An Improved Steganography Method Based on Least-Significant-Bit Substitution and Pixel-Value Differencing

Hsing-Han Liu[1*], Pin-Chang Su[1] and Meng-Hua Hsu[1]
[1] Department of Information Management, National Defense University, Taiwan
No.70, Sec. 2, Zhongyang N. Rd., Beitou Dist., Taipei City 112, Taiwan (R.O.C.)
[e-mail: liu.hansh@gmail.com, spc.cg@msa.hinet.net, b790211@gmail.com]
*Corresponding author: Hsing-Han Liu

Received July 2, 2020;accepted November 5, 2020; published November 30, 2020

Abstract

This research was based on the study conducted by Khodaei et al. (2012), namely, the least-significant-bit (LSB) substitution combined with the pixel-value differencing (PVD) steganography, and presented an improved irreversible image steganography method. Such a method was developed through integrating the improved LSB substitution with the modulus function-based PVD steganography to increase steganographic capacity of the original technique while maintaining the quality of images. It partitions the cover image into non-overlapped blocks, each of which consists of 3 consecutive pixels. The 2nd pixel represents the base, in which secret data are embedded by using the 3-bit LSB substitution. Each of the other 2 pixels is paired with the base respectively for embedding secret data by using an improved modulus PVD method. The experiment results showed that the method can greatly increase steganographic capacity in comparison with other PVD-based techniques (by a maximum amount of 135%), on the premise that the quality of images is maintained. Last but not least, 2 security analyses, the pixel difference histogram (PDH) and the content-selective residual (CSR) steganalysis were performed. The results indicated that the method is capable of preventing the detection of the 2 common techniques.

Keywords: Steganography, irreversible, least-significant-bit, pixel-value differencing, modulus function

1. Introduction

In recent years, with the increasing popularity of the Internet, the development of diverse communications and media, and the improvement of information technologies, the cost of information exchange has dramatically shrunk, thus enabling a higher volume of frequent interpersonal communication. However, if information is not protected during transmission, it can be taken advantage of by criminals, which potentially imposes tremendous danger. Therefore, the increasingly rapid distribution of information and the digitalization of society not only result in a higher level of convenience, but also imply an increased demand for verification and security technologies, thus emphasizing the role of information security [1].

For example, let us consider the information exchange between the national military and governmental agencies. If a piece of confidential information was blatantly posted online, it could be intercepted by criminals, who could then decipher the content and even cause damage. In this regard, when transmitting information between two entities, an increasingly mature technology called information hiding should be considered to better protect information, in addition to the adoption of cryptography as the first layer of protection.

Cryptography is a relatively ancient technology. It originated from the various complex demands of human beings and has been applied for different purposes, which range from protecting personal interests to ensuring the safety of religions, races, and nations. Meanwhile, steganography is a technology related to hiding confidential information in commonly used communication channels that is covert and conceals the existence of the encrypted information. Over time, as human beings have entered the era of digitalized information and the Internet, cryptographic technologies have undergone a remarkable transformation, ranging from utilities such as embedding secrets in substances to employment in various digital media. Particularly, digital information has become the most common medium. In this sense, safety issues related to information transmission have received increasing attention from researchers. The typology of information hiding technologies is shown in **Fig. 1** [2] and includes steganography, as mentioned above.

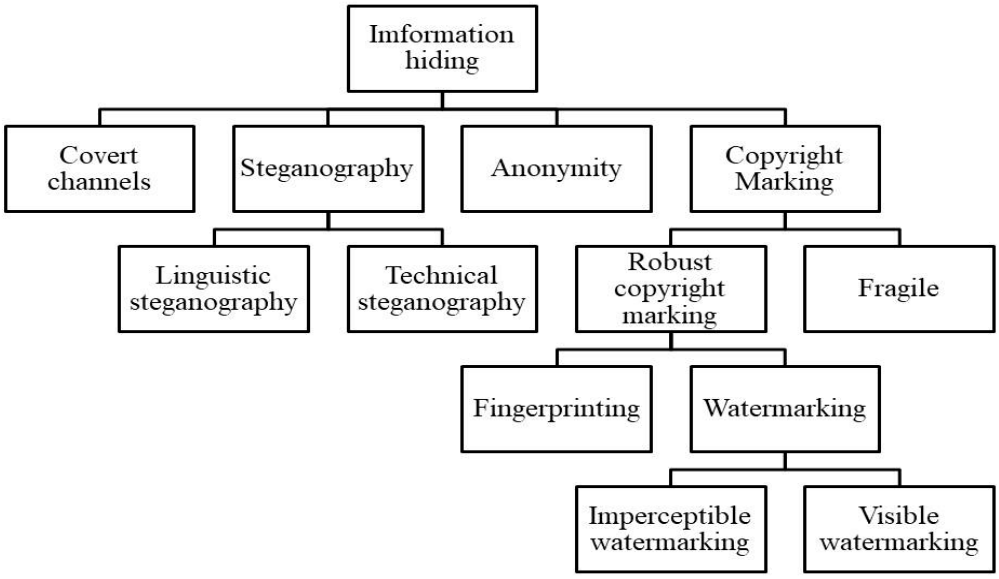


Fig. 1. Classification of information hiding technologies

Information hiding technologies can be divided into four categories:

- Covert Channels: exchanging information in channels that are unlikely to be known by others.
- Steganography: hiding information using common digital media as a carrier.
- Anonymity: exchanging materials in an anonymous manner.
- Copyright Marking: adding special marks to digital media to protect their integrity or copyright.

Today, information hiding technologies are booming and have been widely applied in the military, education, business, and other areas. Among these technologies, the most common are steganography techniques, or more specifically, technologies that hide information within an image during the transmission of digital images. As it is difficult for the human eye to distinguish a small-scale change in the pixels comprising digital images, it is feasible to store secret data in this way that go otherwise unnoticed. Even if the image is stolen by criminals, it would be considered a normal exchange, and no suspicions would be raised regarding the existence of confidential information within. Therefore, these techniques allow one to transmit confidential messages with a high degree of security, without interruption or risk of damage. The receiver can simply extract the information using a specific method. Such techniques are particularly useful in national defense, which requires significantly more safety and protection, whether the information concerns the location of the army, the size of its forces, or the time and location of planned attacks. Currently, several multimedia carriers permit such exchanges of confidential information, including that which is embedded into digital images, sounds, and films [1]; among these, digital images play a major role, primarily due to their common use today, and because the development and popularity of social media allow information to be distributed more rapidly and communication occurs via digital media more frequently. Due to their convenience of access, digital images are often employed on social media to present events ranging from gourmet or cultural activities to social gatherings or travelling, and have therefore become an essential cover for hidden information. Therefore, the process of information hiding that this research will discuss emphasizes their application in the embedding of secret information. The fundamental steps for embedding and extracting secret data are shown in **Fig. 2**.

