Indonesian Journal of Electrical Engineering and Computer Science
Vol. 33, No. 3, March 2024, pp. 1536∼1545
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v33.i3.pp1536-1545

Arnold's cat map secure multiple-layer reversible watermarking

Aulia Arham, Novia Lestari
Department of Information Systems, Faculty of Science and Technology, Universitas Islam Negeri Imam Bonjol, Padang, Indonesia

Article Info

Article history:

Received Oct 16, 2023
Revised Dec 21, 2023
Accepted Jan 6, 2024

Keywords:

Arnold's cat map
Copyright protection
Multiple-layer reversible
Reversible watermarking
Secure digital

ABSTRACT

Reversible watermarking is a novel approach to digital copyright protection that allows the embedding of watermarks into digital data using multiple layers while retaining the ability to recover the original content without data loss. This method provides a unique solution for securing digital data while maintaining the integrity and quality of the content. Nonetheless, new challenges have emerged with the increase in attacks on this method, as reversible watermarking methods lack security keys, making it easy to extract and modify hidden data. In this paper, we present a method for multiple-layer reversible watermarking with security keys, with the goal of addressing the challenges posed by attacks and improving data protection within embedded content. The method uses arnold's cat map to scramble images, and data embedding in predetermined iterations serves as the methods security key. We put the method through its paces with six grayscale images. With this method, the embedding capacity can reach 2.999 bpp across four layers of embedding, while the visual image quality can reach 22.01 dB. The outcomes from this approach are that the security of multiple-layer reversible watermarking can be enhanced while preserving the capacity to embed data in each layer.

Corresponding Author:

Aulia Arham
Department of Information Systems, Faculty of Science and Technology
Universitas Islam Negeri Imam Bonjol
Balai Gadang, Koto Tangah, Padang, West Sumatra, Indonesia
Email: auliaarham@uinib.ac.id

1. INTRODUCTION

In our rapidly evolving digital age, ensuring the security of copyrights and the preservation of digital data integrity have gained increasing significance. One approach used to achieve these goals involves the use of watermark methods, as described in [1], [2]. A watermark, in this context, refers to a symbol or piece of information that is embedded into an image or digital document to establish ownership or validate its authenticity. Watermarking in sensitive images such as military, artwork, and medical images must be approached with great care because of the significance of every piece of information [1]–[6]. One of the methods used is reversible watermarking. Reversible watermarking enables us to embed watermarks into digital data without sacrificing the original information [1]–[10]. In other words, we can add watermarks to an image or digital document without compromising the underlying data and recover the original image without any distortion.

In recent years, reversible watermarking has garnered significant research attention because of its ability to reversibly restore the original image. The difference expansion (DE) method, which Tian [4] invented, is an illustration of straightforward reversible watermarking. This method involves embedding data bits into the disparities between pairs of pixels. These disparities are partitioned into eight-bit planes, with the data bits embedded in the least significant bit. This approach further amplifies the differences between the pixel pairs. Over the last few years, multiple reversible watermarking methods have been rapidly developed, all rooted in the DE method concept. This development focuses on improving the performance of DE methods, such as embedding capacity, visual quality, and complexity.

Numerous researchers have introduced various methods to enhance image quality. For example, Liu *et al.* [5] introduced the concept of reduced difference expansion (RDE), whereas Yi *et al.* [11] presented the improved reduced difference expansion (IRDE). Both of these methods were successful in improving the image quality. In contrast, Abdullah and Manaf [12] suggested embedding data in smooth regions using the DE method, and Maniriho and Ahmad [13] proposed data embedding in smooth areas using the DE method along with the modulus function. These approaches also improved the image quality, although they reduced the embedding capacity. To increase the embedding capacity, researchers have devised various methods to alter the block sizes used. For instance, Alattar introduced methods [14], [15] using blocks of sizes 1×3 and 2×2 with a pixels being the based point. Hsiao *et al.* [16] employed blocks of size 3×3, Based on two predetermined thresholds, all blocks are categorized into three groups: smooth, normal, or complex. Complex blocks are excluded from data embedding, whereas a smooth block will carry twice the amount of data as a normal block, Ahmad *et al.* [17] used 2×2 blocks and reduced the difference value of the pixel with the RDE method, Arham *et al.* [3] used 2×2 blocks and reduced the difference value of the pixel with the IRDE method, Al Huti *et al.* [18] employed 4×5 blocks and reduced the difference value with a modified RDE, and Syahlan and Ahmad [2] used 2×2 blocks and reduced the difference value of the pixel with a modified RDE. All of these outcomes demonstrated that this approach expanded the data embedding and resulted in good visual quality. Various alternative reversible data embedding methods based on the DE method have also been introduced in different sources [19]–[31].

