A Novel Reversible Data Hiding Based on Improved Pixel Value Ordering Method

Cao Thi Luyen[1], Pham Dinh Phong[1], Pham Hoang Hiep[2]

[1]Faculty of Information Technology, University of Transport and Communications, Hanoi, Vietnam
[2]12A3 Informatics - HUS High School For Gifted Students, VNU Hanoi - University of Science, Vietnam

Communication: received 09 February 2022, revised 03 March 2022, accepted 13 March 2022
DOI: 10.32913/mic-ict-research.v2022.n2.1064

Abstract: **Reversible data hiding based on the preservation of sorted pixel value ordering (PVO) technique has been researched and expanded recently due to its high embedding capacity and good image quality. secret data is always embedded in the largest or smallest pixel of the sub-block. Thus, each sub-block will embed two bits. The more blocks an image has, the higher the embedding potential. In this paper, to have more sub-blocks, the paper divides the image into sub-blocks with 3 pixels and embeds 2 bits in the maximum value and 1 bit in the minimum value. Therefore, each sub-block of the proposed embedding scheme is able to hide three bits instead of the two bits as in the original scheme. The experimental results also show that the proposed scheme has a significantly higher embedding capacity.**

Keywords: *Reversible data hiding, improved pixel value ordering (IPVO).*

I. INTRODUCTION

The data transmission over the internet environment brings great benefits, besides it also has many potential risks such as: unauthorized copying, content distortion, tampering. Therefore, the issue of protecting the integrity of digital data is one of the urgent issues in the field of information security. The main methods to protect digital content include encryption, hiding, etc. This article researches on hiding information, a technique to hide secret information into digital products where it is difficult for the naked eye to tell the difference between the original product and the product containing the information. The embedded confidential information is then restored as a tool to verify the integrity of the digital product or as a basis for proving the author's copyright [1]. Hiding techniques are divided into two types: the first is the traditional steganography (it is impossible to recover the original), and the second is the reversible steganography (the embedding technique ensures the integrity of the original). Traditional data hiding techniques [2, 3] often cause permanent distortion in the original image, which limits their application in some sensitive areas such as medical image processing and military communications. Therefore, new data hiding methods called reversible (or lossless) data hiding methods are introduced to overcome this limitation. The reversible data hiding (RDH) method can not only extract the embedded data but also restore the original image. This type of reversible hiding has expanded the applicability of information hiding in life, especially in sensitive fields such as healthcare, military, and education. The hiding domain can be in the transform domain or the image space domain, but embedding information in the image space domain is considered to get high embedding capacity but good image quality. The first scheme embedding on the image space domain was proposed by Tian in 2003 with the famous name of difference extension [4]. Tian splits the image into pairs pixels to embed one bit in each pair. This method achieves the embedding capacity of 50% of the image size but distortion image high. Many authors have improved [5–7] to increase embedding and improve image quality. To increase embedding, [5, 7] expands Tian by generating multiple differences and embeddings on each difference value of the pair pixels, while [6] improves the map construction method of [4] to reduce the size of the compressed map so that there is more space to store secret information.