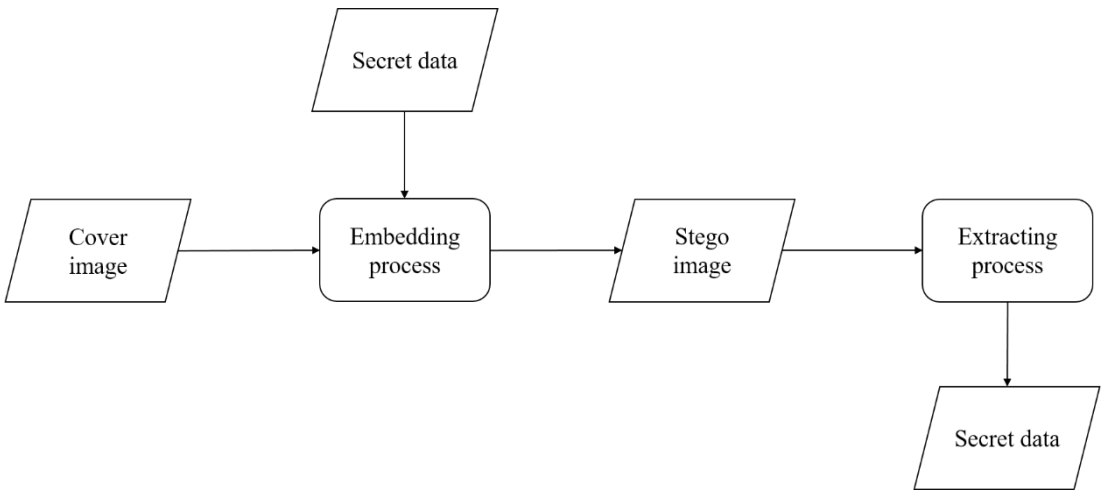


Fig. 2. Steganography and recovery of secret data

Scholars researching information hiding have primarily focused on increasing the capacity or maintaining the quality of digital images. Techniques such as Least Significant Bit (LSB) substitution [3] and Pixel-Value Differencing (PVD) [4] store secret data using certain programming algorithms and based on the difference between the neighboring pixels. Recently, there has been a surge in research on methods that combine LSB substitution and PVD, which was first proposed by Wu et al. [5] in 2005. In 2010, Yang et al. [6] suggested the use of such a combination to improve the embedding capacity and image quality of previous techniques. The method introduced by Khodaei and Faez [7] in 2012, in which data were embedded by partitioning images into 1×3 pixel blocks, also proves useful in augmenting the capacity. Taken together, our research aims to integrate the advantages of the abovementioned method, not only to achieve a higher capacity but also to effectively evade detection by steganalysis. To do so, the paper combine LSB substitution method with the improved Modulus PVD, while maintaining the quality of the images.

Based on the above research background and motivations, our research seeks to merge two common information hiding techniques (that is, LSB substitution and PVD), and to improve on the combined method to develop a technique with a satisfying performance overall. In the meantime, we use Python programming for our implementation and experiments, to evaluate the accuracy of the embedding and extracting processes in our technique. More specifically, we hope to develop an improved PVD technique, such that when combined with LSB, our new method substantially increases the capacity while also preserving the quality of a given cover image.

2. Literature Review

This section discusses the LSB substitution and PVD techniques and analyzes existing methods to provide foundational knowledge regarding our design.

2.1 Least Significant Bit Substitution Technique

Bender et al. [8] first proposed the LSB substitution technique in 1996, from which scholars later developed other information hiding methods. Among them, the method suggested by Chan and Cheng [3] in 2004 entails embedding secret data into an LSB that minimally influences pixel values (see its position in **Fig. 3**). In the case of grayscale images, the size of each luminance component byte is 8 bits, and the pixel value represents the brightness of the color, with possible values ranging from 0 to 255 (or from 0 to 2^8-1). When the luminance component is 0, the color is black, and when the value is 255, it is white. Further, the effect of a change in each bit on the pixels' luminance components differs. When the Most Significant Bit (MSB) changes from 0 to 1, the luminance component value increases by 128. However, for LSB to change from 0 to 1, the value must only increase by 1. Overall, LSB substitution is characterized by its large embedding capacity and low distortion.

Fig. 3. Binary representation of pixel values

In 2011, Medeni and Souidi [9] recommended the four-pixel differencing LSB technique, which entails dividing the cover image into 2×2 non-repeating pixel blocks that are classified as either low- (smooth areas) or high-level (edge areas). The secret data are embedded in the high-level blocks. This method determines the number of bits for hiding secret data based on the difference values among the four pixels. Moreover, in 2016, Xu et al. [10] advocated using the Modulo Three strategy to modify the LSB substitution. Their new method relies on ternary secret data, which are briefer than binary information, more efficient, and allow a larger capacity.

2.2 Pixel-Value Differencing Technique

The PVD technique was first proposed by Wu and Tsai [4] in 2003. Essentially, the method entails determining the number of bits for embedding secret data by examining the difference values of the neighboring pixels based on the characteristics of the human visual system (that is, humans' greater sensitivity to small changes in the smooth areas of an image, and higher tolerance for large changes in areas along the edge), and then hiding the secret information by modifying the pixel values of two neighboring pixels. The operational procedure first requires the definition of the interval that corresponds to the neighboring pixel values to calculate the bit-length n of the information that can be embedded. Next, n bits are selected from the embedded data and transformed into a decimal value b, which is then used to calculate a new difference value d'. Finally, the pixel value (after embedding) is obtained by calculating the difference between the original value d and the new value d'.

Wang et al. [11] devised the Modulus PVD using the modulus function in 2008. This method reduces the distortion of stego images and effectively improves image quality owing to the bidirectional adjustment feature of the modulus function. Specifically, the cover image is first divided into 1×2 non-overlapping pixel blocks and the pixel difference value d' is calculated. Subsequently, one must decide where the pixels belong based on the difference value d', and can then obtain the number of bits of embedded information n. Finally, the 2^n modulus operation is performed on the sum of the block pixels to yield the remainder, which is the embedded secret information.

In 2012, Khodaei and Faez [7] recommended a new method, which combines LSB substitution and PVD techniques (otherwise known as the new LSB/PVD technique). Specifically, one must divide the original cover image into three consecutive non-overlapping blocks and select the second pixel in each block as a base pixel. Next, the secret data are embedded into the base pixels through LSB substitution and the Optimal Pixel Adjustment Process (OPAP). The difference values between the base pixels and their left and right counterparts will then indicate the quantity of information that can be embedded. In their experiment, they first categorized all the difference values into lower- and higher-level intervals and determined the capacity based on the intervals within which the pixel difference values fall. Their results demonstrated that this method can store a large volume of secret data while maintaining high image quality.

To overcome the vulnerability of the original PVD technique from to techniques such as Regular-Singular Analysis (RS) [12] and Histogram Analysis, Shen and Huang [13] proposed a new method integrating Pixel-Value Shifting (a technique based on PVD) and Improved Exploiting Modification Direction (IEMD), or simply stated, the IEMD/PVD steganography technique. According to their research, the cover image is first brought into a Hilbert 1D sequence and then partitioned into two consecutive and non-overlapping blocks. The secret data are then embedded, leveraging the PVD technique, which considers the human visual system. The last step comprises solving the issue of pixel overflow through

optimization.