The capability for reversible watermarking, which enables data to be restored to its original state, permits the use of layered embedding to enhance the capacity of embedding [23]. The concept of multi-layer embedding using the DE method began with Tian [4], who initially focused on the DE method alone. Subsequently, Lou *et al.* [32] introduced the idea of merging the DE and RDE schemes. Yi *et al.* [11] expanded upon this by introducing a multi-layer embedding approach that combines DE with IRDE. Furthermore, Wang *et al.* [33], Pan *et al.* [34] presented multi-layer embedding techniques that use a histogram shifting scheme. Arham *et al.* [3] introduced a reversible watermarking method that used embedding data with a multiple-layer approach, incorporating both the IRDE and the DE of Quad methods. Lee *et al.* [35] proposed multiple-layer reversible watermarking with an improved method of [3] by expanding the size of the block used according to the scanning strategy. Maniriho *et al.* [1] introduced a novel approach to multi-layer reversible watermarking for medical images. Their proposal involves a fresh, uncomplicated data embedding (DE) method and a block-based reversible multiple-layer data hiding method. Mehbodniya *et al.* [24] proposed multiple-layer reversible watermarking based on DE using multiple-layer thresholding. The multiple-level thresholding technique uses a combination of the smile mould algorithm (SMA) and the Otsu evaluation function.

Nevertheless, security concerns persist as a consideration in these reversible data embedding methods. The absence of a security key implies that unauthorized parties can effortlessly access and alter the secret data if they are familiar with the embedding method. To enhance the security of reversible data embedding, numerous researchers have devised diverse tactics. Wu *et al.* [36] proposed embedding encrypted data in encrypted images at specified locations using an inventive RRBE method that leverages APL. Additionally, Meenpal and Majumder [37] introduced a secure method for reversible data embedding, employing the DE technique within the integer wavelet transform (IWT) domain to embed data into low-frequency blocks of an image. Moreover, several other researchers, cited in [38]–[41], have also put forward secure approaches for reversible data embedding, involving the integration of encrypted data into images using histogram shifting methodologies. These aforementioned strategies typically entail the use of encryption for the confidential data to be embedded, thereby reinforcing the security level within the reversible data embedding process.

Nonetheless, reversible watermarking methods do exhibit vulnerabilities, particularly regarding security. In this study, we propose a novel approach to secure multiple-layer reversible watermarking using arnold's cat map as the underlying algorithm. Our objective is to elucidate how this strategy seeks to bolster security in the process of embedding watermarks across diverse layers, thereby introducing complexity for unauthorized entities attempting to access or manipulate the embedded watermark.

This paper is structured into four main sections. In the first section, we introduce the research context and provide an overview of prior studies associated with the difference expansion method. In section 2 provides an extensive elucidation of our proposed approach, covering the embedding process, extraction, and intricacies of the method computations. In section 3 is dedicated to showcasing the experimental outcomes and comparing them with those of the previous method. Finally, section 4 concludes this work.

2. METHOD

This method was proposed to improve the security of embedded data using a secret key in multiple-layer reversible watermarking, particularly those presented by earlier researchers [3]. Secret key used for image obfuscation using arnold's cat map algorithm. The primary contribution of this study lies in enhancing the efficacy of securely embedding data within an image by implementing a secret key for image obfuscation, which involves the utilization of arnold's cat map algorithm.

2.1. Data embedding step

The description below outlines a seven-step embedding process:
Step 1: Select the secret key k to represent the iterations of the arnold cat map for the number of data embedding layers. These iterations k_n should be sequentially defined, starting from 1 and extending until they reach iteration T. The value of T signifies the iteration period of arnold's cat map, and T is less than $3N$, with N representing the images dimensions in an $N \times N$ format. As illustrated, the secret key $k = (5, 25, 56, 95)$ is used for 4 layers of embedding, where $T = 384$, and the image size is $N = 128$.
Step 2: The host image is scrambled using arnold's cat map with the secret key k. This process involves generating new coordinates (x', y') by transforming the existing coordinates (x, y) within an $N \times N$ -sized image. The iteration process is defined by (1). Data embedding occurs at each iteration k_n.

$$\begin{bmatrix} x_{i+1} \\ \hline y_{i+1} \end{bmatrix} = \begin{bmatrix} 1 & b \\ c & bc+1 \end{bmatrix} \begin{bmatrix} x_i \\ \hline y_i \end{bmatrix} \bmod (N) \quad (1)$$

The coordinates (x_i, y_i) signify the positions of pixels in the image that change to (x_{i+1}, y_{i+1}) after the i-th iteration. In this context, b and c represent positive integers. Preserving the transformation within the same image area necessitates the determinant of the matrix $\begin{bmatrix} 1 & b \\ c & bc+1 \end{bmatrix}$ to be equal to 1, indicating that the matrix conserves the area. Arnold's cat map ensures a unique mapping from each pixel to a new pixel position, establishing a one-to-one correspondence. The iteration of arnold's cat map is repeated m times, yielding a different random image with each iteration. The values of b, c, and m, signifying the number of rotations in arnold's cat map, can serve as the secret key. The process involves shifting the image in the y-direction, followed by a shift in the x-direction during each iteration. However, the results might extend beyond the image area; thus, all outcomes are taken modulo N to ensure they remain within the image area, thereby preserving the area. The secret key involves data embedding at different layers during different iterations.
Step 3: During each predetermined iteration, with the aid of the secret key k_n, the image is divided into 2×2 blocks. Subsequently, a pixel vector is generated for each block within each layer to compute the differences between consecutive pixels, as illustrated in Figure 1.