Tian's scheme [8–10] improvement is well known as the extension of the prediction error method to improve the quality of the information image. Instead of expanding the difference between two original pixels, [8–10] embeds information based on the prediction error of two original pixels, thereby greatly reducing image distortion when embedding information. There are also integer transformation methods [11–14], histogram shift [15–17]. Recently, appeared the PVO method [18], which hides information in a reversible manner based on the prediction error so that the pixel order after sorting is preserved. First, [18] divides the original image into none overlapping sub-blocks. Next, each block will be sorted in ascending order. If the maximum error prediction (the smallest error prediction) is 1 (-1) then a bit is hidden in it so that the arrangement of a sequence of pixels remains unchanged to serve the information recovery process. Scheme [19] improves [18] by using blocks with zero error prediction to improve embedding. However, [19], if the block has many pixels with the largest value and many pixels with the second largest value, it is ignored. The scheme [20] is called the improved PVOK method [19] above. [20] has to transform K pixels to embed one more bit, so the image quality is significantly reduced, but the embedding capacity does not increase much. Schema GePVOK [21] continues to develop a scheme [20] to embed K bits in K pixels with the largest (smallest) value. As a result, embedding capabilities, as well as image quality, are improved [20]. The weakness of scheme [21] is that it uses two bits to store the location map while the map is also an information that needs to be embedded in the image. Map size is inversely proportional to embedding capacity. [22] not only has overcome the above disadvantage but also uses a block with a pixel value of {0,1,254,255}, so that the embedding capacity of this scheme is higher than [21]. [23] scheme proposed to divide the image into the size of three elements to create more blocks, each block is embedded two bits to improve embedding. In addition, [23] proposes a criterion for evaluating flat blocks, only the blocks considered planar should be embedded. As a result, the image quality is greatly better especially in case the embedded data is much less. [24] enlarged the block flatness assessment method and used GePVOK to improve embedding. This work will propose a new inverse algorithm that allows 3 bits to be hided in a 3-pixel subblock: 2 bits will be embedded in the maximum prediction error in a new way, and one bit will be hidden in the prediction error. The smallest prediction error is the same as IPVO. It is easy to get the proposed schema with higher embedding capacity than the related sub-block schemes with the size of 3 pixels. The article also prioritizes embedding in the image block with high flatness (flatness is evaluated as [24]) to get better the quality of images containing information. The next content will introduce IPVO scheme and flat block selection criteria in Section II. Section III proposes a reversible embedding scheme. The comparison of embedding capacity and image quality between the proposed scheme and related methods is presented in Section IV. Section V is the conclusion of the paper.

II. RELATED WORKS

1. IPVO scheme

Reversible data hiding based on pixel ordering PVO is the newest method for high embedding capacity and good quality proposed by Li et al. in 2013 [18]. PVO is improved [19] to use the whole block with zero prediction error for embedding information to improve embedding. The original image is first divided into non-intersecting blocks of n pixels $((x_1,\ x_2,\dots,\ x_n)$. Blocks containing $\{0, 1, 254, 255\}$ will not be used for embedding and are marked in the map as 1 to avoid underflow and overflow. Here is how [19] has done to embed and extract information from a block of images.

Data embedding algorithm on a block

Input: original pixel sequence $(x_1,\ x_2,\dots,\ x_n)$ and secret data b

Output: The sequence of pixels contains secret data $(y_1,\ y_2,\dots,\ y_n)$.

Embedding process:

Step 1: Sort the original pixel sequences $(x_1,\ x_2,\dots,\ x_n)$ in ascending order to get the sequence $(x_{\sigma(1)},\ x_{\sigma(2)},\dots,\ x_{\sigma(n)})\ x_{\sigma(1)} \le x_{\sigma(2)} \le \dots \le x_{\sigma(n)})$, where $\sigma : \{1,\dots,n\} \to \{1,\dots,n\}$ is the reversible mapping.

Step 2: Find the maximum and minimum prediction errors $d_{max}\ d_{min}$ according to the following formula: $d_{max} = x_{\sigma(n)} - x_{\sigma(n-1)}$ and $d_{min} = x_{\sigma(1)} - x_{\sigma(2)}$

Step 3: Embedding data

Embed at the largest prediction error:

Case 1: If $d_{max} > 1$ then $x'_{\sigma(n)} = x_{\sigma(n)} + 1$

Case 2: If $d_{max} = 1$ and $\sigma(n) < \sigma(n-1)$ then: $x'_{\sigma(n)} = x_{\sigma(n)} + b$ Otherwise $x'_{\sigma(n)} = x_{\sigma(n)} + 1$

Case 3: If $d_{max} = 0$ and $\sigma(n) > \sigma(n-1)$ then: $x'_{\sigma(n)} = x_{\sigma(n)} + b$

Embed at the smallest prediction error:

Case 1: If $d_{min} < -1$ then $x'_{\sigma(1)} = x_{\sigma(1)} - 1$

Case 2: If $d_{min} = -1$ and $\sigma(1) > \sigma(2)$ then $x'_{\sigma(1)} = x_{\sigma(1)} - b$ otherwise $x'_{\sigma(1)} = x_{\sigma(1)} - 1$

Case 3: If $d_{max} = 0$ and $\sigma(1) < \sigma(2)$ then: $x'_{\sigma(1)} = x_{\sigma(1)} - b$

From the reversible mapping it is easy to infer the sequence of pixels contains secret data $(y_1,\ y_2,\dots,\ y_n)$.

