Enhanced Flipping Technique to Reduce variability in Image Steganography

Samar Kamil12, Siti Norul Huda Sheikh Abdullah 1, Mohammad Kamrul Hasan1, Senior Member, IEEE, and Farah Aqilah Bohani1
1 Center for Cyber Security, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, 43600 Bangi, Selangor, Malaysia
2 College of Computer Science & Informatics Systems, University of Al-Mansour, 10068, Baghdad, Iraq

ABSTRACT Steganography algorithms hide the secret message in the cover image and provide imperceptibility to the attacker. The Least Significant Bit (LSB) algorithm is the preferred data hiding method. In this method, the least significant bit of the cover image pixel is substituted with the secret data bit. However, this method provides high variability if k-LSB bits of the cover image is replaced with secret data bit. To reduce the variability, various optimization algorithms are deployed in image steganography. These algorithms search the optimal pixels in the cover image, and data hiding is performed using the k-bits LSB method. Several iterations of optimization algorithms have increased the time complexity upon achieving this goal. Therefore, in this paper, a data hiding approach is designed based on the flipping approach that reduces variability and provides lesser time complexity. In the proposed method, initially, data hiding is performed using the k-bit LSB method in the cover image, and stego image is obtained. After that, the absolute difference between the cover and stego image is determined and compared with the threshold value. If the absolute difference is higher than the threshold value, then the adjacent bit of the k-bit LSB method is flipped. This process reduces the variability because flipping the adjacent bit will make the pixel value of the stego image closer to the cover image. The simulation evaluates using various performance metrics upon testing on standard dataset images. The simulation results show that the proposed method provides lesser variability, good visual quality, lesser time complexity than Genetic and Bayesian Optimization algorithms and the existing flip method.

INDEX TERMS Flipping method, data hiding, Least Significant Bit, Variability.

I. INTRODUCTION
Steganography is the science and art of secret communication, which aims to hide secret messages (e.g., text, image, or video) in a cover medium such as an image to provide an imperceptibility attack [1]. Besides images, various cover media are available such as text, audio, and videos [2]. Of these media, the image is the most preferred cover medium because it contains many redundant pixels. Thus, a small variation in the pixels after data hiding is not visually impacted [3].

To improve steganography performance, several techniques have been proposed to cover data inside the cover media, such as least significant bit substitution (LSB) [4], a discrete cosine transform [5], Pixel Value ordering [6], Histogram Shifting [7], and Difference Expansion [8]. The most popular approach is the LSB which consists of hiding secret bits in the LSB of some pixels in the cover image [9]. This technique has a high embedding capacity [10]. However, this technique has a high variability that negatively affects the visual quality [11].

On the other hand, to reduce variability without negatively impacting the embedding capacity, some optimization algorithms such as the Genetic Algorithm [12], Particle Swarm Optimization [13], and Bayesian Optimization [14] have been proposed. However, these techniques generate the optimal secret data matrix and try to match the cover and secret data bits, and it takes a long execution time [15]. Hence, [16] proposed another flipping method to address long execution time issues. This method was performed after checking the bits of the cover image. If the bits of the cover image change more than fifty percent, then this technique is carried out on the message bits [17].
In steganography, embedding data within an image has a trade-off between image quality and embedding capacity. Specifically, the more data are covered within a carrier image, the further variability the image suffers, causing a decline in the resultant stego image quality [18]. Although flipping methods have been proposed to improve metaheuristic algorithm issues, these algorithms still have an issue in either less visual or embedding capacity [18]. The question is: how to design a new flipping method that reduces the variability between the cover and stego image after the data hiding process with higher embedding capacity? Therefore, this study aims to design a new flipping method to reduce the variability between the cover and stego images after the data hiding process. Our hypothesis is "If the flipping technique can affect the cover image variability, visual quality, and embedding capacity, then hiding data using an effective flipping technique can produce a good data hiding technique that has small variability, better visual quality, and high embedding capacity".

The remainder of this paper is as follows. Section II presents an overview of the proposed technique. Section III presents several empirical proofs, and scientific justifications underlie the experimental results. In the last, the paper concludes and infers our hypotheses and infers our hypothesis.

II. RELATED WORKS
The study guided us to the best spatial domain methods possible to use in this paper. The most popular type used in literature is the spatial domain due to simplicity and efficiency. Some of the methods that used spatial domain are given below:

A. Substitution Least Significant Bit (K-LSB)
One common technique is to manipulate the least-significant-bit (LSB) planes by directly replacing the LSBs of the cover image with the message bits [19]. The LSB Substitution algorithm uses the least significant bits in the data hiding stage; first, the secret data is converted to the binary number system. After, the secret data bits stream changes the least significant bits of the cover image pixels [19][20]. LSB methods typically achieve high capacity [19][10]. It is easy to implement, fast and has better imperceptibility [20][21]. However, LSB-based hiding schemes have certain limitations, such as visible degradation of the stego image (Higher variability) when embedding extensive data, which affects the quality of stego image became easily vulnerable against attacks [21][22].

Many researchers have investigated the data hiding using LSB substitution by [23] introducing adaptive LSB embedding by utilizing the pixel value differencing (PVD) characteristics. This method divides the cover image into two-pixel blocks, further computing the difference in value between the two pixels. The numbers of secret bits are estimated based on the computed difference value. Next, the adaptive LSB scheme embeds the number of secret data bits inside the selected pixel blocks. This method also introduced a readjustment process to keep the stego-pixels with respective ranges; it successfully achieved the larger embedding capacity by retaining high visual imperceptibility. However, they did not prove that their technique is not vulnerable to pixel difference histogram analysis [24].

Next, [25] proposed an adaptive LSB substitution for data embedding based on edge computation. A maximum of 4 LSBs is utilized during the embedding process. On the other hand, this method did improve the embedding capacity. However, it suffered from low visual imperceptibility [26]. Then, [27] proposed a semi-reversible data hiding technique involving interpolation and LSB substitution. The interpolation methods are used to scale up and scale down the cover image, where the LSB substitution (up to 3 LSBs) is employed for embedding, unlike the use of 4 LSBs by the previous researcher. This method provides a larger embedding capacity while maintaining acceptable visual quality. However, this method did not discuss security evaluation [26]. Another novel LSB-based technique [28] proposed is bit inversion to improve the PSNR value. Certain LSBs of the cover image pixels changed in this inversion technique if matched with a particular pattern. Thus, this enhanced the visual quality. However, this technique suffered from a low payload [26].

In addition, [29] proposed a high-capacity image steganography technique based on the LSB substitution method. In this method, the LSB substitution technique divides the image into two parts: embedding data and indicating the embedding changes. This method increases the embedding capacity and improves the visual quality of the stego image, while the number of LSB substitutions is adaptive, ranging from 1 – 5. The visual quality of this method directly depends on the number of utilized LSBs; however, as the number of LSB substitutions increases, the visual quality of the stego-image decreases [26].

Moreover, [24] integrated the LSB substitution with a PVD method. This method divides the image into 2 × 2-pixel blocks and applies the K-LSB bit in the upper-left pixel of the block. Next, a PVD embedding process is applied to the base (upper-left) pixel with the remaining pixels of the block. This method provides a larger embedding capacity with the improved PSNR value. However, the k-bits ranges are 1 to 3, indirectly indicating that up to 3 LSBs of a pixel can change during the embedding process [26].

Finally, [30] proposed a novel steganography approach based on the combination of LSB substitution mechanism and edge detection. This method divides the cover pixels into edge and non-edge areas. The edge information determines by the three most significant bits (MSBs) of the cover pixels, where the rest of the 5 LSBs are used adaptively by the LSB substitution. This method obtained high embedding capacity...
while retaining acceptable visual quality. However, a maximum of 5 LSBs is used [26].

Previous research [23][24][25][26] has shown that LSB substitution increases the embedding capacity. However, we found trade-offs among all these objectives because these are correlated to one another. Most of the larger capacity-based LSB methods employed the maximum LSBs or modified the maximum number of bits in each pixel (3−5 bits). This maximum bit’s modification in pixels increases the visual distortion between cover and stego-pixels (that can be seen as a low PSNR value 30 to 33 dB). Consequently, it reduces the stego-image visual quality (Higher variability) and increases the risk of steganalysis detection attacks [26].

Optimization algorithms such as genetic algorithm (GA) and Bayesian optimization algorithm (BOA) have gained popularity as they have overcome the challenges of LSB substitution data hiding technique, such as less security, low visual quality, and increasing variability [5][22]. The GA and BOA algorithms match the image pixel bits with data bits to generate minimum variability after data hiding. Furthermore, details of GA and BOA algorithms are explained below.

B. Optimization Algorithm -Based Data Hiding Technique

Steganography methods have good visual quality and variable data embedding capacity using a GA algorithm [22][31]. Utilizing a genetic algorithm avoids exhausting searching and allows us to find the best place in the host image for embedding modified secret data. Thus, the proposed method can achieve high embedding capacity and enhance the stego-image quality (the PSNR value) [12]. However, they require much auxiliary information like gradient, derivatives and are long computationally [31].