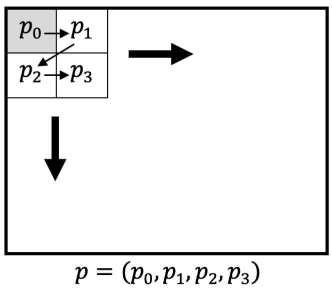

Figure 1. The proposed block of pixels with a base point is (p_0)

Pixels p_0 serves as the reference point, as depicted in Figure 1. This reference point allows us to create vectors $d = (d_1, d_2, d_3)$ which represent the difference values of the vector $p = (p_0, p_1, p_2, p_3)$ and can be expressed as (2).

$$\begin{cases} d_1 = p_1 - p_0 \\ d_2 = p_2 - p_1 \\ d_3 = p_3 - p_2 \end{cases} \tag{2}$$

Step 4: After completing the calculation of the pixel block pair differences to reduce the distortion of the difference values, (3) is employed, where h is computed using (4). The reduction of the difference value was performed if $(-1 > d_i > 1)$ then the location map $(LM_{d_i}) = (0, 1)$ and if $(-1 \leq d_i \leq 1)$ then the location map $(LM_{d_i}) = (-1)$ show in (5). In the extraction phase, the location map (LM) is used to recover the original difference value from the original pixels of the host image.

$$d'_i = \begin{cases} d_i - 2^{\log_2|d_i|-1}, & \text{if } 2 \times 2^{h-1} \leq d_i \leq 3 \times 2^{h-1} - 1 \\ d_i - 2^{\log_2|d_i|}, & \text{if } 3 \times 2^{h-1} \leq d_i \leq 4 \times 2^{h-1} - 1 \end{cases} \tag{3}$$

$$h = \lfloor \log_2 |v_i| \rfloor \tag{4}$$

$$LM_{d_i} = \begin{cases} -1, \text{if } -1 > d_i > 1 \\ 0, \text{if } 2 \times 2^{h-1} \leq d_i \leq 3 \times 2^{h-1} - 1 \\ 1, \text{if } 3 \times 2^{h-1} \leq d_i \leq 4 \times 2^{h-1} - 1 \end{cases} \tag{5}$$

Step 5: To embed a data bit b_i, there are two methods are available, either through (6) or (7). Blocks of pixels that employ (6) to embed data are labelled as expandable and location map $(LM_b) = 1$. In cases where using (6) results in overflow or underflow, the reduction of the difference value was cancelled, and the embedding process was switched to (7) and then labelled as changeable and location map $(LM_b) = 0$. Additionally, blocks are c as unchangeable if they do not fall into the second category and embedding data is not performed then location map $(LM_b) = -1$.

$$\begin{cases} \tilde{d}_1 = 2 \times d'_1 + b_i \\ \tilde{d}_2 = 2 \times d'_2 + b_{i+1} \\ \tilde{d}_3 = 2 \times d'_3 + b_{i+2} \end{cases} \tag{6}$$

$$\begin{cases} d'_1 = 2 \times \left\lfloor \frac{d_1}{2} \right\rfloor + b_i \\ d'_2 = 2 \times \left\lfloor \frac{d_2}{2} \right\rfloor + b_{i+1} \\ d'_3 = 2 \times \left\lfloor \frac{d_3}{2} \right\rfloor + b_{i+2} \end{cases} \tag{7}$$

Each block of pixel has the embedding location map $LM = (LM_1, LM_2, LM_3, LM_4)$, where LM_1 are located to LM_b is categories of the block pixel in expandable, changeable, or non-changeable, and LM_2, LM_3, and LM_4 are located to LM_{d_i} is identify the difference value of the pixel $d = (d_1, d_2, d_3)$ on the block has been reduced. In the case location map $(LM_1) = 0$, LM_2, LM_3, and LM_4 is set to 1 if the difference value of the pixel $d = (d_1, d_2, d_3)$ is odd, and if the difference value of the pixel $d = (d_1, d_2, d_3)$ is even, then LM_2, LM_3, and LM_4 is set to 0.

Step 6: Following the completion of the embedding process, the next step involves creating a new pixel p'_0 as outlined in (8).

$$\begin{cases} p'_0 = p_0 \\ p'_1 = \tilde{d}_1 + p_0 \\ p'_2 = \tilde{d}_2 + p_1 \\ p'_2 = \tilde{d}_2 + p_2 \end{cases} \tag{8}$$

In Table 1, we present an illustration showing steps 3 to 6 of how data are hidden in a scrambled image. We illustrate the embedded data.