Furthermore, Liu et al. [14] proposed a new data-hiding scheme based on pixel-value differencing (PVD) in which 3-by-3 blocks are used to hide data within nine-pixel groups. The PVD scheme and the side match method are combined to ultimately produce eight groups of pixel-value differences, enabling maximum hiding capacity while maintaining an acceptable peak signal-to-noise ratio (PSNR). Experimental results demonstrate that the hiding capacity of this scheme can reach a maximum of 808,760 bits with a PSNR value of 32.0283 dB, which is difficult to detect with human vision. The results of a performance comparison with those of PVD hiding schemes proposed by other researchers confirm that the proposed scheme has a higher capacity than the other methods while maintaining an acceptable PSNR, demonstrating the superiority of the proposed hiding scheme. Finally, to assess the suitability of the proposed method to databases with different patterns, a further 2,260, 9,074, and 10,000 512 × 512-pixel greyscale images were respectively selected from the NRCS, BOSS, and BOWS2 image databases as raw images. The proposed hiding scheme was used to hide data and generate steganographic images in these raw images. Experimental results show that the minimum PSNR mean value is 35.33 dB, while the minimum mean value of the hiding capacity is 720,572 bits. These results confirm the suitability of the proposed hiding scheme for image database patterns.

3. Research Methodology and Structure

In this section, we first explain our system architecture design and then introduce the embedding and extraction processes.

3.1 Research architecture design

Building on the new LSB/PVD steganography technique [7] and in contrast to the original PVD, which incorporates this technique, our improved design takes advantage of the MPVD steganographic method [11] and improves on its algorithm. We substantially increase the embedding capacity by adjusting the number of intervals comprising the pixel difference values and the number of bits required to embed secret data, while maintaining the high quality of the cover image. We avoided simply adopting the method of LSB substitution [3] or PVD steganography [4] alone because the embedded image would then carry an evident feature that could be effectively detected by many steganalysis methods today. In short, we aim to ensure that our method can defend against such detection.

3.2 Embedding process

Our proposed embedding steps are summarized as follows.

Step 1: Partition the cover image C into 1×3 neighboring and non-overlapping pixel blocks.

Step 2: Designate the central pixel P_{ic} as the base pixel (see **Fig. 4**).

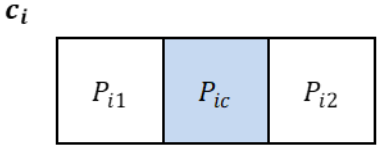


Fig. 4. Selection of base pixel

Step 3: Let the LSB substitution be 3 bits ($k = 3$) and substitute the lowest 3 bits of the binary base pixel with the 3-bit binary secret data to obtain a base pixel value P'_{ic} after embedding is performed. Further, transform the lowest 3 bits of the original base pixel P_{ic} into a decimal value LSB_i and also transform the secret data just embedded in the base pixel into a decimal value s_{ic}.

Step 4: Calculate the difference value d_{ic} between LSB_i and s_{ic} as follows:

$$d_{ic} = LSB_i - s_{ic} \tag{1}$$

Step 5: Use OPAP to adjust base pixel P'_{ic} by:

$$P'_{ic} = \begin{cases} P'_{ic} + 2^k \text{ , if } d_{ic} > 2^{k-1} \text{ and } 0 \le P'_{ic} + 2^k \le 255 \\ P'_{ic} - 2^k \text{, if } d_{ic} < -2^{k-1} \text{ and } 0 \le P'_{ic} - 2^k \le 255 \\ P'_{ic} \text{ , otherwise} \end{cases} \tag{2}$$

Step 6: Compute the difference values d_{i1} and d_{i2} between the pixels P_{i1} and P_{i2} and the base pixel value after embedding P'_{ic} (See Formulas 3, 4, and 5). Determine the interval R_j of the difference values (see **Fig. 6**) and obtain the number of bits n needed to hide the secret information.

$$d_{i1} = |P_{i1} - P'_{ic}| \tag{3}$$
$$d_{i2} = |P_{i2} - P'_{ic}| \tag{4}$$

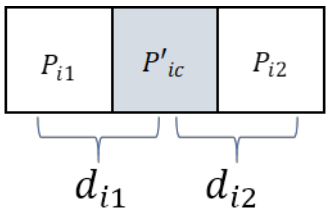


Fig. 5. Pixel difference values

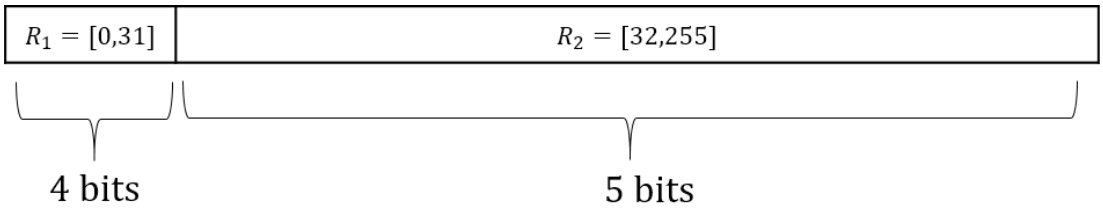


Fig. 6. Interval setting for pixel difference values

Step 7: Calculate the sums of P'_{ic} with P_{i1} and P_{i2}, respectively, and compute the remainders F_{remL} and F_{remR} using the mod 2^n operation as follows:

$$F_{remL} = (P_{i1} + P'_{ic}) \bmod 2^n \tag{5}$$
$$F_{remR} = (P_{i2} + P'_{ic}) \bmod 2^n \tag{6}$$

Step 8: Denote the decimal values of the secret information embedded into pixels P_{i1} and P_{i2} as $b_{i\mathrm{L}}$ and b_{iR}. Compute the m_{ia} and m_{ib} values according to Formulas 7-10, denoted as m_{iaL}, m_{iaR}, m_{ibL}, and m_{ibR}:

$$m_{iaL} = |F_{remL} - b_{i\mathrm{L}}| \tag{7}$$
$$m_{iaR} = |F_{remR} - b_{iR}| \tag{8}$$
$$m_{ibL} = 2^n - |F_{remL} - b_{i\mathrm{L}}| \tag{9}$$
$$m_{ibR} = 2^n - |F_{remR} - b_{iR}| \tag{10}$$

Step 9: According to the values of F_{remL}, F_{remR}, m_{iaL}, m_{iaR}, m_{ibL}, and m_{ibR}, calculate the pixel values in the four cases, as shown in Formulas 11-14 (F_{rem} is F_{remL}or F_{remR}, m_{ia} is m_{iaL}or m_{iaR}, m_{ib} is m_{ibL}or m_{ibR}, b_i is $b_{i\mathrm{L}}$or b_{iR}, P_i is P_{i1} or P_{i2}, and P'_i is P'_{i1} or P'_{i2}):