Some studies used GAs to enhance steganography performance [12][22]. A review of some of their applications is presented by [12] proposed a spatial domain GA-based reversible data embedding method with tunable visual quality. This method modeled the hiding process as a search and optimization problem. As a result, embedding payload and visual quality were improved. However, the computational complexity increases, and security is not proven [22].

Then, [32] proposed a chaotic maps-based method to improve the GA’s data hiding technique. In the method, the GA fitness function was selected based on PSNR. Further, the various sizes of secret data were employed in the cover image using random functions and chaotic maps. Meanwhile, genetic randomness was performed using several chaotic maps versions such as gauss, logistic, and tent. Finally, the chaotic maps were considered the fastest than random function for the steganographic technique. The obtained results of gauss that the obtained results of two dimensions of data hiding applications based on the genetic algorithm are faster. There are limitations in achieving effective results from the genetic algorithm, such as the determination of crossover, mutation techniques, and end conditions [32].

Finally, [33] proposed a discrete cosine transform (DCT) domain with multi-bit LSB steganography that using to exploit the Genetic Algorithm (GA) for optimization. The pseudorandom function is used to sample pixels. The improved robustness in DCT while the multi-bit LSB technique increases the payload capacity in LSB and GA used for optimization. As a result, it attempted to reduce but resulted in variability. However, DCT, including the frequency domain, cannot be manipulated because it destroys the image’s main features. Another limitation is to work with fixed-length genes (here, the fixed-length binary fragment) to produce offspring that restrict taking different variations to capacitate more secreted message bits [33]. The Bayesian optimization algorithm (BOA), compared to the GA, generally finds the optimal solution in fewer iterations, which can decrease the number of cost function calculations and reduce the computational burden [14].

In one of the most recent evaluation studies, the BO algorithm used steganography to embed the secret data [14] proposed a method before embedding. The secret data are mapped to the optimal values using a Bayesian optimization algorithm (long with introducing a novel mutation operator) to reduce the variability. The main idea of the proposed method is that before the embedding process, the secret data are mapped to the optimal values using the BO algorithm (along with introducing a novel mutation operator). Then, the mapped data are embedded into the low-order bits of host pixels using modulus function and a systematic and reversible algorithm. Since the proposed method can embed data into more significant bits, enhancing the payload while preserving its visual image quality is vital. However, the BO algorithm takes the maximum number of iterations equal to 10 [14]. The GA and BOA algorithms provide high security and reduce variability. However, the major challenge of the GA and BOA algorithms operation requires an extra computational complexity [14][31][32]. Therefore, other researchers proposed a new approach named flipping bits to solve variability and computation time.

C. Flipping Method

Studies have incorporated the flipping bits method for data hiding by [18] proposed a reversible bit flipping method for concealing 2 bits of secret data in a segment of 2 pixels. The last LSB bits of the pixels have been exploited in each segment to conceal the secret data at the first embedding layer. In the current work, a segment of 2 pixels hides. Only 2 bits of secret information, and again, it does not directly substitute the LSB bits. However, this method suffers from low embedding capacity, [17] proposed a modified bit flipping technique for hiding information in an image called bit flipping-1 and bit flipping-2. In the method, they change the data to be hidden into a binary form. In addition, the dimension of secret data also changed to 18 bits. This flipping was placed at the top of the binary message. The message is now a combination of both flipping-1 and bit flipping-2. This method provides the PSNR
value increasing around 47.40 dB. However, this method suffers from low embedding capacity.

Similarly, [16] proposed flipping the message bits to increase imperceptibility in the least significant bit of image steganography. Firstly, the message is converted into a binary form. This method is carried out after checking the bits of the cover image. If the cover image’s bits change more than fifty percent, this technique is carried out on the message bit. This method provides the PSNR value increasing around 51 dB to 60 dB and the MSE value dropping from around 0.44 to 0.05. However, it offers less security because executing flipping bits if the change in bits per pixel is more than 50%, so choosing the cover image should be selective. Table 1 summarizes the advantages and disadvantages of the LSB technique, GA algorithm, and flipping methods.

| TABLE 1. Summarization of Advantages and disadvantages of Existing Methods |
|----------------------------------------------------------|
| Algorithm | Advantages | Disadvantages |
|------------|-------------|----------------|
| LSB technique | Higher embedding capacity | Prone to statistical attack. |
| GA and BO algorithms | Low variability | Computationally expensive. |
| Flipping method | Low variability | Prone to statistical attack |
|               | Higher embedding capacity | Higher imperceptibility |

D. Issues and Objectives

The following issues are found from the comparative analysis presented above for the existing methods:

- The most popular technique for steganography is the Least Significant Bit (LSB) technique, as it provides better embedding capacity. Still, the drawback is that it produces higher variability and is prone to attack [21] [22].
- Although these GA and BO algorithms provide low variability and higher embedding capacity, they consume a long iteration time [14] [31][32].
- The flipping method avoids the iteration time in the data hidden. However, it has less embedding capacity.

This research aims to design a data hiding approach that provides better imperceptibility, embedding capacity, and lesser time complexity.

III. PROPOSED FLIPPING METHOD

The proposed layer comprises of two-stages. The first stage uses the k-bits LSB technique for data hiding purposes, while the second stage improves image variability using the proposed flipping method.

A. STAGE I: THE DATA HIDING USING K-LSB

We follow a standard data hiding using the k-LSB technique at this stage. First, read the cover image (C) and the secret data (SD), and split the SD into k-bits chunks known as secret data chunks (SDC). This makes the C and the SDC are ready to begin the data hiding process. The data hiding process is started by applying AND operation between each pixel in the cover image with a constant value (h). The h is based on the contrast of the cover image. For example, if we use the 4-LSB method for 8-data bit (which is equal to 255 or 1111 1111 for maximum value), the 4-least significant bit (1111), will be changed to 0000. Thus, the h value should be (1111-0000) equal to 24010. The detailed steps for Stage I are as follows:

**Step 1:** Read the C and SD.
**Step 2:** Split SD into 4-bits chunks (SDC)
**Step 3:** For each chunk, apply the logical AND operation between the cover pixel and h value, as given in Equation (1):

\[ C' = C \text{ AND } h \]  

**Step 4:** Apply the logical OR operation between the SDC and C’ bits to hide the data in the cover pixels, to produce the S using Equation (2):

\[ S = C' \text{ OR } EDC \]  

**Step 5:** Repeat step 3 and 4 for all remaining chunks of SDC until the whole SDC are embedded into the C’.
**Step 6:** End Stage I. Return the Stego image (S).

In Stage II, the proposed flipping method is then applied to reduce the S variability.

B. MULTIPART FIGURES

The detailed steps for Stage II are:

**Step 1:** Set the threshold value, T.
**Step 2:** For each chunk, calculate the absolute difference (AD) between the S and C image using Equation (3)

\[ AD_{ij} = |C_{ij} - S_{ij}|, \text{ where } 0 \leq i \leq \text{row}, 0 \leq j \leq \text{column} \]  

where AD_{ij} denotes the absolute difference, C_{ij} denotes the cover, S_{ij} denotes the stego image, and i, j are row and column

**Step 3:** If the absolute difference (AD) is higher than the threshold T, we flip the 5th stego pixel (FSP).
**Step 4:** Repeat steps 2 and 3 for all remaining chunks.
Step 5: Return the flipped stego pixels (FSP).  

The proposed flipping method begins with an absolute difference (AD) between the cover and stego images. Then, if the absolute difference (AD) is higher than the threshold T, we flip the adjacent pixel of the k-bits LSB technique (i.e., the 5th bit if we use 4-bit LSB). Equations (4) and (5) are used to calculate the T value and flipping variable.

\[ T = \frac{MV}{2} \]  

where MV denotes the maximum variability, i.e. \(2^4=16\) in the 4-bits LSB technique.

\[ S'_{ij} = \begin{cases} \text{flip, } AD_{ij} > T \\ \text{no flip, } AD_{ij} \leq T \end{cases} \text{ where } 0 \leq i \leq \text{row}, \]

\[ 0 \leq j \leq \text{column} \]  

where \(S'\) denotes the stego image, \(T\) denotes the threshold and \(AD\) denotes the Absolute Difference while flipping the 5th pixel in the stego image (Equation 3).

To further illustrate how the proposed flipping method is performed, please refer to the example shown in Table 2.