Table 1. Illustration of the data embedding and extraction

Embedding	Extraction
Step 3: Assume that we have a block of quad of grayscale values, denoted as $p = (112, 125, 149, 157)$, and we compute the difference value, d, using (3). $d_1 = 125 - 112 = 13$ $d_2 = 149 - 125 = 24$ $d_3 = 157 - 149 = 8$ $d = (13, 24, 8)$	**Steps 2 and 3:** Assuming we have a block of quad of grayscale values, denoted as $p' = (112, 123, 141, 158)$ and $LM = (1110)$, we compute the difference value, $\tilde{d}$, using (3).
Step 4: reduce d using (7) $h = \lfloor \log_2 \lvert 13 \rvert \rfloor = 3\, then, 3 \times 2^{3-1} \leq 13 \leq 4 \times 2^{3-1} - 1$ $d'_1 = 13 - 2^{\log_2 \lvert 13 \rvert} = 13 - 2^3 = 13 - 8 = 5$ $LM_2 = 1$ $h = \lfloor \log_2 \lvert 24 \rvert \rfloor = 4\, then, 3 \times 2^{3-1} \leq 24 \leq 4 \times 2^{3-1} - 1$ $d'_2 = 24 - 2^{\log_2 \lvert 24 \rvert} = 24 - 2^4 = 24 - 16 = 8$ $LM_3 = 1$ $h = \lfloor \log_2 \lvert 8 \rvert \rfloor = 3\, then, 2 \times 2^{3-1} \leq 8 \leq 3 \times 2^{3-1} - 1$ $d'_3 = 8 - 2^{\log_2 \lvert 8 \rvert - 1} = 8 - 2^{3-1} = 8 - 4 = 4$ $LM_4 = 0$ $d' = (5, 8, 4)$	$\tilde{d}_1 = 123 - 112 = 11$ $b_1 = LSB(\tilde{d}_1) = LSB(11) = 1$ $d'_1 = \left\lfloor \frac{\tilde{d}_1}{2} \right\rfloor = \lfloor \frac{11}{2} \rfloor = 5$ $LM_2 = 1$ $d_1 = d'_1 + 2^{\log_2 \lvert d'_1 \rvert + 1} = 5 + 8 = 13$ $p_1 = 13 + 112 = 125$ $\tilde{d}_2 = 141 - 125 = 16$ $b_2 = LSB(\tilde{d}_2) = LSB(16) = 0$ $d'_2 = \left\lfloor \frac{\tilde{d}_2}{2} \right\rfloor = \lfloor \frac{16}{2} \rfloor = 8$ $LM_3 = 1$ $d_2 = d'_2 + 2^{\log_2 \lvert d'_3 \rvert + 1} = 8 + 16 = 24$ $p_2 = 24 + 125 = 149$
Step 5: Consider that we possess confidential data $b = (101)_2$ $\tilde{d}_1 = 2 \times 5 + 1 = 11$ $\tilde{d}_2 = 2 \times 8 + 0 = 16$ $\tilde{d}_3 = 2 \times 4 + 1 = 9$ $\tilde{d} = (11, 16, 9)$	$\tilde{d}_3 = 158 - 49 = 16$ $b_3 = LSB(\tilde{d}_3) = LSB(9) = 1$ $d'_3 = \left\lfloor \frac{\tilde{d}_3}{2} \right\rfloor = \lfloor \frac{9}{2} \rfloor = 4$ $LM_4 = 0$ $d_3 = d'_3 + 2^{\log_2 \lvert d'_3 \rvert} = 4 + 4 = 8$ $p_3 = 8 + 149 = 157$
Step 6: Create a new pixel using (12) $p'_0 = p_0 = 112$ $p'_1 = \tilde{d}_1 + p_0 = 11 + 112 = 123$ $p'_2 = \tilde{d}_2 + p_1 = 16 + 125 = 141$ $p'_2 = \tilde{d}_2 + p_2 = 9 + 149 = 158$ $LM_1 = 1$ $p' = (112, 123, 141, 158)$ $LM = (1110)$	$p' = (112, 125, 149, 157)$

2.2. Data extraction step

The extraction process is executed in the following step to retrieve the embedded data:

Step 1: The extraction process is performed as in the embedding stage, with the distinction that data retrieval starts from the last layer. Using the same secret key k as in the embedding process, the watermark image is scrambled using arnold's cat map algorithm using (9), employing the secret key k_n.

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} 1 & b \\ c & bc+1 \end{bmatrix} \begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} \bmod (N) \tag{9}$$

Step 2: The scrambled watermark in iteration k_n is divided into 2×2 pixel blocks, transformed into vector p', and the pixel block pair differences $\tilde{d}$ is performed using (3). Uses the LM to uncover hidden data from the LSB differences $\tilde{d}$ where value of LM_b is not -1. If location map LM_b is 1, use the location map LM_{d_i} restores the image to its original appearance using (10), and if location map LM_b is 0, use (11).

$$d_i = \begin{cases} d'_i + 2^{\log_2 |d'_i| + 1}, & \text{if } LM_{d_i} = 1 \\ d'_i + 2^{\log_2 |d'_i} |, & \text{if } LM_{d_i} = 0 \end{cases} \tag{10}$$

$$d_i = \begin{cases} 2 \times \left\lfloor \frac{d'_i}{2} \right\rfloor + 1, & \text{if } LM_{d_i} = 1 \\ 2 \times \left\lfloor \frac{d'_i}{2} \right\rfloor, & \text{if } LM_{d_i} = 0 \end{cases} \tag{11}$$

Step 3: To reconstruct the initial image, we apply (12) as:

$$\begin{cases} p_0 = p_0 \\ p_1 = d_1 + p_0 \\ p_2 = d_2 + p_1 \\ p_3 = d_3 + p_2 \end{cases} \tag{12}$$

In Table 1, we present an illustration showing steps 2 and 3 of how data is extracted in a scrambled image.
Step 4: Extract the data and recovery image performed up to the initial iteration. Once all procedures are completed, the image will be restored to its original form.