Case 1：$F_{rem} > b_i$ and $m_{ia} \le (2^n/2)$

$$P'_i = P_i - m_{ia} \tag{11}$$

Case 2：$F_{rem} > b_i$ and $m_{ia} > (2^n/2)$

$$P'_i = P_i + m_{ib} \tag{12}$$

Case 3：$F_{rem} \le b_i$ and $m_{ia} \le (2^n/2)$

$$P'_i = P_i + m_{ia} \tag{13}$$

Case 4：$F_{rem} \le b_i$ and $m_{ia} > (2^n/2)$

$$P'_i = P_i - m_{ib} \tag{14}$$

Step 10: If pixel overflow occurs after embedding, adjust the values as follows:

$$P'_i = \begin{cases} P'_i + 2^n, \text{if } P_i \ge 0 \text{ and } P'_{ic} \ge 0 \text{ and } P'_i < 0 \\ P'_i - 2^n, \text{if } P_i \le 255 \text{ and } P'_{ic} \le 255 \text{ and } P'_i > 255 \end{cases} \tag{15}$$

Step 11: After hiding the secret data as instructed in the formulas above, confirm whether the intervals of the pixel difference values are the same before and after embedding (See **Fig. 7**). If the intervals are unequal, adjust the values using Formulas 16 and 17 to avoid errors in data extraction. For example, if the pixel difference value $d_i \in [0, 31]$ while the difference value after embedding is $d'_i > 31$, use Formula 16; if the pixel difference value $d_i \in [32, 255]$ while the difference value after embedding is $d'_i < 32$, use Formula 17.

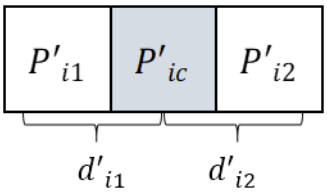


Fig. 7. Pixel difference values after embedding

$$P'_i = \begin{cases} P'_i - 2^n, & \text{if } P'_i > P'_{ic} \\ P'_i + 2^n, & \text{otherwise} \end{cases} \tag{16}$$

$$P'_i = \begin{cases} P'_i + 2^n, & \text{if } P'_i > P'_{ic} \\ P'_i - 2^n, & \text{otherwise} \end{cases} \tag{17}$$

Step 12: Check for any overflow again. In the case of an overflow, adjust the values using Formula 18 to avoid extraction errors.

$$P''_i = \begin{cases} P'_i + 2 \times 2^n, \text{if } P'_i < 0 \\ P'_i - 2 \times 2^n, \text{if } P'_i > 255 \end{cases} \tag{18}$$

3.3 Extraction process

The extraction process is less complex than that of embedding, and is explained as follows.

Step 1: Partition the embedded image into three consecutive pixel blocks.
Step 2: Designate the central pixel P'_{ic} as the base pixel and extract the secret data from it.
Step 3: Calculate the pixel difference values between the base pixel and the two neighboring pixels, respectively, and confirm whether the difference values fall into the intervals. Extract the decimal secret information b_i using Formula 19 and transform it into a binary value.

$$b_i = (P'_i + P'_{ic}) \bmod 2^n \tag{19}$$

3.4 Description of example for embedding and extraction

In this section, we explain the entire embedding and extraction process of our steganography technique using a default 1×3 pixel block.

3.4.1 Description of embedding process

As shown in **Fig. 8**, a cover image is divided into three consecutive pixel blocks. Suppose the pixel value of a block $(P_{i1}, P_{ic}, P_{i2}) = (151,122,75)$, and s comprises the secret data to be hidden.

First, we embed the information into the base pixel P_{ic}. Suppose the secret data $s = (001001010001)_2$ and the base pixel $P_{ic} = 122 = (1111010)_2$. If we define $k = 3$, then $LSB_i = (010)_2 = 2$. We then read the first three bits of the secret information $s_{ic} = (001)_2$ and substitute the lowest 3 bits of P_{ic} using LSB substitution to obtain the base pixel $P'_{ic} = (1111001)_2 = 121$. Next, according to Formula 1, we calculate the difference value d_{ic} between LSB_i and s_{ic}, which results in $d_{ic} = LSB_i - s_{ic} = 2 - 1 = 1$. According to OPAP, we adjust P'_{ic} with Formula 2, such that the base pixel after embedding $P'_{ic} = P'_{ic} = 121$. We later compute the difference values between the embedded base pixel P'_{ic} and the remaining two pixels P_{i1} and P_{i2}, respectively, and obtain $d_{i1} = |151 - 121| = 30$ and $d_{i2} = |75 - 121| = 46$. The comparison suggests that d_{i1} belongs to R_1 and d_{i2} belongs to R_2. Therefore, the numbers of bits for hiding secret data are $n_{iL} = 4$ and $n_{iR} = 5$, respectively. Further, based on the number of bits, we transform the binary secret information into decimal values $b_{i1} = (0010)_2 = 2$ and $b_{i2} = (10001)_2 = 17$. We then calculate F_{remL}, F_{remR}, m_{iaL}, m_{iaR} and m_{ibL} following Formulas 5-10.

For the next step, we embed pixel P_{i1}. As $F_{remL} = (151 + 121) \bmod 2^4 = 0$, $m_{iaL} = |0 - 2| = 2$ and $m_{ibL} = 16 - 2 = 14$, the scenario is in line with Case 3 in Formula 13. As such, we have the embedded pixel $P'_{i1} = P_{i1} + m_{iaL} = 151 + 2 = 153$ (no overflow). We then check the difference value d'_{i1} between P_{i1} and the embedded base pixel P'_{ic} to

determine whether the difference value falls into the same interval as the original difference value d_{i1}. As $d'_{i1} = |153 - 121| = 32$, $d'_{i1} \in R_2$, which is different from the interval of the original difference value ($d_{i1} \in R_1$). Thus, we adjust the value using Formula 18 to obtain $P'_{i1} = 153 - 16 = 137$ (no overflow), and $d'_{i1} = |137 - 121| = 16$.

Finally, we examine the embedding of P_{i2}. As $F_{remR} = (75 + 121) \bmod 2^5 = 4$, $m_{iaR} = |4 - 17| = 13$ and $m_{ibR} = 32 - 13 = 19$, the scenario is in line with Case 3 in Formula 13. As such, we have the embedded pixel $P'_{i2} = P_{i2} + m_{iaR} = 75 + 13 = 88$ (no overflow). Furthermore, as $d'_{i2} = |88 - 121| = 33$, $d'_{i2} \in R_2$, which is the same as the interval of the original difference value, there is no need to adjust the pixel. Hence, the final embedded pixel blocks $(P'_{i1}, P'_{ic}, P'_{i2}) = \{137,121,88\}$.

