image. Similar to the zero-watermarking method,⁸⁾ K low-frequency components are extracted from the DCT coefficients of the μ -th image, excluding the DC components. The extracted DCT coefficients are binarized by (1) to obtain the features $\boldsymbol{\eta}^\mu = (\eta_1^\mu, \eta_2^\mu, \dots, \eta_K^\mu)^\top$.

The features $\boldsymbol{\eta}^\mu$ and watermarks $\boldsymbol{\xi}^\mu$ are regarded as key patterns and associative patterns, respectively. The synaptic weights \mathbf{W}^h and \mathbf{W}^a are given by (7) and (16), respectively.

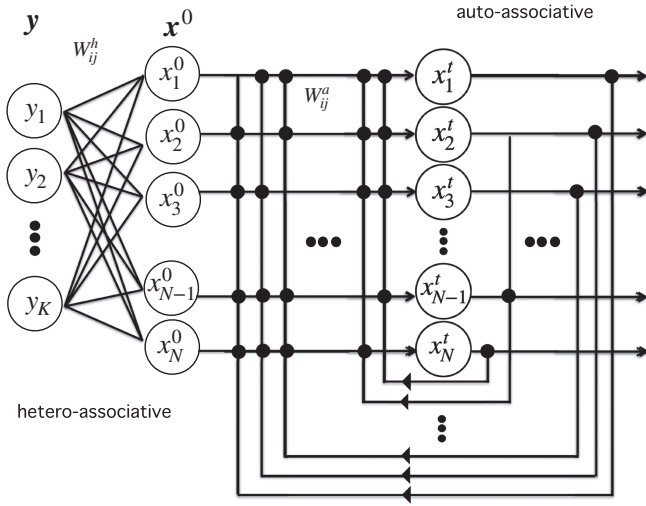


Fig. 2. Structure of the associative watermarking memory.

The mapping between features and watermarks can be managed by storing the weight \mathbf{W}^h , and the watermarks can be recalled almost without error by storing the weights \mathbf{W}^a .

Suppose that the μ -th image is attacked and that a feature is degraded. In the HMM, the sum of the products of the degraded feature $\tilde{\eta}^\mu = (\tilde{\eta}_1^\mu, \tilde{\eta}_2^\mu, \dots, \tilde{\eta}_K^\mu)^\top$ and the synaptic weight \mathbf{W}^h is calculated by (9) to obtain the output \mathbf{x}^0 . This output \mathbf{x}^0 is given as the initial value of the AMM. The state of the neurons in the AMM at each time is determined by (17). The watermark ξ^μ is known to be successfully recalled if the overlap between the state of the neurons \mathbf{x}^t and the watermark ξ^μ at time t is $m_t^\mu = 1$ after sufficient time.

4.2 The recall process and basin of attraction

The AWM corresponds to the one-to-one association case of the HASP.¹¹⁾ The recall process and memory capacity of this model were analyzed.^{11,17)} We evaluated the performance of our method using the macroscopic state equations derived in A.1 and A.2 and using computer simulations.

First, we evaluated the recall process of watermarks from degraded features. Figure 3 shows the watermark recall process. The horizontal axis represents time t , and the vertical axis represents the overlap. Here, the overlap at time $t = -1$ represents the overlap m_{-1}^μ for the feature η^μ , and the overlap at $t = 0, 1, 2, \dots$ represents the overlap m_t^μ for the watermark ξ^μ . The solid line represents the theoretical value of overlap represented by (15) and (23). The cases recalled from various initial overlaps are shown. In the computer simulation, the

features and watermarks were randomly generated by (5) and (6). The degraded figures input to the AWM were flipped to be the given initial overlap. The bit lengths of the features and watermark were set to $N = 10000$, $K = 10000$, respectively. That is, $\gamma = 1.0$. The overlap at each time was illustrated by an error bar. The error bars represent the mean and the standard deviation for 100 trials. (a) and (b) show the results when the number of patterns was $P = 800$ ($\alpha = 0.08$) and $P = 1200$ ($\alpha = 0.12$), respectively.