Figures 1 and 2. In this example, we use 4-bit LSB data hiding technique. The MV for 4-bit is \(2^4=16\) (1111), and T value is set to be 8 (refer to Equation (4)). Therefore, the maximum variability value will affect half of the 8-bit pixels.

| K-bits LSB Technique | Max Variability | Threshold (T) |
|----------------------|----------------|--------------|
| 4-bits               | \(2^4=16\)    | 8            |

Table 2: Value for the 4-bits LSB Technique

Figure 2 shows the entire process of the proposed flipping method can be summarized as follows: First, it reads the cover image (33 F7 0F CC16) and secret data (i.e. BF 55 33 33). Then, split secret data into 4-bits chunks (SDC) and apply AND operation between the cover pixel and h value. Then, the \(C'\) is to apply OR operation with the \(SDC\) bits to hide the data in the cover pixels to produce the stego image (S). After calculating the OR and AND operations between \(C'\) and \(SDC\). The next the absolute difference (AD) between the cover image and stego image. Then, if the absolute difference (AD) is higher than the threshold T=8, it flips the adjacent pixel of the k-bits LSB technique is flipped (5th bit); otherwise, the pixel value remains.

**FIGURE 1.** The proposed flipping method


### FIGURE 2.

Steps for the proposed flipping method

| C (0011 1011) - S (0011 1011) = 1000 (8) | 8 | 10 | 7 |
|--------------------------------------------|---|----|---|
| 9                                           | 6 | 9  | 0 |
| 0                                           | 3 | 0  | 3 |
| 0                                           | 3 | 3  | 15 |

Pixel adjustment using proposed flipping method if Absolute difference (AD) greater than the threshold (T), T=8

| 8 | 8 | 10 | 7 |
|---|---|----|---|
| 9 | 6 | 9  | 0 |
| 0 | 3 | 0  | 3 |
| 0 | 3 | 3  | 15 |

Flipping the 5th pixel bits

| 0011 1011 | 1111 1111 | 0000 1001 | 1100 0101 |
|-----------|-----------|-----------|-----------|
| 1011 0011 | 0011 0011 | 1110 0011 | 1010 1111 |

Stage I: The Data Hiding Using K-LSB

Stage II: Proposed Flipping Method For Reducing The Variability

Data Hiding Using K-LSB

| 0011 1011 | 1111 1111 | 0000 1001 | 1100 0101 |
|-----------|-----------|-----------|-----------|
| 1011 0011 | 0011 0011 | 1110 0011 | 0011 1011 |
| 0000 1000 | 1100 1100 | 1100 1100 | 1100 1100 |

| 0011 1011 | 1111 1111 | 0000 1001 | 1100 0101 |
|-----------|-----------|-----------|-----------|
| 1011 0011 | 0011 0011 | 1110 0011 | 0000 1010 |
| 0000 1111 | 0011 0000 | 1100 1100 | 1100 1111 |
| 1100 1011 | 1111 0111 | 0000 1011 | 1101 1111 |

Absolute difference (AD) between C and S
On the receiver side, the process of encrypted data extraction is shown in Figure 3. The FSP read and its LSB k-bits were extracted using the AND operation with a constant value \( h \). The extracted k-bits are SDC bits. In the last, the SDC bits concatenate for determining the original encrypted data.

**Algorithm 1: The Data Hiding Using K-LSB**

Input: Cover Image C and Secret Data (SD).

**Output:** Stego Image (S)

**Begin**

**Step 1:** Read the cover image C and Secret data (SD).

**Step 2:** Process the image into Block Size 4×4

**Step 3:** Divide (SD) into 4-bit Chunk (SDC)

**Step 4:** For each chunk, apply the logical AND operation between the cover pixel and \( h \) value using in Equation (1)

**Step 5:** Apply the logical OR operation between the SDC and C’ bits to hide the data in the cover pixels, to produce the S using Equation (2)

**Step 6:** Repeat step 3 and 4 for all remaining chunks of SDC until the whole SDC are embedded into the C’.

**Step 7:** Perform data hiding using K-LSB.

**Step 8:** End Stage I. Return the Stego image S.

**End**

**Algorithm 2: Enhanced Flipping Technique to Reduce variability in Image Steganography**

Input: Cover Image C and Stego image (S).

**Output:** Stego Image (FPS)

**Begin**

**Step 1:** Read the cover image C and Stego Image (S).

**Step 2:** Set the threshold value, T.

**Step 3:** For each chuck, calculate the absolute difference (AD) between the S and C image using Equation (3)

**Step 4:** If the absolute difference (AD) is higher than the threshold T, then we flip the 5th stego pixel (FSP).

**Step 5:** Repeat steps 2 and 3 for all remaining chunks.

**Step 6:** Return the flipped stego pixels (FSP).

**End**

**IV. EXPERIMENTAL RESULT AND DISCUSSION**

The proposed method was simulated using MATLAB 2013a. We conducted all the experiments on a system with 3.2GHz, 6-core 8th-gen i7 processor, and 8GB RAM. The standard dataset images were taken from the USC-SIPI image dataset [34]. Fifteen secret data were randomly generated using MATLAB. Experimental results were evaluated using visual, statistical, and security analyses to measure the performance of the proposed method.

**A. VISUAL ANALYSIS**

This analysis measures the visual quality of the stego image after the data hiding [35] using Mean Square Error (MSE). MSE represents the error between the cover and stego images generated due to data hiding in the stego image, as shown in Equation (6)[35].
\[ \text{MSE} = \frac{1}{MN} \sum_{i=1}^{A} \sum_{j=1}^{B} (C_{(i,j)} - S_{(i,j)})^2 \]  

where C and S denote the cover and stego image, i,j represents the row and column of the stego image, and A,B represents the row and column of the cover image.

The resolution of the cover image was 256×256. Table 3 shows that the proposed flipping method affects the visual quality of stego image after the K-LSB data hiding. The experiment was conducted based on two, three, and four (LSB) bits. Table 3 shows that the average MSE value for the 2-bit LSB is 1.24, 3-bit LSB is 5.17, and 4-bit LSB is 20.982, while the proposed flipping method is the average MSE value for the 2-bit is 0.98, 3-bit is 3.94, and 4-bit is 15.91. Indeed, when comparing the proposed method (2-bit) with GA [12] (2-bit) and BO algorithms [14], the proposed method outperformed the GA and BO algorithms that were used to reduce the visual quality of the stego image. Our MSE value is 0.98, whilst GA and BO algorithms were 2.37 and 1.48, respectively.

**TABLE 3. The visual quality analysis of the proposed flipping method with existing methods on the USC-SIPI dataset**

| Cover Image | GA [12] | BO algorithm [14] | Before applying the proposed flipping | After applying the proposed flipping |
|-------------|---------|-------------------|---------------------------------------|-------------------------------------|
| 2-bit       | 2.35    | NA                | 1.24                                  | 1.00                                |
| 3-bit       | NA      | NA                | 5.22                                  | 4.09                                |
| 2-bit       | 2.40    | 1.48              | 1.24                                  | 0.99                                |
| 3-bit       | NA      | NA                | 5.22                                  | 3.99                                |
| 2-bit       | 2.38    | 1.48              | 1.25                                  | 0.96                                |
| 3-bit       | NA      | NA                | 5.09                                  | 3.76                                |
| 3-bit       | NA      | NA                | 19.15                                 | 14.22                               |

**B. STATISTICAL PARAMETERS**

This experiment was conducted to demonstrate the analytical performance of the stego image produced by the proposed flipping method and K-LSB, and various statistical parameters were measured for the proposed method.

The following results are shown in Table 4, which shows how the proposed flipping method affects the visual quality of the stego image after K-LSB data hiding. The experiment was conducted based on two, three, and four (LSB) bits. Table 4 shows the average MSE value for the (2-bit LSB) was 1.23, for (3-bit LSB) was 4.96, and for (4-bit LSB) was 20.74, while the proposed flipping method after the data hiding process was applied on the stego image, the average MSE value for the (2-bit LSB) was 0.98, for (3-bit LSB) was 3.91, and for (4-bit LSB) was 15.25. As a result, the MSE of the stego image after applying the proposed flipping method improved the stego image variability, owing to the lower MSE value.

The above results show that the proposed flipping method achieved a lower MSE rate for stego images than the K-LSB. The proposed flipping method flips at the 5th bit after data hiding, which leads to less variability than the original stego image results.