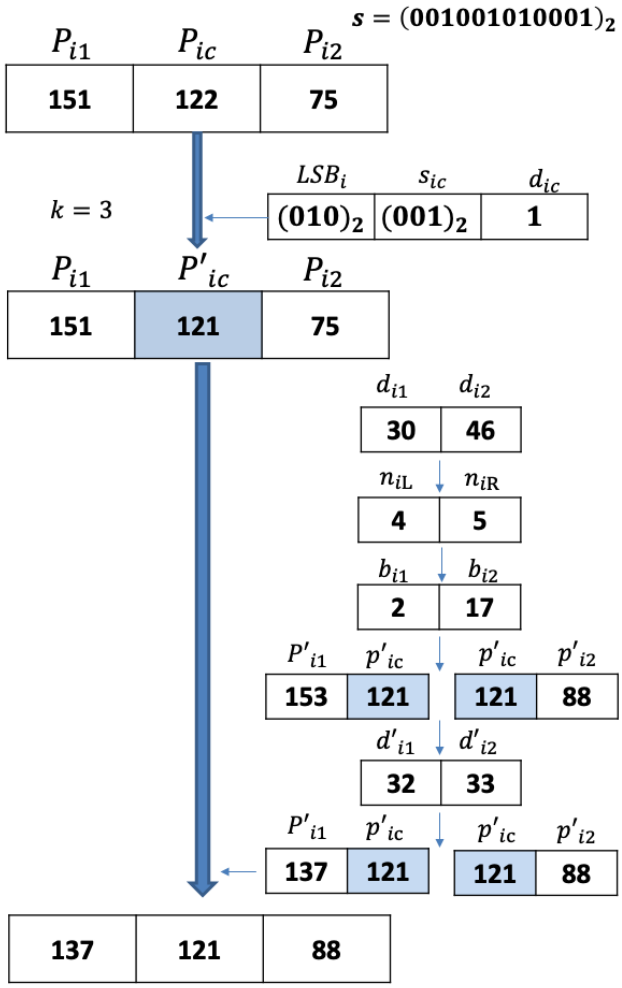


Fig. 8. Embedding process of proposed method

3.4.2 Description of extraction process

As shown in **Fig. 9**, the embedded image is partitioned into three consecutive pixel blocks. Among them, it is already known that the embedded base pixel $P'_{ic} = 121 = (1111001)_2$

and $k = 3$. We extract the secret information from the lowest 3 bits in the base pixel, which is $(001)_2$.

We then extract the secret data from the other two pixels P'_{i1} and P'_{i2}. First, we calculate the difference value $d'_{i1} = |137 - 121| = 16$ and $d'_{i2} = |88 - 121| = 33$. As $d'_{i1} \in R_1$ and $d'_{i2} \in R_2$, we infer that $n_{iL} = 4$ and $n_{iR} = 5$. As such, we are able to compute the decimal value of the secret data in P'_{i1}, which is $b_{i1} = (137 + 121) \bmod 2^4 = 2$, and then transform b_{i1} into a binary value and obtain $(0010)_2$. Likewise, we calculate the decimal value of the secret data in P'_{i2}, which is $b_{i2} = (88 + 121) \bmod 2^5 = 17$ and transform b_{i2} into a binary value $(10001)_2$. For the last step, we combine the three pieces of information in order and the complete set of secret data is therefore $s = (001001010001)_2$.

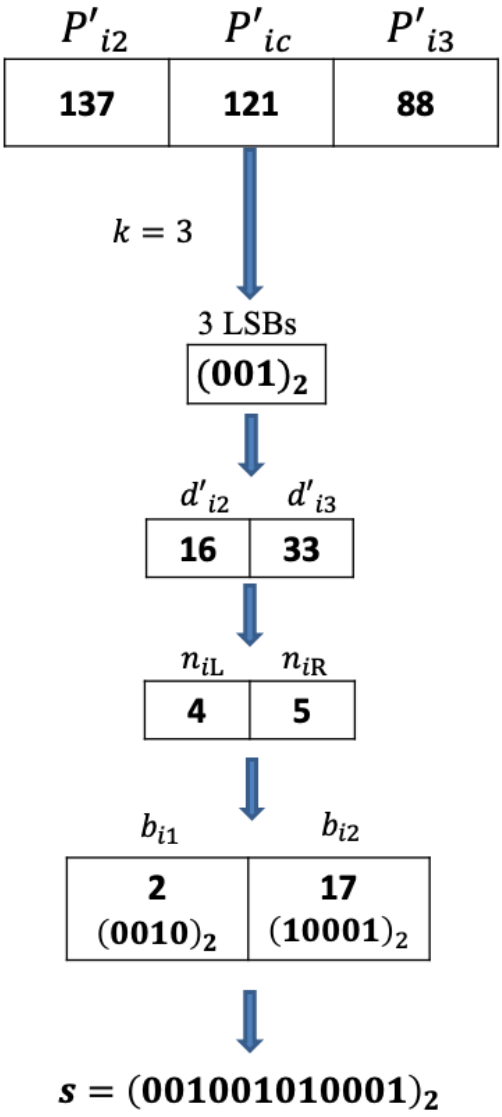


Fig. 9. Extraction process of proposed method

4. Experimental Results

4.1 Experimental environment and analysis of results

The hardware and software environments for our experiment are detailed as follows:

- Hardware environment: we employ MacBook Pro 3.1 GHz Dual-core Intel Core i5 with 8GB RAM to calculate the capacities and the image quality of the different steganography techniques. In the meantime, we use a desktop computer with Intel Core i5-6500 3.2 GHz and 16GB RAM to test for wide applicability and safety.
- Software environment: we use Jupiter Application in Anaconda, and we adopted Python programming to implement the original PVD, MPVD, the new LSB/PVD, IMED/PVD, the new Side Match/PVD, and our proposed technique. Additionally, we generate Pixel Difference Histograms (PDH) via MATLAB.

For the test cover images, we select the 8 512×512 grayscale classic images commonly used in the field of information hiding, such as that of Barbara (See **Fig. 10**). The collection features both complex textures and smooth regions, and also includes portraits, landscapes, objects, modes of transportation, and other images. Thus, it is representative of most digital images. Further, to assess image quality, we use PSNR and SSIM as indicators.