Figure 3 shows the overlap m_0^μ of the output layer of the HMM was larger than the overlap m_{-1}^μ of the input layer. This means that the HMM can not only store the mapping between features and watermarks but also correct watermark errors. The recallable boundary value is called the critical overlap m_c . For $\alpha = 0.08$ in Fig. 3(a), the critical overlap was $m_c > 0.2$ for the fourth-order theory ($n = 4$). Computer simulations also showed that $m_c > 0.2$. Similarly, for $\alpha = 0.12$ in Fig. 3(b), both theory and computer simulations showed $m_c > 0.3$. These results revealed that fourth-order theory can quantitatively represent the results of computer simulations.

Figure 4 shows the basins of attraction to evaluate the error correction capability of AWM. (a) is a case where the bit lengths of the features and watermark were $K = 5000$, $N = 10000$, i.e., $\gamma = 0.5$, respectively, and (b) is a case where $K = N = 10000$ ($\gamma = 1.0$). The horizontal axis represents the loading rate α , and the vertical axis represents the overlap. The lower curves show the critical overlaps m_c . The upper curves show the equilibrium overlaps m_∞ when the network was in equilibrium at time $t \rightarrow \infty$. The equilibrium overlap represents the stability of the memorized pattern and is the overlap at time $t \rightarrow \infty$ when the network is recalled from an initial state with initial overlap $m_0 = 1$. The region bounded by the equilibrium overlap and the critical overlap is the basin of attraction.

We can see from Fig. 3 that the overlaps were well converged by $t = 20$. Therefore, the equilibrium overlaps obtained from the computer simulations were the overlap m_{20} at $t = 20$. The error bars were the results obtained from the computer simulations, and they represent the mean and standard deviation of the overlap m_{20}^μ over 100 trials at each loading rate α . The black and red dashed lines are the results obtained from the fourth-order theory for AMM and AWM, respectively. The results of the computer simulations were in good agreement with the theoretical curves of the AWM. In Fig. 4(a), the basin of attraction of AWM was lower than that of AMM, while the basin of attraction of AWM was higher in

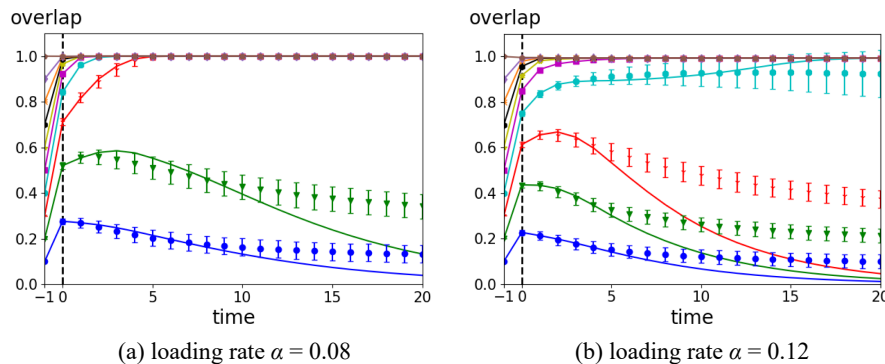


Fig. 3. (Color online) Time evolution of overlaps m_{-1}^μ, m_t^μ for the loading rates $\alpha = 0.08, 0.12$ and $\gamma = 1.0$.