**TABLE 4. K-bits LSB stego image for before and after applying the proposed flipping method on the USC-SIPI dataset**

| Cover Image | Stego 2-bit | Stego 3-bit | Stego 4-bit |
|-------------|-------------|-------------|-------------|
| Before Flip | After Flip  | Before Flip | After Flip  |
| Before Flip | After Flip  | Before Flip | After Flip  |
| Lena        | 1.24        | 0.34        | 5.24        | 4.09        | 22.28       | 17.37       |
| Baboon      | 1.25        | 0.34        | 5.24        | 3.97        | 21.18       | 16.01       |
| Barbara     | 1.24        | 0.34        | 5.24        | 4.00        | 21.20       | 16.83       |
| Pepper      | 1.20        | 0.33        | 5.02        | 3.73        | 19.13       | 14.20       |
| Boat        | 1.25        | 0.35        | 5.16        | 3.94        | 21.25       | 15.99       |
| Female      | 1.23        | 0.34        | 4.96        | 3.76        | 19.48       | 13.29       |
| Couple      | 1.23        | 0.34        | 5.00        | 3.78        | 19.52       | 12.49       |
| House       | 1.27        | 0.42        | 6.22        | 5.36        | 29.65       | 26.49       |
| Tree        | 1.20        | 0.33        | 5.33        | 4.16        | 22.85       | 18.05       |
| Aeroplane   | 1.26        | 0.34        | 5.09        | 3.93        | 21.12       | 14.50       |
| Jellybeans  | 1.23        | 0.31        | 5.33        | 4.01        | 20.14       | 13.89       |
| Splash      | 1.24        | 0.34        | 5.22        | 3.99        | 21.08       | 18.98       |
| Oakland     | 1.24        | 0.34        | 5.27        | 3.97        | 21.21       | 16.24       |
| Earth       | 1.24        | 0.34        | 5.63        | 3.86        | 19.82       | 15.20       |
| Flower      | 1.10        | 0.27        | 4.32        | 3.06        | 14.53       | 9.92        |

Figure 4 summarizes the comparison for the MSE values between the stego images data hiding and after data hiding and applying the proposed flipping method.
The following results are shown in Table 5 also shows how the proposed flipping method affects the visual quality of the stego image after the K-LSB data hiding executes. The experiment was conducted on 2, 3, and 4 (LSB) bits. Table IV shows the average PSNR value for the (2-bit LSB) was 47.24dB, for (3-bit LSB) was 40.91 dB, and for (4-bit LSB) was 34.94 dB. After the data hiding process, the proposed flipping method was applied to the stego image. Their average PSNR value for the (2-bit LSB) was 48.18db, for (3-bit LSB) was 42.20dB, and for (4-bit LSB) was 35.99db. As a result, the PSNR of the stego image after applying the proposed flipping method outperformed the stego image after K-bits LSB data hiding, owing to achieving a higher PSNR value. This value is defined in Equation (7)[36].

$$\text{PSNR} = 10 \log_{10} \frac{\text{Max}^2}{\text{MSE}},$$  
(7)  
where \( \text{Max} \) represents the maximum value represented in the cover image while \( \text{MSE} \) denotes the mean square error.

As shown in Table 4, the PSNR values of the proposed flipping method are much higher than the image quality standard of 30 dB, which proves that stego images have high image quality. This occurs because the proposed flipping method replaces the K least significant bits of each pixel in the cover image with secret k-bits from the secret message and flipping the 5th after data hiding. In contrast, the K-bits LSB data hiding is slightly lower than that of the proposed flipping method because of the cover image with the secret message.

| Cover | Stego 2bit | Stego 3-bit | Stego 4-bit |
|-------|------------|------------|------------|
|       | Before     | After      | Before     | After      | Before     | After      |
| Lena  | 47.17      | 52.81      | 40.93      | 42.01      | 34.65      | 35.73      |
| Baboon| 47.15      | 52.78      | 40.92      | 42.10      | 34.87      | 36.08      |
| Barbara| 47.17     | 52.81      | 40.92      | 42.10      | 34.86      | 35.86      |
| Pepper| 47.31      | 52.85      | 41.12      | 42.40      | 35.31      | 36.60      |
| Boat  | 47.15      | 52.68      | 41.00      | 42.17      | 34.85      | 36.09      |
| Female| 47.23      | 52.78      | 41.17      | 42.37      | 35.23      | 36.89      |
| Couple| 47.22      | 52.79      | 41.13      | 42.34      | 35.22      | 37.16      |
| House | 47.06      | 51.82      | 40.18      | 40.83      | 33.41      | 33.89      |
| Tree  | 47.31      | 52.95      | 40.85      | 41.93      | 34.54      | 35.56      |
| Aeroplane| 47.11     | 2.74       | 41.06      | 42.18      | 34.88      | 36.51      |
| Jellybeans| 47.20    | 53.12      | 40.86      | 42.09      | 35.08      | 36.70      |
| Splash| 47.16      | 52.82      | 40.94      | 42.11      | 37.89      | 35.34      |
| Oakland| 47.18     | 52.82      | 40.90      | 42.13      | 34.86      | 36.02      |
| Earth | 47.18      | 52.75      | 40.61      | 42.25      | 35.15      | 36.31      |
| Flower| 47.70      | 53.70      | 41.77      | 43.26      | 36.50      | 38.16      |
| Average| 47.22     | 52.81      | 40.95      | 42.15      | 35.15      | 36.19      |

Embedding Capacity (EC) is the performance measurement parameter used to evaluate the proposed flipping method in this experiment. Table V shows that the EC values achieved by the proposed flipping method are very high. The experiment was conducted on the dataset consisting of 15 images, and the experiment was based on 2, 3, and 4 (LSB) bits. Table 6 shows the average EC value for the (2-bit LSB) was 13107 bits, for (3-bit LSB) was 196608 bits, and for (4-bit LSB) was 262144; after the data hiding process, the proposed flipping method was applied to the stego image, the average EC value for the (2-bit ) was 524288 bits, for (3-bit ) was 786432 bits, and for (4-bit) was 1048576bits. As a result, the PSNR of the stego image after applying the proposed flipping method outperformed the stego image after K-bits LSB data hiding because it achieved a higher value of EC value, as presented in Equation (8)[37].

$$EC(\text{bpp}) = \frac{\text{No.of embedding bits}}{W \times H}$$  
(8)  
where \( W \) and \( H \) are the width and height of the image, respectively.

It is clearly observed that the proposed flipping method provides high-quality performance with various hiding capacities. In addition, the EC is fixed for each image in the dataset, and it depends on the method used. The proposed flipping method achieved a PSNR of 48.16 dB when hiding 524288 bits. Furthermore, the proposed scheme attains the minimum PSNR value, 35.99 dB, hiding approximately 1048576 using all 4 LSBs. The hiding of this large number of bits indicates a very high hiding capacity. Therefore, the proposed flipping method provides a high embedding capacity.

| Cover | Stego 2bit | Stego 3-bit | Stego 4-bit |
|-------|------------|------------|------------|
|       | Before     | After      | Before     | After      | Before     | After      |
| Lena  | 13107      | 524288     | 196608     | 786432     | 262144     | 1048576    |

The experiment was conducted based on 2, 3, and 4 (LSB) bits. Table 7 shows the average entropy value for the (2-bit LSB) was 7.2655, for (3-bit LSB) was 7.28008, and for (4-bit LSB) was 7.28008, and for (3-bit LSB) was 7.28008, and for (4-bit LSB) was 7.29908. As a result, the stego image's entropy after applying the proposed flipping method outperformed the stego image after K-bits LSB data hiding due to the obtained entropy results in Table 7. These images show that the entropy of the stego-image is slightly greater than the cover image, and this is because more hidden information is added to the cover image without a significant change in pixel values. The entropy value is defined in Equation (9)[38].

$$E(P) = \sum_{i=1}^{A} p(i) \log p(i)$$  
(9)  
where \( p(i) \) is the probability density function of a given image at intensity level, \( l \) and \( A \) is the total number of gray levels in the image, the highest value of entropy indicates good quality and information about an image.
ping method with KLSB bits. Thus, this parameter
01628, and for (4
ions.

\[ P \]

| Image          | Cover Image Entropy | Stego 2bit | Stego 3bit | Stego 4bit |
|----------------|---------------------|-----------|-----------|-----------|
| Lena           | 6.9798              | 6.9834    | 6.9741    | 6.9569    |
| Baboon         | 7.7614              | 7.7659    | 7.7554    | 7.6833    |
| Barbara        | 7.5150              | 7.5188    | 7.5092    | 7.4901    |
| Pepper         | 7.1615              | 7.1941    | 7.1879    | 7.2122    |
| Boat           | 7.1763              | 7.1837    | 7.1744    | 7.1894    |
| Female         | 6.8334              | 6.8867    | 6.8892    | 6.8879    |
| Couple         | 6.2105              | 6.2150    | 6.2161    | 6.2304    |
| Tree           | 6.3887              | 6.4468    | 6.4515    | 6.6157    |
| Aeroplane      | 6.9827              | 6.9903    | 6.9921    | 7.0411    |
| Jellybeans     | 6.5953              | 6.6233    | 6.6130    | 6.6407    |
| Splash         | 6.2126              | 6.2278    | 6.2261    | 6.2016    |
| Oakland        | 6.0453              | 6.0491    | 6.0401    | 5.4696    |
| Earth          | 6.7365              | 6.7464    | 6.7390    | 6.7228    |
| Flower         | 6.5069              | 6.5301    | 6.5418    | 6.5064    |
| Average        | 6.7616              | 6.7718    | 6.7767    | 6.7471    |

C. SECURITY ANALYSIS

This section discusses the security analysis parameters for the proposed method in steg flips method and stego K-LSB. The first parameter of the two measured security parameters is KL divergence, whereas the second parameter is histogram analysis. The USC-SIPI image dataset was used for experimental purposes for both security parameters.