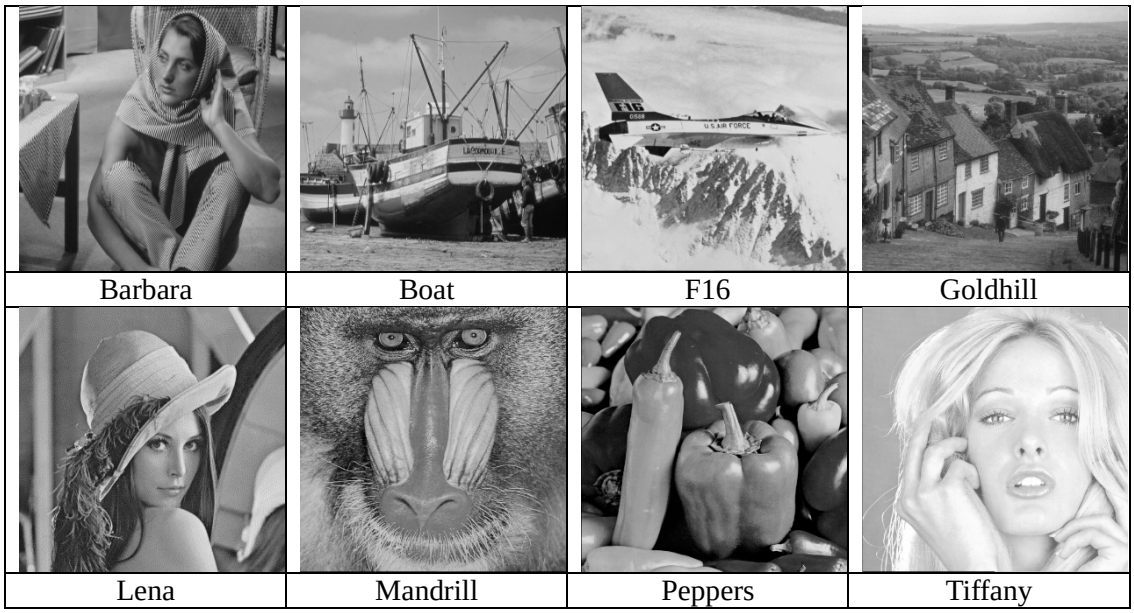


Fig. 10. Test images

We compare the experimental results of our method, the original PVD [4], MPVD [11], Type 2 of the new LSB/PVD [7], IMED/PVD [13], and the new Side Match/PVD [14]. The results are summarized in **Tables 1**, **2**, and **3** (the comparison of embedding capacities is shown in **Table 1**, that of PSNR in **Table 2**, and that of SSIM in **Table 3**).

Table 1. Comparison of embedding capacity by method (bits)

Method / Image	Original PVD (2003)	MPVD (2008)	New LSB/ PVD (2012)	IMED/ PVD (2015)	New Side Match/ PVD (2018)	Proposed method
Barbara	453,247	453,247	892,917	433,647	764,388	983,749
Boat	418,232	418,232	821,774	411,574	724,317	965,941
F16	409,834	409,834	809,262	405,136	717,511	963,471
Goldhill	406,041	411,896	813,968	408,096	720,574	962,633
Lena	409,807	409,807	809,966	406,132	712,168	962,452
Mandrill	457,105	424,107	886,516	417,772	808,760	977,179
Peppers	407,643	407,643	802,228	404,301	713,062	961,563
Tiffany	407,365	407,365	806,847	404,136	709,758	961,449
Average	421,159	417,766	830,435	411,349	733,817	**967,305**

From **Table 1**, we discover that the original PVD, MPVD, and IMED/PVD methods have relatively low embedding capacity, as they only feature one type of steganography technique. In contrast, due to the combination of two steganography techniques, the new LSB/PVD and the new Side Match/PVD can embed 3 and 8 sets of secret data into a single block, respectively, and therefore have higher embedding capacities than the abovementioned three methods (each block can only embed 1 set of secret data). However, our method can store three sets of secret information in each of the 1×3 pixel blocks, therefore indicating a higher embedding capacity. Moreover, the average embedding capacity of our method is 967,305 bits, which constitutes an increase of 16.5% compared to that of the new type of LSB/PVD with high capacity (Type 2). Therefore, proposed method has achieved satisfactory performance in terms of its embedding capacity.

As demonstrated by the quality indicators in **Table 2** and **Table 3**, the original PVD, MPVD, and IEMD/PVD methods have better image quality because of their lower embedding capacity. In comparison, the image quality of our method is slightly lower, but is still indistinguishable to the human eye (PSNR > 30 dB), as ours has a relatively higher capacity. Specifically, the average PSNR of our method is around 34.795 dB with an average SSIM of 0.907. Compared to the new type of LSB/PVD with a higher capacity (Type 2), our method has increased the embedding capacity (by around 16.5%) while the image quality has only moderately decreased (PSNR is approximately 6.5% lower and SSIM is approximately 8% lower). These results confirm that our method can maintain a certain level of image quality while effectively increasing the embedding capacity.

Table 2. Comparison of PSNR by method (dB)

Image \ Method	Original PVD (2003)	MPVD (2008)	New LSB/ PVD (2012)	IMED/ PVD (2015)	New Side Match/ PVD (2018)	Proposed method
Barbara	35.814	38.239	36.127	41.01	33.56	33.762
Boat	39.741	41.740	37.293	41.88	35.87	35.006
F16	40.161	42.429	37.534	42.75	36.19	35.335
Goldhill	41.040	42.860	37.556	41.64	36.23	35.313
Lena	41.057	43.121	37.632	42.26	36.7	35.354
Mandrill	39.904	41.764	36.293	40.90	32.04	33.934
Peppers	40.483	41.814	37.976	41.66	34.83	34.891
Tiffany	40.939	42.830	37.393	36.77	35.91	34.764
Average	39.893	41.850	37.226	41.109	35.166	**34.795**

Table 3. Comparison of SSIM by method

Image \ Method	Original PVD (2003)	MPVD (2008)	New LSB/ PVD (2012)	IMED/ PVD (2015)	New Side Match/ PVD (2018)	Proposed method
Barbara	0.978	0.987	0.933	0.985	0.934	0.918
Boat	0.977	0.987	0.936	0.982	0.931	0.900
F16	0.974	0.968	0.931	0.979	0.923	0.882
Goldhill	0.982	0.991	0.949	0.986	0.944	0.920
Lena	0.979	0.988	0.936	0.981	0.929	0.896
Mandrill	0.990	0.994	0.960	0.993	0.956	0.956
Peppers	0.979	0.988	0.933	0.982	0.927	0.900
Tiffany	0.975	0.986	0.929	0.979	0.951	0.883
Average	0.979	0.986	0.938	0.983	0.937	**0.907**

Note: SSIM is between -1~1 (1 denotes that the embedded image is the same as the original image while 0 signifies the absence of structural similarity)

To assess the applicability of our method to different images, we further calculate the embedding capacities and image quality using the 10,000 512×512 8-bit grayscale images from the BOSSBase Database [15] (See examples in **Fig. 11**). As shown in the experimental results (See **Table 4**), the average embedding capacity of our method is around 962,718 bits,

with an average PSNR of around 34.906 dB and an average SSIM of around 0.878. Compared to the new type of LSB/PVD with a higher capacity (Type 2), which is promoted by Khodaei and Faez, our embedding capacity has increased by approximately 19.5% while PSNR has slightly decreased by 0.7% and SSIM has dropped marginally by 4.5%. This suggests that our proposed method has a higher embedding capacity and can still maintain an acceptable image quality compared to other existing methods, thus lending qualified support to its superiority.