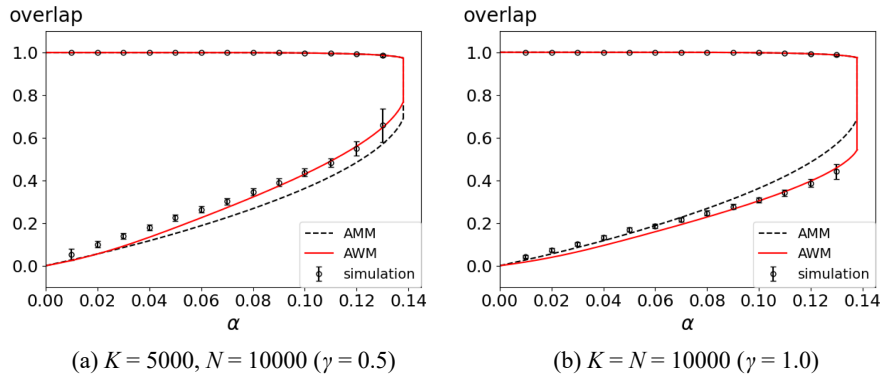


Fig. 4. (Color online) Basin of attraction for the AMM and AWM.

(b). As the bit length ratio of the features and watermarks, $\gamma = K/N$, decreases, the overlap of the HMM's output decreases from (15). In other words, when γ is low, the error correction capability of the HMM is low.

4.3 Amount of information required to store

Let us calculate the amount of information required to store the secret keys, C , in AWM. Information on the weight matrices W_{ik}^h and W_{ij}^a must be stored. If the activation function in (8) and (17) is $\text{sgn}()$, the division by N operation in (7) and (16) can be omitted. Thus, the weights can be stored as integers. The maximum value of W_{ik}^h and W_{ij}^a is P if the values of the patterns are $\eta_k^\mu = 1$ and $\xi_i^\mu = 1$ for all μ . In practice, the maximum is less than P because this case is unlikely. It takes $\log_2 P$ bits to store the value of a weight. Since W_{ik}^h is an asymmetric matrix, $C_h = NK \log_2 P$ bits are required to store this weight. In addition, since W_{ij}^a is a symmetric matrix and the diagonal components are not used, $C_a = (N(N-1)/2) \log_2 P$ bits are required. Accordingly, the total information to be stored is $C = C_h + C_a = N(K + (N-1)/2) \log_2 P$ bits. Let us compare the zero-watermarking method and AWM in terms of the amount of information. Obviously, the zero-watermarking method requires less information than AWM. Note, however, that the proposed AWM does not require indexing. In other words, the proposed method is suitable for managing multiple images.

5. Performance Evaluation of the AWM

The order parameter equations are equivalent to those of the HASP.¹¹⁾ The previous section presented an evaluation of the performance of AWM with the basin of attraction as an AMM. This section features an evaluation of its performance as a watermarking model. It was evaluated quantitatively in terms of the bit error rate (BER) of the watermark. The BERs obtained from the zero-watermarking method, HMM, and AWM were evaluated using features obtained from images. The BER is defined by

$$\text{BER}(m) = \frac{1-m}{2}, \quad (25)$$

where m is an overlap. Therefore, the BER for the μ -th watermark at time t is represented by $\text{BER}(m_t^\mu)$.

5.1 JPEG compression attack

We evaluated the BERs of the watermarks when the images were JPEG compressed and when the degraded

features were extracted. JPEG compression was applied to 1200 original images¹⁸⁾ of 512×512 pixels. The bit lengths of the features were set to $K = 5000, 10000$. The watermarks were randomly generated by (6). The bit length of the watermarks was set to $N = 10000$. That is, $\gamma = 0.5, 1.0$ and $\alpha = 0.12$. With weak JPEG compression, the watermark could be recalled correctly due to the error correction capability of the HMM. Therefore, we examined the case of strong compression, where the watermark was incorrectly recalled at the output of the HMM. Here, the quantization level of JPEG compression was set to $Q = 5$.