1) KL DIVERGENCE

Solomon Kullback and Richard Leibler invented the KL Divergence parameter. It is measured to determine the difference between two probability distributions. The histogram distribution varies when data are hidden in the cover image in steganography. Thus, this parameter calculates divergence between the cover and stego images. It is calculated using Equation (10)[40].

\[ K(\pi \parallel \phi) = \sum_i \pi_i \log \frac{\pi_i}{\phi_i} \] (10)

where \( i \) denotes subscript variable and its value varies from 0 to 255. \( P \phi \) denotes the probability distribution of the histograms of the cover and stego images.

In the ideal case, KL divergence should be zero. The experiment was conducted based on 2, 3, and 4 (LSB) bits. Table VII shows the average KL divergence for the (2-bit LSB) was 0.00336, for (3-bit LSB) was 0.01628, and for (4-bit LSB) was 0.03872, also conducted based on 2, 3, and 4 (LSB) bits, but after the data hiding process, the proposed flipping method was applied on the stego image, the average KL divergence value for the (2-bit) was 0.00224, for (3-bit) was 0.01226, and for (4-bit) was 0.03454. As a result, the KL divergence of the stego image after applying the proposed flipping method is the minimum value of KL divergence compared to the K-LSB technique stego. It has proven that embedding to the 4LSB with a small statistical probability can achieve higher steganographic security, and the minimum value of KL probability statistics is more accurate for steganographic security measures.

### Table 7. The entropy of the proposed flipping method with KLSB bits on the USC-SIPI dataset

| Cover Images | Cover Image Entropy | Stego 2bit | Stego 3bit | Stego 4bit |
|--------------|---------------------|-----------|-----------|-----------|
| Lena         | 6.9798              | 6.9834    | 6.9741    | 6.9569    |
| Baboon       | 7.7614              | 7.7659    | 7.7554    | 7.6833    |
| Barbara      | 7.5150              | 7.5188    | 7.5092    | 7.4901    |
| Pepper       | 7.1615              | 7.1941    | 7.1879    | 7.2122    |
| Boat         | 7.1763              | 7.1837    | 7.1744    | 7.1894    |
| Female       | 6.8334              | 6.8867    | 6.8892    | 6.8879    |
| Couple       | 6.2105              | 6.2150    | 6.2161    | 6.2304    |
| Tree         | 6.3887              | 6.4468    | 6.4515    | 6.6157    |
| Aeroplane    | 6.9827              | 6.9903    | 6.9921    | 7.0411    |
| Jellybeans   | 6.5953              | 6.6233    | 6.6130    | 6.6407    |
| Splash       | 6.2126              | 6.2278    | 6.2261    | 6.2016    |
| Oakland      | 6.0453              | 6.0491    | 6.0401    | 5.4696    |
| Earth        | 6.7365              | 6.7464    | 6.7390    | 6.7228    |
| Flower       | 6.5069              | 6.5301    | 6.5418    | 6.5064    |
| Average      | 6.7616              | 6.7718    | 6.7767    | 6.7471    |

### Table 8. The KL divergence between the cover and stego image of the K-bits LSB stego image before and after applying the proposed flipping method on the USC-SIPI dataset

| Cover Image | Stego 2bit | Stego 3bit | Stego 4bit |
|-------------|-----------|-----------|-----------|
| Lena        | 0.0035    | 0.0023    | 0.0164    |
| Barbara     | 0.0033    | 0.0022    | 0.0160    |
| Baboon      | 0.0034    | 0.0024    | 0.0162    |
| Female      | 0.0032    | 0.0021    | 0.0165    |
| Average     | 0.00336   | 0.00224   | 0.01628   |

2) HISTOGRAM ANALYSIS

Histogram analysis is the most preferred parameter for security purposes. The data hiding in the cover image changes the pixel values, and thus, histogram distribution varies [40]. In an ideal scenario, the histogram should be identical. For experimental purposes, Table 9 shows the histogram analysis of the cover and stego images for the proposed flipping methods. The result shows that the stego image histogram is similar to the cover image histogram for 2-bit and varies for a 3-bit and 4-bit method. The horizontal axis of the histogram denotes the tonal deviations, while the vertical axis denotes the number of pixels in that specific tone. It is observed that the histograms of the carrier and stego images provide a negligible difference. The proposed flipping method can sustain statistical attacks as secret pixels can not be observed from the histogram.
| Cover Images | Cover Image Histogram | Stego Image Histogram for 2-bit | Stego Image Histogram for 3-bit | Stego Image Histogram for 4-bit |
|--------------|-----------------------|-------------------------------|-------------------------------|-------------------------------|
| [Image]      | [Image]               | [Image]                       | [Image]                       | [Image]                       |
| [Image]      | [Image]               | [Image]                       | [Image]                       | [Image]                       |
| [Image]      | [Image]               | [Image]                       | [Image]                       | [Image]                       |
| [Image]      | [Image]               | [Image]                       | [Image]                       | [Image]                       |

**TABLE 9.** HISTOGRAM ANALYSIS COVER AND STEGO IMAGE PERFORMANCE OF THE PROPOSED METHOD ON THE USC-SIPI DATASET
D. COMPARATIVE ANALYSIS WITH OPTIMIZATION AND FLIPPING METHODS

To validate the performance of the proposed flipping method in reducing variability, it compared with other optimization and flipping methods that were used to improve the stego image variability. As shown in Table 11 and Table 12, the proposed flipping method outperformed other methods in PSNR and EC. The proposed method achieved higher PSNR and was compared with optimization and state-of-the-art flipping methods. The average PSNR for the proposed flipping method was 48.13 dB, while for GA [12], BO algorithms [14], and the basic flipping method were 45.14dB, 44.96 dB, respectively.

Furthermore, the basic flipping method uses the basic flipping method on the secret data and before the data hiding process, improving the visual quality and variability. In the proposed method, the flipping method is used after the data hiding process to improve visual quality and variability. In the second experiment, the EC values of the proposed method were higher than those of state the art flipping method. The basic flipping method values of 1 Bpp, whereas the proposed method obtained 2 Bpp in the same EC with optimization algorithms. The proposed method shows that the PSNR is more effective than previous flipping. In addition, the proposed method avoids certain problems in state-of-the-art. It can achieve the highest EC without minimizing the visual degradation of the stego image.

In order to validate the performance of the proposed flipping method in reducing variability, it compared with other optimization and flipping methods used to improve the stego image variability. Tables 10 to 12 show that the proposed method for PSNR, MSE, and EC was better than the other methods' results. The proposed method also achieved the best performance in MSE; the average MSE for the proposed method was 0.35, compared to 2.37, 1.48, and 1.24, respectively, for GA, BO algorithms, and K-bits LSB stego image are shown in Table 12. Moreover, the average PSNR for the proposed flipping method was 52.64 dB, compared to 45.14dB, 44.96 dB, 47.37dB, and 47.16 dB, respectively, for GA, and BO algorithms, which are shown in Table 10. At the same time, the proposed method has obtained 2 bpp in the same EC with GA and BO algorithms shown in Table 12.

Moreover, Table 13 to 15 illustrates the proposed method results for PSNR, MSE, and EC with existing flipping methods. The proposed method achieved higher PSNR and lower MSE and higher EC compared with the state-of-the-art flipping methods [16][17][18]. The MSE, PSNR, and EC values were 0.35, 52.64 dB, and two bpp for the proposed method, compared to the existing flipping methods such as [16][17][18] values of 1.17, 47.32db 1.5 bpp, 1.24, 47,16db, and 1bpp, respectively, are presented in Figures 5 until 9. The experimental study points out that the proposed method is better than the existing method in terms of higher visual quality, as indicated by the high PSNR of hiding secret message bits in the image, thus reducing the MSE with the same embedding capacity shown in Table 12.

| COVER IMAGES | GA ALGORITHM [12] | BO ALGORITHM [14] | PROPOSED FLIPPING METHOD |
|--------------|-------------------|-------------------|--------------------------|
| Lena         | 2.3               | 1.4               | 0.3                      |
| Baboon       | 2.4               | 1.4               | 0.3                      |
| Pepper       | 2.3               | 1.4               | 0.3                      |
| Barbara      | NA                | NA                | 0.3                      |
| Aeroplane    | NA                | NA                | 0.3                      |
| Boat         | NA                | NA                | 0.3                      |
| House        | NA                | NA                | 0.4                      |
| Average      | 2.3               | 1.4               | 0.3                      |

FIGURE 5. The MSE performance of the proposed flipping method, GA and BO algorithms on the USC-SIPI dataset

| COVER IMAGE | GA ALGORITHM [12] | BO ALGORITHM [14] | PROPOSED FLIPPING METHOD |
|-------------|-------------------|-------------------|--------------------------|
| Lena        | 45.1              | 45.1              | 52.8                     |
| Baboon      | 45.1              | 44.31             | 52.7                     |
| Pepper      | 45.1              | NA                | 52.8                     |
| Barbara     | NA                | NA                | 52.8                     |
| Aeroplane   | NA                | 45.3              | 52.7                     |
| Boat        | NA                | NA                | 52.6                     |
| House       | NA                | NA                | 51.8                     |
| Average     | 45                | 44.9              | 52.6                     |
Based on the results shown in Table 16, the proposed method outperformed optimization and flipping methods to
improve visual quality and variability. The proposed method is used after the data hiding process, improve visual quality and variability. The proposed flipping method achieved a lower MSE than optimization and basic flipping methods. The average MSE for the proposed flipping method was 1.00, whereas for GA, BO, and basic flipping methods achieved approximately 2.37, 1.48, and 1.24, respectively. The proposed method shows that the PSNR and MSE are more effective than previous flipping.