Fig. 11. Examples of grayscale images from BOSSBase Database

Table 4. Calculated means of indicators using images from BOSSBase Database

Image \ Method	Original PVD 2003	MPVD 2008	New LSB/ PVD 2012	IMED/ PVD 2015	New Side Match/ PVD 2018	Proposed method
Embedding capacity	409,780	409,780	805,717	405,605	719,708	962,718
PSNR	40.858	43.209	35.146	42.724	35.502	34.906
SSIM	0.969	0.986	0.919	0.976	0.909	0.878

4.2 Security analysis

To confirm our method's ability to defend against various steganalysis techniques, we conduct two typical security analyses, including a Pixel Difference Histogram (PDH) and Content-Selective Residuals (CSR) [16].

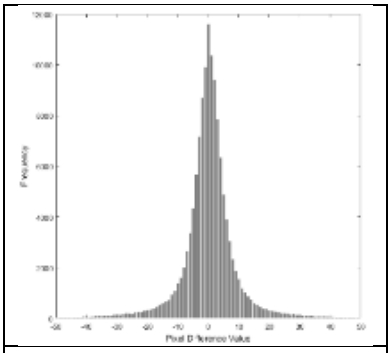
4.2.1 Pixel Difference Histogram analysis

The original PVD steganography technique poses a security risk because the technique determines the number of bits needed to embed secret data using fixed intervals. Moreover, the length of all intervals is 2^n (n is the number of bits for embedding), and the difference values of the new neighboring pixels are related to the length of the intervals. Further, the technique embeds information into the LSB, which makes it similar to LSB substitution, and it therefore generates a pixel-pair feature that is easily detectable by current steganalysis methods.

In this regard, this study performs a security analysis on the original PVD, MPVD, the new LSB/PVD, IEMD/PVD, the new Side Match/PVD, and our method, by examining the change in the features of the PDH after embedding the cover image (see the PDH of the cover image in **Fig. 12** and that of the embedded image in **Fig. 13**). The PDH of the cover image has a distribution that is nearly normal. However, as shown in **Fig. 13**, the height and width of the PDHs change after embedding and some are no longer normally distributed (see (a)(c)(e) in **Fig. 13**). Such changes in PDH features after embedding constitute an opportunity for steganalysis. In comparison, as shown in (b)(d)(f) of **Fig. 13**, the PDHs generated from MPVD, IEMD/PVD, and our method still conform to a normal distribution, and are thus relatively successful at defending against steganalysis based on the PDH features.

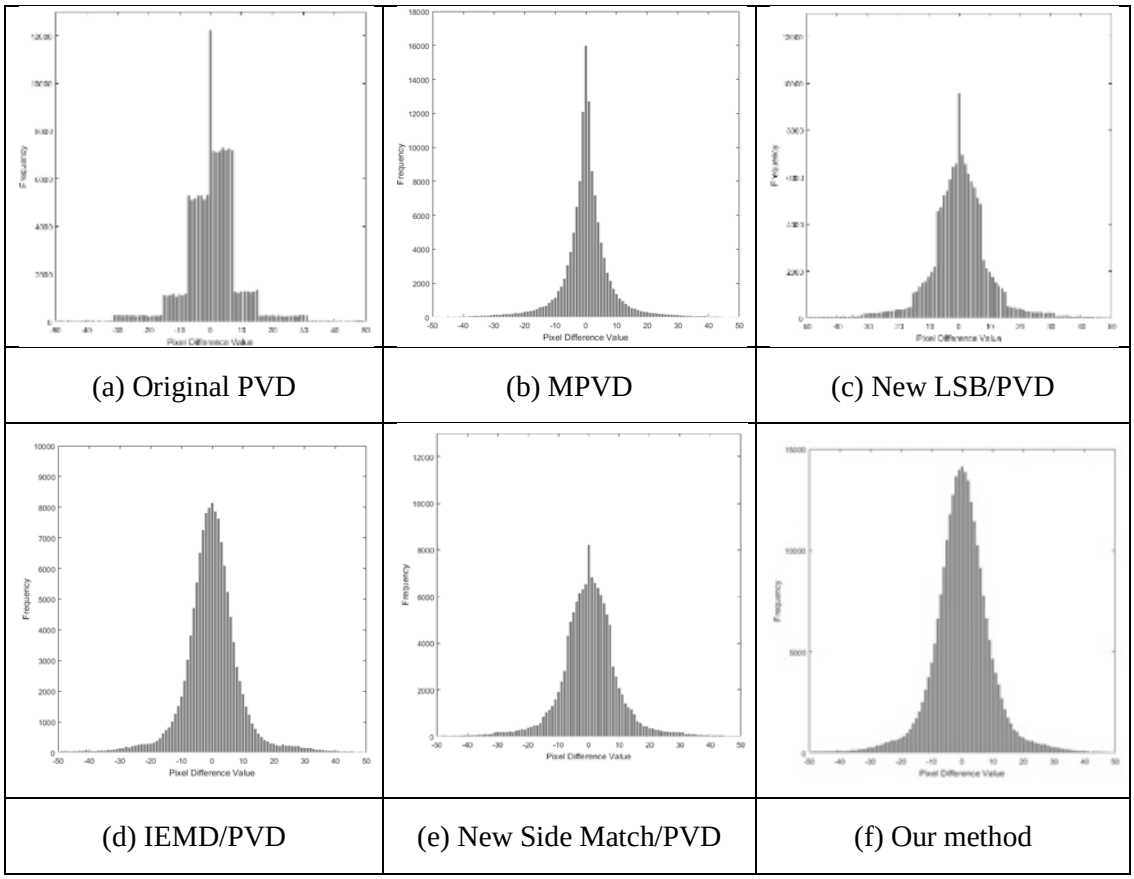
4.2.2 CSR analysis

Content-Selective Residuals or CSR analysis [16], which was proposed by Denemark et al. in 2014, can be used to detect the 1183 features that are generated using spatial domain steganography. To prove that our method is robust against CSR detection, we analyze the features after embedding using 10,000 512×512 8-bit grayscale images from the BOSSBase Database as test images. We evaluate the accuracy with AC (see Formula 20). The lower the value, the more security our method provides. Moreover, TP represents the number of embedded images that are correctly identified, TN represents that of cover images correctly identified, FP represents that of images falsely identified as embedded images, and FN represents that of images falsely identified as cover images.



Cover image

Fig. 12. PDH of pixel difference values of cover image



(a) Original PVD (b) MPVD (c) New LSB/PVD

(d) IEMD/PVD (e) New Side Match/PVD (f) Our method

Fig. 13. PDH of pixel difference values with different steganographic methods

$$AC = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \quad (20)$$

Table 5. Experimental results of CSR detection

Indicator / Method	TP	TN	FP	FN	Accuracy (%)
Our method	5000	659.2	4148.8	0	56.6%

The experimental results in **Table 5** show a detection accuracy of 56.6% for our method, which is only slightly higher than the probability of random judgments (50.5%). This implies that the CSR steganalysis failed to detect the difference between the features of cover images and those of embedded images using our method. Thus, we have solid evidence that our method is robust against CSR detection.