The BERs of the watermarks are shown in Fig. 5 when the images were JPEG compressed. The horizontal and vertical axes represent the overlap m_*^μ of the feature η^μ and the $\text{BER}(m_t^\mu)$ of the watermark ξ^μ , respectively. Because the degraded feature $\tilde{\eta}^\mu$ was given as input y of the HMM, the overlap m_*^μ was calculated using the (10). The blue and red solid lines represent, respectively, the theoretical BER for the output of the HMM obtained by the (15) and the output of the AWM at time $t = 20$ obtained by the (23). The black, blue, and red points represent the BERs of the watermarks obtained from the zero-watermarking model, the HMM, and the AWM obtained from computer simulations. The overlaps for AWM were the value of m_{20}^μ at time $t = 20$. The BERs were plotted using 12 images compressed at a high compression ratio ($Q = 5$) so as to generate errors in the output of the HMM. Note that it is difficult to obtain a small overlap m_*^μ even with a high compression rate. Figure 5(a) shows the results for bit lengths of $K = 5000, N = 10000$ ($\gamma = 0.5$) and (b) for those of $K = N = 10000$ ($\gamma = 1.0$), where the loading rate was $\alpha = 0.12$. The theoretical curves showed that the BER of the HMM varied gradually, while the BER of the AWM varied sharply at the critical overlap m_c , owing to the AMM. Because of their error correction capabilities, both the HMM and AWM had smaller errors than the zero-watermarking method. In addition, errors that could not be corrected by the HMM could be corrected by the AMM. The error was almost zero for the AWM in total. The overlap for the HMM appeared to be out of the theoretical curve because the features extracted from the images did not satisfy the condition (5).

5.2 Gaussian noise attack

Next, we show the results for additive Gaussian noise, since the JPEG compression attack did not degrade the watermarks below the critical overlap. Gaussian noise with a

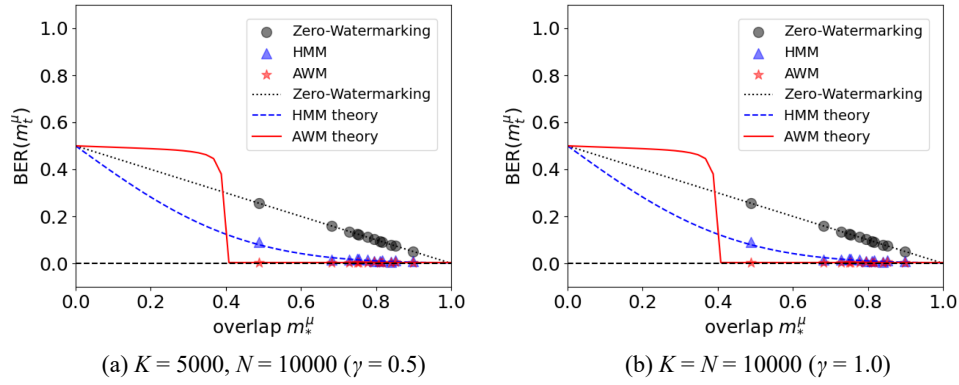


Fig. 5. (Color online) BER of the zero-watermarking method, HMM, and AWM for JPEG compression attack ($\alpha = 0.12$).

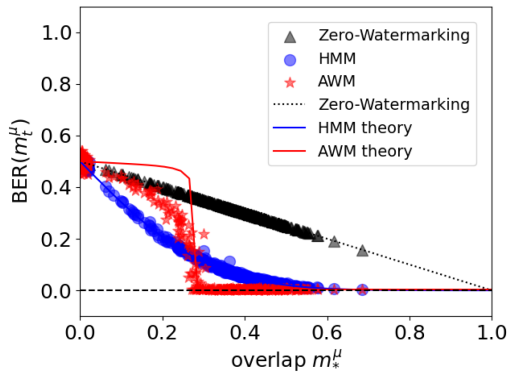


Fig. 6. (Color online) BER for Gaussian noise attack ($\alpha = 0.12, \gamma = 2.0$).

mean of 100 and a standard deviation of 100 was used to generate large errors. Although pixel values usually take 256 gray levels, we assume that the pixel values in this experiment take values beyond that range. Figure 6 shows the BER in the case of a Gaussian noise attack, where $K = 20000$, $N = 10000$ ($\gamma = 2.0$), and $P = 1200$ ($\alpha = 0.12$). There are differences in BER for AWM below the critical overlap because theory and computer simulations do not match below the critical overlap, as shown in Fig. 3. However, they are in good agreement above the critical overlap. If the image is highly degraded and the overlap of the watermark is below the critical overlap (which is rarely the case), only the HMM performs better because the BER of the HMM is lower than that of the AWM.