The Astuti flipping method [16] works by flipping the secret bits in the opposite direction. The flipping process results in a mirror version of secret bits. Their secret data bits also reduced the variability. In addition, the reversible bit flipping method for concealing 2 bits of secret data in a segment of 2 pixels. The last LSB bits of the pixels were exploited in each segment to conceal the secret data at the first embedding layer. However, flipping is performed on the stego image (after data hiding), not on the secret message in the proposed flipping method. Hence the chance of attacks is avoided, and embedding capacity is higher.

To evaluate the proposed flipping method and compare it with others, even when they are used in the secret message. A new experiment was conducted. In this experiment, the Astuti flipping method [16] was applied based on (2 bit) data hiding, and then the proposed flipping method was applied after data hiding, which clearly improves the results of the Astuti method, as shown in Table 16. The results show that the use of the proposed method after basic flipping method results showed that the average PSNR, MSE, and EC were improved. The results for the basic flipping method were average PSNR (47.14 dB), MSE (1.24), and EC (1). In comparison, the results after applying the proposed method were PSNR (48.16 dB), MSE (0.9912), and EC (2Bpp). It is observed that the basic flipping method with the proposed method provides high-quality performance, are presented in Figures 9 and 10.

Moreover, the proposed flipping method improved stego image quality with less computational complexity. The proposed flipping method consumed the second least processing time, 1.17 seconds, while the state-of-the-art flipping of the data hiding spent 2.66 seconds.

The performance of the proposed method is evaluated using a single simulation execution applying two iterations for achieving a target stable stego image. Thus, the flipping proposed method if the absolute difference (AD) is higher than the threshold $T$, then we flip the adjacent pixel of the $k$-bits LSB technique is flipped (5th bit), while the state-of-the-art flipping method flip bits if the change in bits per pixel is more than 50%. Hence it needs to be selective in choosing the cover image are presented in Figure 11. In corresponding to Figure 11, it shows that the proposed method takes lesser time for data hiding as compared to existing methods, and Table 11 shows the PSNR parameter for visual quality better than existing methods.

**TABLE 16.** PSNR, MSE, and EC of the Basic flipping using the proposed flipping method and state of the art flipping method on the USC-SIPI dataset

| Image    | Basic flipping method [16] | Basic flipping method with the proposed flipping method |
|----------|-----------------------------|--------------------------------------------------------|
|          | PSNR | MSE | EC | PSNR | MSE | EC |
| Lena     | 1.2488 | 47.16 | 262144 | 1.0040 | 48.11 | 52.4288 |
| Baboon   | 1.2502 | 47.16 | 262144 | 0.9906 | 48.17 | 52.4288 |
| Barbara  | 1.2500 | 47.18 | 262144 | 0.8614 | 48.77 | 52.4288 |
| Pepper   | 1.2006 | 47.33 | 262144 | 0.9599 | 48.30 | 52.4288 |
| Boat     | 1.2633 | 47.11 | 262144 | 1.0117 | 48.08 | 52.4288 |
| Female   | 1.2325 | 47.22 | 262144 | 0.9753 | 48.23 | 52.4288 |
| Couple   | 1.2310 | 47.22 | 262144 | 0.9776 | 48.22 | 52.4288 |
| House    | 1.2888 | 47.02 | 262144 | 1.0715 | 47.83 | 52.4288 |
| Tree     | 1.2107 | 47.30 | 262144 | 0.9755 | 48.23 | 52.4288 |
| Aeroplane| 1.2563 | 47.13 | 262144 | 0.9891 | 48.17 | 52.4288 |
| Jellybeans | 1.2395 | 47.19 | 262144 | 0.9736 | 48.23 | 52.4288 |
| Splash   | 1.2472 | 47.17 | 262144 | 1.0004 | 48.12 | 52.4288 |
| Oakland  | 1.2499 | 47.16 | 262144 | 0.9963 | 48.14 | 52.4288 |
| Earth    | 1.2434 | 47.18 | 262144 | 1.0034 | 48.11 | 52.4288 |
| Flower   | 1.1071 | 47.68 | 262144 | 0.8626 | 48.77 | 52.4288 |
| Average  | 1.23  | 47.21 | 262144 | 0.97  | 48.23 | 52.4288 |
well-known measurements used to compare the differences between the original and their changes, is presented in Figure 12. It is impossible to judge which is better to be inferred from knowing which one is original and which has undergone additional treatment such as data compression. The SSIM index is defined as in Eq (11):

$$SSIM(x,y) = \frac{(2\mu_x \mu_y + c_1)(2\sigma_{xy} + c_2)}{\mu_x^2 + \mu_y^2 + \sigma_x^2 + \sigma_y^2 + c_1}$$

(11)

where \(\mu_x\) and \(\mu_y\) are defined by illuminating each image in the x and y direction, \(\sigma_x\) and \(\sigma_y\) are the standard deviations and the contrast estimation of the signal, \(c_1\), and \(c_2\) are small constants involved in controlling instability when either is very close to zero. However, while respecting the standards of the global quality index, these constants are ignored and equal to zero. Therefore, the closer the SSIM is to the unity, the better the image performance it can reach [41].

$FIGURE~11.~The~Computational~complexity~of~the~proposed~flipping~method~with~existing~puzzles$

In addition, to statistically analyze the results obtained for the proposed flipping method, the Friedman test is the best-known statistical method for testing the performance differences between more than two algorithms [41]. Friedman’s test was used to assess the independence of repeated experiments resulting in ranks, the ranks PSNR computed by this test for the K-bits LSB stego image, basic flipping method, and basic flipping method with the proposed flipped method are tabulated in Table 17. It can be seen in the table that the proposed flipping method was successfully achieved as the best performing algorithm, whereas the lowest-performing algorithm was the basic flipping method.

The Friedman test was used to conduct statistical comparisons between the proposed flipping method and other flipping method variants. Two comparisons were conducted using MSE and PSNR dimensions to evaluate the results. Table 17 shows the ranks achieved by this test for both sets of comparisons. It can be seen in these tables that the proposed flipping method ranked and basic flipping method with the proposed flipped method first, followed by basic flipping method, K-bits LSB stego image, and basic flipping method in both cases. The p-values calculated using the statistics from the Friedman’s test were 0.002 and 0.018 in the MSE and PSNR cases, respectively, as shown in Table 17. These p-values suggest significant differences in the performance of the methods considered.

$TABLE~17.~Average~rankings~PSNR~rankings~of~the~K-bits~LSB~stego~image~for~before~and~after~applying~the~proposed~flipping~method~to~the~USC-SIPI~dataset$

| Algorithms                  | Ranking MSE | Ranking PSNR |
|-----------------------------|-------------|--------------|
| Proposed flipping method    | 6.000       | 1.500        |
| k-LSB                       | 22.000      | 5.500        |
| basic flipping              | 22.000      | 5.500        |
| p-values                    | 0.002       | 0.018        |

In addition, the structurality index (SSIM) is another measurement tool used to measure the quality of imperceptibility in steganographic images [41]. SSIM are two
distortion when hiding a large message. Optimization algorithms are used before the embedding process, and the secret data are mapped to the optimal pixels using BO or GA optimization algorithms to reduce the mean square error (MSE). The mapped data are then embedded into the lower bits of host pixels using modulus function and a systematic and reversible algorithm. Optimization algorithms strongly influence visual quality. The results are better with small data hiding, which has less variability.

The flipping method was able to reduce variability and avoid long iterations. However, their flipping method consumes less embedding capacity. In the flipping method before the embedding process, the secret data are flipped 2-bits of secret data to reduce the variability. Then, the flipping bits of the message are embedded in the smallest bits in each cover image. However, it was only able to produce a high-quality stego image. It failed to secure data hiding and low embedding capacity.