3. RESULTS AND DISCUSSION

In this section, we conduct a comprehensive evaluation to assess the performance, payload capacity, and imperceptibility of the proposed method compared with that of the existing method [3]. In embedding data not use threshold of difference value. We conducted extensive simulations using six grayscale image variations in Figure 2. Figure 2(a) lena, Figure 2(b) baboon, Figure 2(c) barbara, Figure 2(d) airplane, Figure 2(e) peppers, and Figure 2(f) boat), each with dimensions of 512×512 pixels. Figure 2 provides visual representations of the six test images. The evaluation encompassed the integration of the watermark into the four layers of the image. To compare the similarity between the watermarked image and the original image, we employed two metrics: peak signal-to-noise ratio (PSNR), calculated using (13). PSNR generates higher values to signify greater similarity, whereas mean squared error (MSE) yields smaller values to indicate improved similarity. The secret key $k = (32, 128, 512, 1024)$ is used for 4 layers of embedding, where $T = 1536$, and the image size is $N = 512$.

$$MSE = \frac{1}{m \times n} \sum_{i}^{m-1} \sum_{j}^{n-1} |I(i,j) - I'(i,j)|^2 \quad (13)$$

$$PSNR = 10 \log_{10} \frac{255^2}{MSE} \quad (14)$$

(a) (b) (c) (d) (e) (f)

Figure 2. The six images used for testing; (a) lena, (b) baboon, (c) barbara, (d) airplane, (e) peppers, and (f) boat

3.1. Performance

The examinations demonstrated that both the embedding and extraction processes operate effectively, allowing the image to be accurately reconstructed to its initial state. The experiments indicated that enhancing the security of multiple-layer reversible watermarking in images can be achieved by first scrambling the image before the embedding process. According to the data in Tables 2 and 3, our method can achieve the maximum data hiding capacity. The results indicate that our approach matches the capacity of the method [3], but it requires more computational time. This is primarily due to the image randomization process using arnold's cat map algorithm, which consumes time proportional to the image resolution. For single-layer embedding, our approach exhibits an average computational time of 767.09, whereas the method [3] has an average computational time of 111.83. In the case of four-layer embedding, our method shows an average computational time of 984.13, whereas method [3] has an average computational time of 406.92.

In Table 2, the average gap in computational time between our method and that of [3] amounts to 655.26. Contrasting outcomes are evident in Table 3, specifically in the case of four-layer embedding, where the average computational time difference between our method and that of [3] is 577.21. This disparity arises from the embedding process occurring during specific iterations within a single randomization iteration period using arnold's cat algorithm.

3.2. Payload (incorporating capacity)

Our test results indicate that in a single layer, as shown in Table 2, our method and method [3] have the same data embedding capacity, both averaging 0.75 bits per pixel (bpp). This embedding capacity can be achieved without applying the threshold of the pixel difference value. Furthermore, when we consider the average data embedding capacity across all four layers based on Table 3, our method exhibits a slight reduction of approximately 0.0004 bpp compared with the method [3]. This reduction is still manageable, primarily because our proposed method involves an initial randomization of the image before the data embedding process.

Table 2. Comparison of embedding capacities and computational time in a single layer

		Capacity (bits)		Embedding capacity (bpp)		Computational time (second)	
		Method [3]	Our method	Method [3]	Our method	Method [3]	Our method
Lena	512×512	196,608	196,608	0.75	0.75	125.31	751.18
Baboon	512×512	196,608	196,608	0.75	0.75	101.36	718.61
Barbara	512×512	196,608	196,608	0.75	0.75	117.06	876.69
Airplane	512×512	196,608	196,608	0.75	0.75	108.25	817.72
Peppers	512×512	196,608	196,608	0.75	0.75	109.72	724.08
Boat	512×512	196,608	196,608	0.75	0.75	109.28	714.24

Table 3. Comparison of the embedding capacities and PSNR values in the four layers

	Capacity (bits)		Embedding capacity (bpp)		PSNR (dB)		Computational time (second)	
	Method [3]	Our method	Method [3]	Our method	Method [3]	Our method	Method [3]	Our method
Lena	786,432	786,294	3.0000	2.9995	35.96	21.75	366.56	981.61
Baboon	786,432	786,411	3.0000	2.9999	30.02	22.22	429.59	1,004.37
Barbara	786,420	786,279	2.9999	2.9994	30.32	21.47	407.27	1,008.47
Airplane	786,420	786,231	2.9999	2.9992	34.72	22.61	376.74	965.47
Peppers	786,432	786,354	3.0000	2.9997	36.14	21.41	414.04	968.64
Boat	786,411	786,384	2.9999	2.9998	33.37	22.64	447.31	976.21

3.3. Imperceptibility

The test results provide assurance that the image quality remains within acceptable standards following the data embedding process. Table 4 displays the detailed comparison results with the proposed method [3]. On average, in the case of single-layer embedding, the image quality post-embedding attains a commendable 28.25 dB as per the PSNR metric. For the four-layer embedding approach, the average image quality after the embedding process registers at 22.01 dB according to the PSNR.

Among the individual images subjected to single-layer embedding, the boat image stands out with the highest quality, boasting a PSNR of 29.05 dB. Conversely, the lena image exhibited the lowest quality among this group, with a PSNR of 27.54 dB. In the context of four-layer embedding, as shown in Table 3, the boat image emerges as the leader in image quality, recording a PSNR of 22.64 dB, whereas the peppers image again exhibits the lowest quality, with a PSNR of 21.41 dB.