6. Conclusion

While the zero-watermarking method does not degrade the original image, it cannot correct watermark errors. Also, managing the mapping between features and secret keys becomes difficult because the number of secret keys increases as the number of images increases. However, the mapping between the features and secret keys in the associative watermarking model can be managed collectively by simply storing synaptic weights. In addition, the features and watermarks have no restrictions on the bit length. Furthermore, even if the image is attacked, the watermark can be corrected and extracted.¹⁵⁾ In this paper, we presented an evaluation of the BER in the AWM through theory and computer simulations. The macroscopic state equations for

the AWM were equivalent to the one-to-one associative case of the HASP, and the retrieval process and storage capacity were calculated using statistical neurodynamics. The retrieval processes obtained from fourth-order theory and those from computer simulations were in good agreement. The storage capacity of the AWM was the same as that of the AMM. We also found that the basin of attraction for the AWM varied in size depending on the ratio $\gamma = K/N$ of the bit lengths of the features and the watermark.

Actual images were usually JPEG compressed, and the BER of the extracted watermark was evaluated to assess the effectiveness of AWM as a watermarking model. The results showed that the error rates of both the HMM and AWM were less than those of conventional zero-watermarking. Furthermore, the BERs of the AWM were almost zero. As a result, our method extends the capabilities of the zero-watermarking method and has excellent functions. Although we discussed the JPEG compression and additive Gaussian noise attacks, the results would be the same if other image processing were applied. The performance was determined by the overlap at the input to the HMM.

When processing multiple images, the independence of the features must be taken into account. For example, in the case of chest X-ray images, since they are all similar images, they are highly correlated with each other. The extraction of independent features from such correlated images is an open problem.

Acknowledgment This work was supported by JSPS KAKENHI Grant Numbers 20K11973 and 24K15106. The computations were performed using the supercomputer facilities at the Research Institute for Information Technology, Kyushu University.

Appendix: Order Parameter Equations

To analyze the dynamics of the recall process in the associative memory model, we derived a macroscopic state equation for this model using statistical neurodynamics.^{13,14)} This theory describes the state of the network using multiple macroscopic state variables, and the recall process is represented by the macroscopic state equations. Okada's theory¹⁴⁾ considers temporal correlations of states up to n previous time. In this paper, we call it n -order theory. For $n = 1$, it is equivalent to the Amari–Magainu theory.¹³⁾ The storage capacity converges to the results obtained from the equilibrium state theory¹⁹⁾ as the order n increases.

A.1 Macroscopic state equations for the HMM

We herein discuss the case where the μ -th pattern is recalled. The superscript for μ is omitted. Macroscopic state equations are derived for the HMM. From (12), the mean of the crosstalk noise term at time $t = -1$ is 0, and its variance is given by

$$\sigma_*^2 = E[(z_i^*)^2] = \alpha\gamma, \quad (\text{A}\cdot 1)$$

where $\alpha = P/N$ and where $\gamma = K/N$. The macroscopic state equations at time $t = 0$ are given by

$$m_0 = \int D_z \langle \xi \operatorname{sgn}(\gamma \xi m_* + \sigma_* z) \rangle_\xi, \quad (\text{A}\cdot 2)$$

$$\sigma_0^2 = \alpha + \sigma_*^2 U_0^2, \quad (\text{A}\cdot 3)$$

$$U_0 = \frac{1}{\sigma_*} \int D_z z \langle \operatorname{sgn}(\gamma \xi m_* + \sigma_* z) \rangle_\xi, \quad (\text{A}\cdot 4)$$

where $\langle \cdot \rangle_\xi$ represents the average over μ -th pattern ξ^μ and where D_z is defined by

$$D_z = \frac{dz}{\sqrt{2\pi}} \exp\left(-\frac{z^2}{2}\right). \quad (\text{A}\cdot 5)$$