The proposed flipping method aims to resolve the problem of variability resulting from data hiding by depending on the flipping technique to adjust the cover pixels after hiding the data bits that provide reduced variability. Based on the previous result in [16], the proposed flipping method achieved a higher embedding capacity, similar to the optimized data hiding technique. Simultaneously, it solves the problem of variability caused by a flipping method. The experimental results showed that our method could solve problems and special cases better than the flipping method.

The above analysis shows that the proposed flipping method is more effective than previous methods. Although each of the previous methods was able to deal very well with particular challenges, the proposed flipping method was able to deal very well with specific challenges. The proposed flipping method solves the problems of the data hiding technique that endures a higher variability and less embedding capacity. The outcome shows that optimization algorithms gave good results for higher embedding capacity but required long iterations. The proposed flipping method provided the best results for reducing variability, higher embedding capacity, and more security for all images.

V. CONCLUSION

In this paper, a data hiding approach is designed based on the flipping method to enhance imperceptibility, embedding capacity, and reduce variability. Secret data is initially hidden in the cover image using the k-bit LSB method to achieve this goal. After that, an absolute difference is found between the cover and stego image. If the absolute difference is higher than the threshold value, the adjacent bit of k-bit LSB pixel is flipped. The flipping process reduces the absolute difference and enhances the impermeability of the stego image. The simulation results show that the proposed method for both PSNR and MSE is better than state-of-the-art results based on the experimental results. The average PSNR was 52.64 dB for the proposed method, compared to 45.11 db and 44.64db, respectively, for GA and BO methods. Next, the average MSE was 0.53 for the proposed method compared to 2.37 and 1.48, respectively, for GA and BO methods. Finally, the proposed flipping method was compared with state-of-the-art methods that produced better EC.

ACKNOWLEDGMENT

This work was supported by the Ministry of Higher Education, Malaysia, under research grant LRGS-1-2019-UKM-UKM-2-7 and Universiti Kebangsaan Malaysia.

REFERENCES

[1] S. Kamil, M. Ayob, S. N. H. Sheikh Abdullah, and Z. Ahmad, “Optimized Data Hiding in Complemented or Non-Complemented Form in Video Steganography,” Proc. 2018 Cyber Resil. Conf. CRC 2018, pp. 1–4, 2019, doi: 10.1109/CR.2018.8626871.

[2] S. Kamil, M. Ayob, S. N. H. Sheikh Abdullah, and Z. Ahmad, “Challenges in Multi-Layer Data Security for Video Steganography Revisited,” Asia-Pacific J. Inf. Technol. Multimed., vol. 07, no. 02(02), pp. 53–62, 2018, doi: 10.17576/apjitm-2018-0702(02)-05.

[3] T. Denemark, M. Boroumand, and J. Fridrich, “Steganalysis Features for Content-Adaptive JPEG Steganography,” IEEE Trans. Inf. Forensics Secur., vol. 11, no. 8, pp. 1736–1746, 2016, doi: 10.1109/TIFS.2016.2555281.

[4] Y. Wu, Y. Xiang, Y. Guo, J. Tang, and Z. Yin, “An Improved Reversible Data Hiding in Encrypted Images Using Parametric Binary Tree Labeling,” IEEE Trans. Multimed., vol. 22, no. 8, pp. 1929–1938, 2020, doi: 10.1109/TMM.2019.2952979.

[5] S. A. El Rahman, “A comparative analysis of image steganography based on DCT algorithm and steganography tool to hide nuclear reactors confidential information,” Comput. Electr. Eng., vol. 70, pp. 380–399, 2018, doi: 10.1016/j.compeleceng.2016.09.001.

[6] S. Weng, Y. Chen, B. Ou, C. C. Chang, and C. Zhang, “Improved K-Pass Pixel Value Ordering Based Data Hiding,” IEEE Access, vol. 7, pp. 34570–34582, 2019, doi: 10.1109/ACCESS.2019.2904174.

[7] M. A. AlZain and J. F. Al-Amri, “Application of data steganographic method in video sequences using histogram shifting in the discrete wavelet transform,” Int. J. Appl. …, vol. 13, no. 8, pp. 6380–6387, 2018. [Online]. Available: https://www.ripublication.com/ijaer18/ijaerv13n8_106.pdf.

[8] T. Li, H. Li, L. Hu, and H. Li, “A Reversible
Steganography Method with Statistical Features Maintained Based on the Difference Value,” IEEE Access, vol. 8, pp. 12845–12855, 2020, doi: 10.1109/ACCESS.2020.2964830.

[9] L. Guo, J. Ni, and Y. Q. Shi, “Uniform embedding for efficient JPEG steganography,” IEEE Trans. Inf. Forensics Secur., vol. 9, no. 5, pp. 814–825, 2014, doi: 10.1109/TIFS.2014.2312817.

[10] M. Hussain, A. W. A. Wahab, Y. I. Bin Idris, A. T. S. Ho, and K. H. Jung, “Image steganography in spatial domain: A survey,” Signal Process. Image Commun., vol. 65, no. December 2017, pp. 46–66, 2018, doi: 10.1016/j.image.2018.03.012.

[11] M. M. Sadek, A. S. Khalifa, and M. G. M. Mostafa, “Video steganography: a comprehensive review,” Multimedia Tools Appl., vol. 74, no. 17, pp. 7063–7094, 2015, doi: 10.1007/s11431-014-1952-z.

[12] H. R. Kanan and B. Nazeri, “A novel image steganography scheme with high embedding capacity and tunable visual image quality based on a genetic algorithm,” Expert Syst. Appl., vol. 41, no. 14, pp. 6123–6130, 2014, doi: 10.1016/j.eswa.2014.04.022.

[13] V. Snasel, P. Kromer, J. Safarik, and J. Platos, “JPEG steganography with particle swarm optimization accelerated by AVX,” Concurr. Comput., vol. 32, no. 8, pp. 1–11, 2020, doi: 10.1002/cpe.5448.

[14] A. M. Molaei and A. Ebrahimzadeh, “Optimal steganography with blind detection based on Bayesian optimization algorithm,” Pattern Anal. Appl., vol. 22, no. 1, pp. 205–219, 2019, doi: 10.1007/s10243-018-00773-0.

[15] U. Mehboob, J. Qadir, S. Ali, and A. Vasilakos, Genetic algorithms in wireless networking: techniques, applications, and issues, vol. 20, no. 6. Springer Berlin Heidelberg, 2016.

[16] E. Z. Astuti, D. R. I. Moses Setiadi, E. H. Rachmawanto, C. A. Sari, and M. K. Sarker, “Flipping the Message Bits to Increase Imperceptibility in the Least Significant Bit Image Steganography,” J. Phys. Conf. Ser., vol. 1201, no. 1, 2019, doi: 10.1088/1742-6596/1201/1/012030.

[17] A. K. Sahu, G. Swain, and E. Suresh Babu, “Digital image steganography using bit flipping,” Cybern. Inf. Technol., vol. 18, no. 1, pp. 69–80, 2018, doi: 10.2478/cait-2018-0006.

[18] R. Kumar and S. Chand, “A reversible data hiding scheme using bit flipping strategy,” J. Discret. Math. Sci. Cryptogr., vol. 19, no. 2, pp. 331–345, 2016, doi: 10.1080/09720529.2015.1085743.

[19] S. Singh, “Adaptive PVD and LSB based high capacity data hiding scheme,” Multimed. Tools Appl., vol. 79, no. 25–26, pp. 18815–18837, 2020, doi: 10.1007/s11422-020-08745-5.

[20] S. Solak, “High embedding capacity data hiding technique based on emsd and LSB substitution algorithms,” IEEE Access, vol. 8, pp. 16653–166524, 2020, doi: 10.1109/ACCESS.2020.3023197.

[21] S. Rahman et al., “A novel approach of image steganography for secure communication based on LSB substitution technique,” Comput. Mater. Cont., vol. 64, no. 1, pp. 31–61, 2020, doi: 10.32604/cm.cmc.2020.09186.

[22] R. Wazirali, W. Alasmr, M. M. E. A. Mahmoud, and A. Alhindi, “An Optimized Steganography Hiding Capacity and Imperceptibility Using Genetic Algorithms,” IEEE Access, vol. 7, pp. 133496–133508, 2019, doi: 10.1109/ACCESS.2019.2941440.

[23] M. Khodaei and K. Faez, “New adaptive steganographic method using least-significant-bit substitution and pixel-value differencing,” IET Image Process., vol. 6, no. 6, pp. 677–686, 2012, doi: 10.1049/iet-ipr.2011.0059.

[24] G. Swain, “A Steganographic Method Combining LSB Substitution and PVD in a Block,” Procedia Comput. Sci., vol. 85, no. Cms, pp. 39–44, 2016, doi: 10.1016/j.procs.2016.05.174.

[25] H. W. Tseng and H. S. Leng, “High-payload block-based data hiding scheme using hybrid edge detector with minimal distortion,” IET Image Process., vol. 8, no. 11, pp. 647–654, 2014, doi: 10.1049/iet-ipr.2013.0584.