Table 4. Comparison of the PSNR values for single-layer embedding

		PSNR value (dB) in bits per piexl (bpp)						
		0.1	0.2	0.3	0.4	0.5	0.6	0.7
Lena	Method [3]	55.05	50.44	47.60	45.46	43.94	42.83	41.93
	Our method	36.27	33.26	31.51	30.27	29.30	28.50	27.83
Baboon	Method [3]	41.14	38.44	37.52	37.01	36.55	36.05	35.45
	Our method	37.18	34.20	32.43	31.17	30.20	29.41	28.75
Barbara	Method [3]	49.54	47.00	43.91	40.69	38.33	36.75	35.74
	Our method	36.59	33.57	31.84	30.59	29.64	28.85	28.18
Airplane	Method [3]	52.96	50.73	47.81	43.77	42.05	41.13	40.65
	Our method	37.64	34.65	32.88	31.63	30.66	29.87	29.21
Peppers	Method [3]	49.71	46.90	45.13	43.66	42.95	42.40	41.88
	Our method	36.48	33.44	31.67	30.43	29.46	28.67	28.00
Boat	Method [3]	49.94	45.97	42.98	41.07	39.96	39.51	39.28
	Our method	37.77	34.75	33.03	31.79	30.82	30.03	29.35

According to the comparison results in Tables 3 and 4, the use of arnold's cat map algorithm for randomization before data embedding has an influence on the image quality following the embedding process. This influence is evident when comparing our approach with a previous method [3]. In the case of single-layer embedding, as shown in Table 4, the average decrease in visual quality was 11.43 dB. The most significant decrease was observed in the lena image, with an average reduction of 15.76 dB, whereas the smallest decrease was observed in the baboon image, with an average reduction of 5.55 dB. In the context of four-layer embedding, the average decrease in visual quality was 11.41 dB, with the lena image experiencing the most substantial reduction at 14.21 dB, whereas the baboon image exhibited the lowest decrease with an average of 7.80 dB.

4. CONCLUSION

Reversible watermarking is a creative way to protect digital copyrights because it allows watermarks to be added to digital content with multiple-layers without changing the original data. However, despite its utility, this technique is susceptible to attacks because of the absence of security keys in reversible watermarking methods. This study bolsters the security level of multiple-layer reversible watermarking methods by leveraging arnold's cat map as the foundational algorithm. The proposed approach effectively elevates security in watermark embedding. Through the use of security keys during the embedding process, we adeptly address the vulnerabilities inherent in security-key-lacking reversible watermarking methods while still preserving the capacity to recover the original data. Although our method experiences an increased computational time and reduction in visual quality compared with the previously proposed method, it maintains embedding capacity, ease of implementation, and simplicity. The post-data hiding image quality remains acceptable, as corroborated by the substantial PSNR values achieved. The security of the proposed technique can be further enhanced by encrypting the embedded data. This research can make valuable contributions to safeguarding digital copyrights and preserving data integrity in the upcoming era.

REFERENCES

[1] P. Maniriho, L. J. Mahoro, Z. Bizimana, E. Niyigaba, and T. Ahmad, "Reversible difference expansion multi-layer data hiding technique for medical images," *International Journal of Advances in Intelligent Informatics*, vol. 7, no. 1, pp. 1–11, Mar. 2021, doi: 10.26555/ijain.v7i1.483.

[2] Z. Syahlan and T. Ahmad, "Reversible data hiding method by extending reduced difference expansion," *International Journal of Advances in Intelligent Informatics*, vol. 5, no. 2, pp. 101–112, Jul. 2019, doi: 10.26555/ijain.v5i2.351.

[3] A. Arham, H. A. Nugroho, and T. B. Adji, "Multiple layer data hiding scheme based on difference expansion of quad," *Signal Processing*, vol. 137, pp. 52–62, Aug. 2017, doi: 10.1016/j.sigpro.2017.02.001.

[4] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 8, pp. 890–896, Aug. 2003, doi: 10.1109/TCSVT.2003.815962.

[5] C. L. Liu, D. C. Lou, and C. C. Lee, "Reversible data embedding using reduced difference expansion," *Proceedings - 3rd International Conference on Intelligent Information Hiding and Multimedia Signal Processing, IIHMSP 2007*, vol. 1, pp. 433–436, 2007, doi: 10.1109/IIH-MSP.2007.267.

[6] R. Anushiadevi and R. Amirtharajan, "Separable reversible data hiding in an encrypted image using the adjacency pixel difference histogram," *Journal of Information Security and Applications*, vol. 72, p. 103407, Feb. 2023, doi: 10.1016/j.jisa.2022.103407.

[7] A. Arham, H. A. Nugroho, and T. B. Adji, "Combination schemes reversible data hiding for medical images," *Proceedings - 2016 2nd International Conference on Science and Technology-Computer, ICST 2016*, pp. 44–49, 2017, doi: 10.1109/ICSTC.2016.7877345.

[8] A. Arham and N. Lestari, "Secure medical image watermarking based on reversible data hiding with Arnold's cat map," *International Journal of Advances in Intelligent Informatics*, vol. 9, no. 3, p. 445, Oct. 2023, doi: 10.26555/ijain.v9i3.1029.