These equations can be solved as

$$m_0 = \operatorname{erf}\left(\frac{\gamma m_*}{\sqrt{2}\sigma_*}\right), \quad (\text{A}\cdot 6)$$

$$U_0 = \sqrt{\frac{2}{\pi\sigma_*^2}} \exp\left(-\frac{\gamma^2 m_*^2}{2\sigma_*^2}\right). \quad (\text{A}\cdot 7)$$

A.2 Macroscopic state equations for the AMM

The macroscopic state equations for the AMM were derived by Okada.¹⁴⁾ The n -order macroscopic state equations at time $t \geq 0$ are given as follows.

$$m_{t+1} = \int D_z \langle \xi \operatorname{sgn}(\xi m_t + \sigma_t z) \rangle_\xi, \quad (\text{A}\cdot 8)$$

$$\sigma_{t+1}^2 = \alpha + U_{t+1}^2 \sigma_t^2 + 2\alpha \sum_{\tau=t-n+1}^t q_{t+1,\tau} \prod_{k=\tau+1}^{t+1} U_k, \quad (\text{A}\cdot 9)$$

$$U_{t+1} = \frac{1}{\sigma_t} \int D_z z \langle \operatorname{sgn}(\xi m_t + \sigma_t z) \rangle_\xi, \quad (\text{A}\cdot 10)$$

$$q_{t+1} = \int D_z \langle \operatorname{sgn}(\xi m_t + \sigma_t z)^2 \rangle_\xi. \quad (\text{A}\cdot 11)$$

These equations can be solved as

$$m_{t+1} = \operatorname{erf}\left(\frac{m_t}{\sqrt{2}\sigma_t}\right), \quad (\text{A}\cdot 12)$$

$$U_{t+1} = \sqrt{\frac{2}{\pi\sigma_t^2}} \exp\left(-\frac{m_t^2}{2\sigma_t^2}\right), \quad (\text{A}\cdot 13)$$

$$q_{t+1} = 1. \quad (\text{A}\cdot 14)$$

The $q_{t+1,\tau}$ represents the correlation between the state x_{t+1} at time $t+1$ and the state x_τ at time τ , and it is defined as

$$q_{t+1,\tau} = E[x_{t+1}^{t+1} x_\tau^\tau]. \quad (\text{A}\cdot 15)$$

This variable can be expressed as

$$q_{t+1,\tau} = \int D_c \int D_a \int D_b \langle \operatorname{sgn}(\xi m_t + \sigma_t(d_0 a + d_1 c)) \times \operatorname{sgn}(\xi m_{\tau-1} + \sigma_{\tau-1}(d_0 b + d_1 c)) \rangle_\xi, \quad (\text{A}\cdot 16)$$

where

$$d_0 = \sqrt{1 - \frac{C_{t,\tau-1}}{\sigma_t \sigma_{\tau-1}}}, \quad d_1 = \sqrt{\frac{C_{t,\tau-1}}{\sigma_t \sigma_{\tau-1}}}, \quad (\text{A}\cdot 17)$$

and where $C_{t,\tau-1}$ denotes the correlation $C_{t,\tau-1} = E[z_t z_{\tau-1}]$ between the crosstalk noise term z_t at time t and the term $z_{\tau-1}$ at time $\tau-1$. These correlations can be expanded as

$$C_{t,\tau-1} = 0 \quad (\tau = t - n + 1, n \geq 1), \quad (\text{A}\cdot 18)$$