[26] A. A. Zakaria, M. Hussain, A. W. A. Wahab, M. Y. I. Idris, N. A. Abdullah, and K. H. Jung, “High-capacity image steganography with minimum modified bits based on data mapping and LSB substitution,” Appl. Sci., vol. 10, no. 11, 2018, doi: 10.3390/app8112199.

[27] K. H. Jung and K. Y. Yoo, “Steganographic method based on interpolation and LSB substitution of digital images,” Multimed. Tools Appl., vol. 74, no. 6, pp. 2143–2155, 2015, doi: 10.1007/s11042-013-1832-y.

[28] N. Akhtar, “An LSB substitution with bit inversion Steganography method,” Smart Innov. Syst. Technol., vol. 43, pp. 515–521, 2016, doi: 10.1007/978-81-322-2538-6_53.
[29] M. H. Mohamed and L. M. Mohamed, “High capacity image steganography technique based on LSB substitution method,” Appl. Math. Inf. Sci., vol. 10, no. 1, pp. 259–266, 2016, doi: 10.18576/amis/100126.

[30] J. Bai, C. C. Chang, T. S. Nguyen, C. Zhu, and Y. Liu, “A high payload steganographic algorithm based on edge detection,” Displays, vol. 46, pp. 42–51, 2017, doi: 10.1016/j.displa.2016.12.004.

[31] P. D. Shah and R. S. Bichkar, “Secret data modification based image steganography technique using genetic algorithm having a flexible chromosome structure,” Eng. Sci. Technol. an Int. J., vol. 24, no. 3, pp. 782–794, 2021, doi: 10.1016/j.estersch.2020.11.008.

[32] S. Doğan, “A new data hiding method based on chaos embedded genetic algorithm for color image,” Artif. Intell. Rev., vol. 46, no. 1, pp. 129–143, 2016, doi: 10.1007/s10462-016-9459-9.

[33] R. Biswas and S. K. Bandyapadhy, “Random selection based GA optimization in 2D-DCT domain color image steganography,” Multimed. Tools Appl., vol. 79, no. 11–12, pp. 7101–7120, 2020, doi 10.1007/s11042-019-08497-x.

[34] http://sipi.usc.edu/database/database.php?volume= aerials&amp;image=5#top

[35] S. Kamil, M. A. Authors, S. N. H. S. Abdullah, and Z. Ahmad, “Lightweight and optimized multi-layer data hiding using video steganography paper,” Int. J. Adv. Comput. Sci. Appl., vol. 9, no. 12, pp. 256–262, 2018, doi: 10.14569/IJACSA.2018.091237.

[36] M.K. Hasan, S. Islam, R. Sulaiman, S. Khan, A.H.A. Hashim, S. Habib, M. Islam, S. Alyahya, M.M. Ahmed, S. Kamil, M.A. Hassan, Lightweight Encryption Technique to Enhance Medical Image Security on Internet of Medical Things Applications, IEEE Access. 9 (2021) 47731–47742. https://doi.org/10.1109/ACCESS.2021.3061710.

[37] W. Su, J. Ni, X. Hu, J. Fridrich, Image Steganography with Symmetric Embedding Using Gaussian Markov Random Field Model, IEEE Trans. Circuits Syst. Video Technol. 31 (2021) 1001–1015.

https://doi.org/10.1109/TCSVT.2020.3001122.

[38] R.B. Sushma, M. V. Namitha, G.R. Manjula, S. Johar, G.S. Hiriyanna, DNA based Steganography Using 2-3-3 Technique, 2019 Int. Conf. Data Sci. Commun. IIconDSC 2019. (2019) 19–24. https://doi.org/10.1109/IIconDSC.2019.8816945.

[39] Hasan MK, Islam S, Sulaiman R, Khan S, Hashim AH, Habib S, Islam M, Alyahya S, Ahmed MM, Kamil S, Hassan MA. Lightweight encryption technique to enhance medical image security on internet of medicalthings applications. IEEE Access. 2021 Feb 24;9:47731-42.

[40] Hasan MK, Ahmed MM, Musa SS, Islam S, Abdullah SN, Hossain E, Nafi NS, Vo N. An improved dynamic thermal current rating model for PMU-based wide area measurement framework for reliability analysis utilizing sensor cloud system. IEEE Access. 2021 Jan 18;9:14446-58.

[41] Hasan MK, Shafiq M, Islam S, Pandey B, Baker El-Ebiary YA, Nafi NS, Ciro Rodriguez R, Vargas DE. Lightweight Cryptographic Algorithms for Guessing Attack Protection in Complex Internet of Things Applications. Complexity. 2021 Apr 5;2021.

SAMI KAMIL received the B.Sc. in computer science from Baghdad College of Economic Sciences University, Baghdad, Iraq, in 2004. Then, she received an M.Sc. degree in computer science from University Technology Malaysia, Johor Bahru, Malaysia, in 2014, and a Ph.D. degree in cyber security from the Universiti Kebangsaan Malaysia. Her major research interests include information hiding, cryptography, and watermarking.

SATI NORUL HUDA SHEIKH ABDULLAH (M’11) received her first Degree in Computing at the University of Manchester Institute of Science and Technology, United Kingdom. She furthered her master’s study in Artificial Intelligence in Universiti Kebangsaan Malaysia. Later, she continued her Ph.D. Study in the area of Computer Vision at Faculty of Electrical Engineering, Universiti Teknologi Malaysia. Starting the career, she involved in conducting national and international activities such as Royal Police Malaysia, CyberSecurity Malaysia, Cyber Security Academy Malaysia, Federation of International RobotSoccer Association (FIRA), Asian Foundation, Global Ace Professional Certification Scheme, MIAMI, MACE and IDB Alumni. After being appointed as the Chairperson of the Center for Cyber Security, she holds a post as Deputy Dean (Research and Innovation). Her research focuses are Digital Forensics, Pattern Recognition, and Computer Vision Surveillance System. She has published two books entitled “Pengecaman Pola” or “Pattern Recognition” and “Computational Intelligence for Data Science Application” and more than 50 and 100 journal and conferences manuscripts correspondingly.

MOHAMMAD KAMRUL HASAN (MIEEE’13, SMIEEE’17) is currently working as a Assistant Professor in the Network and Communication Technology research cluster, Center for Cyber Security, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia (UKM). He completed a Doctor of Philosophy (Ph.D.) degree in Electrical and Communication Engineering from the faculty of Engineering, International Islamic University, Malaysia, in 2016. Dr. Hasan completed his M.Sc. degree in Communication Engineering from the Department of
Dr. Kamrul has been a Senior Member of the Institute of Electrical and Electronics Engineers since 2013, a Member of the Institution of Engineering and Technology, and a member of the Internet Society. He has served the IEEE IIUM student branch as Chairperson from 2014 to 2016. Before starting the doctoral study, Dr. Kamrul served the related industry for ten (10) years as a lead engineer, deputy manager, manager for the supervision, installation, setup, and fault handling of Radio and Microwave Communication systems and networks, Wide Area Network design/planning and implementation, PLC, Wide Area Measurement System, Microcontrollers, and Merging Units. He is a certified professional technologist and has been consulting the related industries. His research interest includes Telecommunications, Cyber-Physical and Network Security, the Internet of Things, Vehicular Networks, and smart grid technologies. Dr. Kamrul has been secured research Grants in 5G wireless network, Industrial communication network, and cyber securities. He has published 160 peer-reviewed high-impact SCIE/SCI/ISI articles and Scopus indexed Journals. Dr. Kamrul received 586 citations in Google Scholar, and his H-index is 14, and i10-Index is 26; his ResearchGate RG Score is 31.78. Dr. Kamrul was also awarded for the best publications in the exhibition. He is the reviewer in IEEE transactions on Industrial Informatics, IEEE transactions on Sensor, IEEE Access, ACM transactions on sensor networks, Expert Systems, Electrical, and Computer Engineering-Elsevier, Digital Communications and Networking-ScienceDirect, Measurement-Elsevier, IEEE transactions on Networks, Electronics-MDI, PlosOne, Telecommunication Systems-Springer, International Journal of Distributed Sensor Networks-SAGE, Wireless Communication and Mobile Computing-Hindawi, CMC-Tech Science, Sustainable Cities, and Society, Elsevier and many other reputed journals. Currently, he has been engaged with the book and journal editorial committee.

FARAH AQILAH BOHANI received her B.Sc. in Information Technology with Honours (Intelligent System) at the Universiti Kebangsaan Malaysia (UKM). Later, she continued her Ph.D. study in the area of Computer Science at Faculty of Information Science and Technology, UKM. Currently, she is a Post-Doctoral Researcher in College of Computing and Informatics, Universiti Tenaga Nasional (UNITEN). Her research interests in artificial intelligence area including, pattern recognition, optimization, neural network, classification, and image processing.