[9] C. C. Islamy and T. Ahmad, "Histogram-based multilayer reversible data hiding method for securing secret data," *Bulletin of Electrical Engineering and Informatics (BEEI)*, vol. 8, no. 3, pp. 1128–1134, Sep. 2019, doi: 10.11591/eei.v8i3.1321.

[10] K. M. Kamath and R. S. Kunte, "Framework for reversible data hiding using cost-effective encoding system for video steganography," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 5, pp. 5487–5496, Oct. 2020, doi: 10.11591/IJECE.V10I5.PP5487-5496.

[11] H. Yi, S. Wei, and H. Jianjun, "Improved reduced difference expansion based reversible data hiding scheme for digital images," in *ICEMI 2009 - Proceedings of 9th International Conference on Electronic Measurement and Instruments*, Aug. 2009, pp. 4315–4318, doi: 10.1109/ICEMI.2009.5274054.

[12] S. M. Abdullah and A. A. Manaf, "Multiple layer reversible images watermarking using enhancement of difference expansion techniques," *Communications in Computer and Information Science*, vol. 87 CCIS, no. PART 1, pp. 333–342, 2010, doi: 10.1007/978-3-642-14292-5_35.

[13] P. Maniriho and T. Ahmad, "Information hiding scheme for digital images using difference expansion and modulus function," *Journal of King Saud University - Computer and Information Sciences*, vol. 31, no. 3, pp. 335–347, Jul. 2019, doi: 10.1016/j.jksuci.2018.01.011.

[14] A. M. Alattar, "Reversible watermark using difference expansion of triplets," in *IEEE International Conference on Image Processing*, 2003, vol. 1, pp. 501–504, doi: 10.1109/icip.2003.1247008.

[15] A. M. Alattar, "Reversible watermark using difference expansion of quads," ICASSP, *IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, vol. 3, 2004.

[16] J. Y. Hsiao, K. F. Chan, and J. Morris Chang, "Block-based reversible data embedding," *Signal Processing*, vol. 89, no. 4, pp. 556–569, Apr. 2009, doi: 10.1016/j.sigpro.2008.10.018.

[17] T. Ahmad, M. Holil, W. Wibisono, and I. R. Muslim, "An improved Quad and RDE-based medical data hiding method," in *Proceeding - IEEE CYBERNETICSCOM 2013: IEEE International Conference on Computational Intelligence and Cybernetics*, Dec. 2013, pp. 141–145, doi: 10.1109/CyberneticsCom.2013.6865798.

[18] M. H. A. Al Huti, T. Ahmad, and S. Djanali, "Increasing the capacity of the secret data using DEpixels blocks and adjusted RDE-based on grayscale images," in *Proceedings of 2015 International Conference on Information and Communication Technology and Systems*, ICTS 2015, Sep. 2016, pp. 225–230, doi: 10.1109/ICTS.2015.7379903.

[19] N. J. De La Croix, C. C. Islamy, and T. Ahmad, "Reversible data hiding using pixel-value-ordering and difference expansion in digital images," in *Proceeding - IEEE International Conference on Communication, Networks and Satellite, COMNETSAT 2022*, Nov. 2022, pp. 33–38, doi: 10.1109/COMNETSAT56033.2022.9994516.

[20] A. Arham and O. S. Riza, "Reversible data hiding using hybrid method of difference expansion on medical image," *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika*, vol. 6, no. 2, p. 11, Jan. 2021, doi: 10.26555/jiteki.v6i2.16965.

[21] C. F. Lee, J. J. Shen, and C. Y. Wu, "Hiding scheme based on improved reduced difference expansion in multi-block shape," *Smart Innovation, Systems and Technologies*, vol. 278, pp. 297–310, 2022, doi: 10.1007/978-981-19-1053-1_27.

[22] W. Ding, H. Zhang, R. Reulke, and Y. Wang, "Reversible image data hiding based on scalable difference expansion," *Pattern Recognition Letters*, vol. 159, pp. 116–124, Jul. 2022, doi: 10.1016/j.patrec.2022.05.014.

[23] C.-F. Lee, J.-J. Shen, and C.-Y. Wu, "Overlapping difference expansion reversible data hiding," *International Journal of Network Security*, vol. 25, pp. 201–211, 2023, doi: 10.6633/IJNS.202303 25(2).03.

[24] A. Mehbodniya *et al.*, "Multilayer reversible data hiding based on the difference expansion method using multilevel thresholding of host images based on the slime mould algorithm," *Processes*, vol. 10, no. 5, p. 858, Apr. 2022, doi: 10.3390/pr10050858.

[25] W. Wang, "A reversible data hiding algorithm based on bidirectional difference expansion," *Multimedia Tools and Applications*, vol. 79, no. 9–10, pp. 5965–5988, Mar. 2020, doi: 10.1007/s11042-019-08255-z.

[26] P. Maniriho and T. Ahmad, "Enhancing the capability of data hiding method based on reduced difference expansion," *Engineering Letters*, vol. 26, no. 1, pp. 45–55, 2018.

[27] C. F. Lee, J. J. Shen, and Y. H. Lai, "Data hiding using multi-pixel difference expansion," in *2018 3rd International Conference on Computer and Communication Systems, ICCCS 2018*, Apr. 2018, pp. 61–66, doi: 10.1109/CCOMS.2018.8463244.