5. Conclusion

This study aims to improve the embedding capacity of steganography based on LSB and PVD. While developing the algorithm, the original design intended to determine the volume

of embedded information by increasing the number of intervals of difference values. However, as our method matches the base pixel with the right and left pixels, respectively, and as the base pixel cannot be further changed once the secret data are embedded into the 3 LSB, our design has altered the bidirectional adjustment of the original MPVD. As such, a circumstance may arise in which the intervals of the pixel difference values before and after embedding are not the same. Moreover, if we augment the embedding capacity of each interval, the image quality may fall below the range that is acceptable for the human eye. As such, we modified the design of the algorithm after performing a series of experiments to lower the number of intervals comprising pixel difference values to only 2. In the meantime, we let the embedding capacities of the two intervals be 4 and 5 bits, respectively, for our calculation. In so doing, we simultaneously improve the capacity and maintain image quality.

In the field of information hiding, the embedding capacity and quality of embedded images are the two most common criteria for evaluating the performance of steganography techniques. This study demonstrates that our method is effective in its ability to increase the capacity of digital images while also preserving the quality of the image at a level that is acceptable to the human eye. Compared to the embedding capacities of the original PVD, MPVD, the new LSB/PVD, IEMD/PVD, and the new Side Match/PVD, the embedding capacity resulting from proposed method has expanded by 546,146 bits (129%), 549,539 bits (132%), 136,870 bits (16.5%), 555,956 bits (135%), and 233,488 bits (31.8%). In terms of image quality, our method prompted decreases in the PSNR and SSIM values of only around 6.5% and 8%, respectively, compared to the new type of LSB/PVD with a high capacity (Type 2). Therefore, our proposed method can increase embedding capacity to a large extent while also maintaining image quality.

References

[1] N. I Wu, K. C. Fu and C. M. Wang, "A Novel Data Hiding Method for Grayscale Images Based on Pixel-Value Differencing and Modulus Function," *J. Internet Technol.*, vol. 11, no. 7, pp. 1071-1081, 2010. Article (CrossRef Link)

[2] F. A. P. Petitcolas, R. J. Anderson and Kuhn, M. G., "Information hiding-a survey," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1062-1078, Jul. 1999. Article (CrossRef Link)

[3] C. K. Chan and L. M. Cheng, "Hiding Data in Images by Simple LSB Substitution," *Pattern Recognit.*, vol. 37, no. 3, pp. 469-474, Mar. 2004. Article (CrossRef Link)

[4] D. C. Wu and W. H. Tsai, "A steganographic method for images by pixel-value differencing," *Pattern Recognit. Lett.*, vol. 24, no. 9-10, pp.1613-1626, Jun. 2003. Article (CrossRef Link)

[5] H. C. Wu, N. I. Wu, C. S. Tsai and M. S. Hwang, "Image steganographic scheme based on pixel-value differencing and LSB replacement methods," *IEE P-Vis Image Sign*, vol. 152, no. 5, pp. 611-615, Oct. 2005. Article (CrossRef Link)

[6] C. H. Yang, C. Y. Weng, S. J. Wang and H. M. Sun, "Varied PVD+LSB evading detection programs to spatial domain in data embedding systems," *J. Syst. Softw.*, vol. 83, no. 10, pp.1635-1643, Oct. 2010. Article (CrossRef Link)

[7] 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, Oct. 2012. Article (CrossRef Link)

[8] W. Bender, D.Gruhl, N.Morimoto and A. Lu, "Techniques for Data Hiding," *IBM Syst. J.*, vol. 35, no. 3-4, pp. 313-336, 1996. Article (CrossRef Link)

[9] M. B. O. Medeni and E. M. Souidi, Ouarzazate, MR, "A novel steganographic method for gray-level images with four-pixel differencing and LSB substitution," in *Proc. of 2011 International Conference on Multimedia Computing and Systems*, Jul. 2011. Article (CrossRef Link).

[10] W. L. Xu, C, C. Chang, T. S. Chen and L. M. Wang, "An improved least-significant-bit substitution method using the modulo three strategy," *Displays*, vol. 42, pp.36-42, Apr. 2016. Article (CrossRef Link)
[11] C. M. Wang, , N. I. Wu, , C.-S.Tsai and M.-S. Hwang, "A high quality steganographic method with pixel-value differencing and modulus function," *J. Syst. Softw.*, vol. 81, no. 1, pp. 150-158, Jan. 2008. Article (CrossRef Link)
[12] Association for Computing Machinery, "Reliable detection of LSB steganography in color and grayscale images," in *Proc. of ACM Multimedia 2001 Workshops-Multimedia and Security: New Challenges*, pp.27-30, Oct. 2001. Article (CrossRef Link).
[13] S. Y. Shen and L. H. Huang, "A data hiding scheme using pixel value differencing and improving exploiting modification directions," *Comput. Secur.*, vol. 48, pp. 131-141, Feb. 2015. Article (CrossRef Link)
[14] H. H. Liu, Y. C. Lin and C. M. Lee, "A Digital Data Hiding Scheme based on Pixel-Value Differencing and Side Match Method," *Multimed. Tools Appl.*, vol. 78, no. 9, pp. 12157-12181, 2019. Article (CrossRef Link)
[15] BOSS Break Oure Steganographic System, "BOSSBases (v0.93) [Online]," Available: http://agents.fel.cvut.cz/boss/index.php?mode=VIEW&tmpl=materials, Accessed on: Mar. 2, 2020.
[16] T. Denemark, J. Fridrich and V. Holub, "Further study on the security of S–UNIWARD," in *Proc. of SPIE - The International Society for Optical Engineering*, vol. 9028, Feb. 2014. Article (CrossRef Link)

Dr. Hsing-Han Liu received the B.S. and M. S. degree in information management from National Defense Management College, Taipei, Taiwan, R.O.C. and Shih Hsin University , Taipei, Taiwan, R.O.C., in 1997 and 2003, respectively, and the Ph.D. degree in department of electrical and electronic engineering from National Defense University, Taoyuan, Taiwan, R.O.C., in 2013.Currently, he is an associate professor in the department of information management, National Defense University, Taipei, Taiwan. His current research interests include information hiding, steganalysis, and information security.

Dr. Pin-Chang Su is working as a professor in the Department of Information Management at National Defense University, Taiwan. He received his Ph.D. degree in Electrical Engineering from Chang Gung University, Taiwan in 2007. He teaches courses like Information Security and Cryptography, Database Systems, Programming Languages, E-Commerce Systems, etc. His research mainly focuses on Algorithms Design in Error-Control Coding, Information Security, Cryptographic Systems and E-Commerce Technologies. His published articles can be found in most academic journals like Security and Communication Networks, Computers and Electrical Engineering, Journal of Internet Technology, Journal of E-Business, Journal of Chung Cheng Institute of Technology and so forth.

Meng-Hua Hsu received M. S. degree in department of information management from National Defense University, Taipei, Taiwan. His current research interests include information hiding and information security.