$$C_{t,\tau-1} = \alpha q_{t,\tau-1} + U_t C_{t-1,\tau-1} \quad (\tau = t - n + 2, n \geq 2), \quad (\text{A}\cdot 19)$$

$$C_{t,\tau-1} = \alpha q_{t,\tau-1} + U_t U_{\tau-1} C_{t-1,\tau-2} + \alpha \sum_{\eta=\tau-n+1}^{\tau-2} q_{t,\eta} \prod_{k=\eta+1}^{\tau-1} U_k + \alpha \sum_{\eta=\tau-n+1}^{\tau-1} q_{\eta,\tau-1} \prod_{k=\eta+1}^t U_k \quad (t - n + 3 \leq \tau \leq t, n \geq 3). \quad (\text{A}\cdot 20)$$

A.3 Temporal correlation between the HMM and the AMM

The AWM consists of both the HMM and the AMM. Therefore, we need to consider the temporal correlation between the input ($t = -1$) of the HMM and the state of the AMM at time t . However, the input at time $t = -1$ is the feature η^μ and the state at time t is the state recalling the watermark ξ^μ . Because the feature and the watermark are uncorrelated, the temporal correlation between the input and the state can be ignored. Therefore, the correlation of states at time $t = -1$ can be $q_{t+1,-1} = 0$.

*kawamura@sci.yamaguchi-u.ac.jp

- 1) F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, *Proc. IEEE* **87**, 1062 (1999).
- 2) S. P. Mohanty, Digital Watermarking: A Tutorial Review. <https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=0463206e28b320e747e6dc34f2cca8829f0c4005> (1999) (accessed 2024-08-09).
- 3) A. Mohanarathinam, S. Kamalraj, G. K. D. Prasanna Venkatesan, R. V. Ravi, and C. S. Manikandababu, *J. Ambient Intell. Humaniz. Comput.* **11**, 3221 (2020).
- 4) M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, *IEEE Trans. Image Process.* **14**, 253 (2005).
- 5) I. J. Cox, J. Kilian, T. Leighton, and T. Shamoan, *Proc. 3rd IEEE Int. Conf. Image Processing, 1996, Vol. 3*, p. 243.
- 6) I. J. Cox, J. Kilian, F. T. Leighton, and T. Shamoan, *IEEE Trans. Image Process.* **6**, 1673 (1997).
- 7) H. Luo, X. Sun, H. Yang, and Z. Xia, *Radioengineering* **20** [2], 525 (2011).
- 8) C. Dong, H. Zhang, J. Li, and Y.-W. Chen, 2011 6th Int. Conf. Computer Sciences and Convergence Information Technology (ICCIT), 2011, p. 900.
- 9) A. Rani, A. K. Bhullar, D. Dangwal, and S. Kumar, *Procedia Comput. Sci.* **70**, 603 (2015).
- 10) H. D. Kim, *Int. J. Contents* **11**, 39 (2015).
- 11) M. Kawamura, M. Okada, and Y. Hirai, *IEEE Trans. Neural Networks* **10**, 704 (1999).
- 12) J. J. Hopfield, *Proc. Natl. Acad. Sci. U.S.A.* **79**, 2554 (1982).
- 13) S. Amari and K. Maginu, *Neural Networks* **1**, 63 (1988).
- 14) M. Okada, *Neural Networks* **8**, 833 (1995).
- 15) R. Kanegae and M. Kawamura, 2022 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), 2022, p. 1376.
- 16) Y. Hirai, *IEEE Trans. Syst. Man Cybern.* **SMC-13**, 851 (1983).
- 17) M. Kawamura and Y. Hirai, *Syst. Comput. Jpn.* **28** [1], 24 (1997).
- 18) T. Pathak, Natural Images with Synthetic Noise, <https://www.kaggle.com/dsv/1958494> (2021) (Accessed 2023-09-26).
- 19) D. J. Amit, H. Gutfreund, and H. Sompolinsky, *Phys. Rev. Lett.* **55**, 1530 (1985).