[28] P. C. Mandal, I. Mukherjee, and B. N. Chatterji, "High capacity reversible and secured data hiding in images using interpolation and difference expansion technique," *Multimedia Tools and Applications*, vol. 80, no. 3, pp. 3623–3644, Jan. 2021, doi: 10.1007/s11042-020-09341-3.

[29] T. S. Nguyen, V. T. Huynh, and P. H. Vo, "A novel reversible data hiding algorithm based on enhanced reduced difference expansion," *Symmetry*, vol. 14, no. 8, p. 1726, Aug. 2022, doi: 10.3390/sym14081726.

[30] Y. Samudra and T. Ahmad, "Improved prediction error expansion and mirroring embedded samples for enhancing reversible audio data hiding," *Heliyon*, vol. 7, no. 11, p. e08381, Nov. 2021, doi: 10.1016/j.heliyon.2021.e08381.

[31] A. J. Ilham and T. Ahmad, "Reversible data hiding scheme based on general difference expansion cluster," *International Journal of Advances in Soft Computing and its Applications*, vol. 12, no. 3, pp. 11–24, 2020.

[32] D. C. Lou, M. C. Hu, and J. L. Liu, "Multiple layer data hiding scheme for medical images," *Computer Standards and Interfaces*, vol. 31, no. 2, pp. 329–335, Feb. 2009, doi: 10.1016/j.csi.2008.05.009.

[33] J. Wang, J. Ni, and J. Pan, "A high performance multi-layer reversible data hiding scheme using two-step embedding," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7128 LNCS, pp. 42–56, 2012, doi: 10.1007/978-3-642-32205-1_6.

[34] Z. Pan, S. Hu, X. Ma, and L. Wang, "Reversible data hiding based on local histogram shifting with multilayer embedding," *Journal of Visual Communication and Image Representation*, vol. 31, pp. 64–74, Aug. 2015, doi: 10.1016/j.jvcir.2015.05.005.

[35] C.-F. Lee, J.-J. Shen, and Y.-H. Lai, "Reversible data hiding based on improved reduced difference expansion and multi-layer recursive embedding," in *Proceedings of The 6th IIAE International Conference on Intelligent Systems and Image Processing 2018*, 2018, pp. 25–31, doi: 10.12792/icisip2018.009.

[36] X. Wu, T. Qiao, M. Xu, and N. Zheng, "Secure reversible data hiding in encrypted images based on adaptive prediction-error labeling," *Signal Processing*, vol. 188, p. 108200, Nov. 2021, doi: 10.1016/j.sigpro.2021.108200.

[37] A. Meenpal and S. Majumder, "Image content based secure reversible data hiding scheme using block scrambling and integer wavelet transform," *Sadhana - Academy Proceedings in Engineering Sciences*, vol. 47, no. 2, p. 54, Jun. 2022, doi: 10.1007/s12046-022-01828-z.

[38] I. M. S. De Rosal *et al.*, "Secure reversible data hiding in the medical image using histogram shifting and RC4 encryption," *Proceedings - 2019 International Seminar on Application for Technology of Information and Communication: Industry 4.0: Retrospect, Prospect, and Challenges, iSemantic 2019*, pp. 28–33, 2019, doi: 10.1109/ISEMANTIC.2019.8884306.

[39] S. Kukreja and G. Kasana, "A secure reversible data hiding scheme for digital images using random grid visual secret sharing," in *Proceedings - 2019 Amity International Conference on Artificial Intelligence, AICAI 2019*, Feb. 2019, pp. 864–869, doi: 10.1109/AICAI.2019.8701360.

[40] M. Nasir *et al.*, "Secure reversible data hiding in images based on linear prediction and bit-plane slicing," *Mathematics*, vol. 10, no. 18, p. 3311, Sep. 2022, doi: 10.3390/math10183311.

[41] S. Kamil, M. Sahu, K. R. Raghunandan, and A. K. Sahu, "Secure reversible data hiding using block-wise histogram shifting," *Electronics (Switzerland)*, vol. 12, no. 5, p. 1222, Mar. 2023, doi: 10.3390/electronics12051222.

BIOGRAPHIES OF AUTHORS

Aulia Arham received the B.Sc. degrees in Informatics Engineering from the *Sekolah Tinggi Manajemen Informatika dan Komputer Duta Bangsa* (STMIKDB), Surakarta, Indonesia and the M.S. degree in Electrical Engineering at Universitas Gadjah Mada, Yogyakarta, Indonesia, in 2016. In 2018, he is appointed as lecturer in the Department of Information System, Faculty of Science and Technology, Universitas Negeri Islam Imam Bonjol, Padang, Indonesia. His research interests are watermarking, steganography, cryptography, security, image processing, and medical imaging. He can be contacted at email: auliaarham@uinib.ac.id.

Novia Lestari was born in Lubuk Alung, West Sumatra, Indonesia on November 1, 1990. She received the B.Sc. degrees in Informatics Engineering and the M.S. degree in information technology from the University of Putera Indonesia YPTK Padang. Research interest in data mining and artificial intelligence. She can be contacted at email: noviia.lestari@uinib.ac.id.