Data restoring algorithm on a block

Input: The sequence of pixels contains secret data $(y_1,\ y_2,\dots,\ y_n)$.

Output: original pixel sequence $(x_1,\ x_2,\dots,\ x_n)$ and secret data b

Step 1: Sort the sequences $(y_1, y_2, \dots . y_n)$ in ascending order to get the sequence $(x'_{\sigma(1)},\ x'_{\sigma(2)},\dots,\ x_{\sigma(n)})'$ $x'_{\sigma(1)} \le x'_{\sigma(2)} \le \dots \le x'_{\sigma(n)}$.

The maximum and minimum prediction errors d_{max}, d_{min} are computed according to the following formula:

$d'_{max} = x'_{\sigma(n)} - x'_{\sigma(n-1)}$ and $d'_{min} = x'_{\sigma(1)} - x'_{\sigma(2)}$

Restore at the largest prediction error:

Case 1: $d'_{max} > 2$ In this case, the secret information is

0	0	0	c6
0	0	0	c5
c1	c2	c3	c4

Figure 1. Illustration of the flatness of a block

not embedded and the original pixel is restored according to the formula: $x_{\sigma(n)} = x'_{\sigma(n-1)} - 1$
Case 2: If $d'_{max} = 2$ and $\sigma(n) < \sigma(n-1)$ then: $b = 1, x_{\sigma(n)} = x'_{\sigma(n)} - 1$
Otherwise no secret data is recovered and original pixels are restored as follows: $x_{\sigma(n)} = x'_{\sigma(n)} - 1$
Case 3: $d'_{max} = 1$
If $\sigma(n) < \sigma(n-1)$ then $b = 0, x_{\sigma(n)} = x(\sigma(n))'$
If $\sigma(n) > \sigma(n-1)$ then: $b = 1, x_{\sigma(n)}) = x'_{\sigma(n)} - 1$
Case 4:If d_{max}'=0 then $b = 0, x_{\sigma(n)} = x'_{\sigma(n)}$
Restore at the smallest prediction error:
Case 1: If d_{min}'<-2. In this case, the secret information is not embedded and the original pixel is restored according to the formula: $x_{\sigma(1)} = x'_{\sigma(1)} + 1$
Case 2: If $d'_{min} = -2$ and $\sigma(2) > \sigma(1)$ then: $b = 1, x_{\sigma(1)} = x'_{\sigma(1)} + 1$ otherwise, there is no embedded data and the original pixel is restored according to the formula $x_{\sigma(1)} = x'_{\sigma(1)} + 1$
Case 3: d'_{min}=-1 If $\sigma(2) > \sigma(1)$ then: $b = 0, x_{\sigma(1)} = x'_{\sigma(1)}$
If $\sigma(2) < \sigma(1)$ then: $b = 1, x_{\sigma(1)} = x'_{\sigma(1)} + 1$
Case 4: If $d'_{min} = 0$ then $b = 0, x_{\sigma(1)} = x'_{\sigma(1)}$.
The proposed scheme will use the embedding-extraction algorithm at the smallest value to embed one bit.

2. Determining the flatness of the block

The higher the flatness of the image blocks, the better the image quality, as well as the concealment of the scheme. [24]evaluates the flatness of the current block (blue region of Figure 1) based on neighboring pixels (yellow region of figure 1) according to formula 1:

$$T(I_i) = \frac{1}{mxn}\sqrt{\sum_{i=1}^{m \cdot n}(c_i - s)} \qquad (1)$$

where $s = 1/(m+n+1)\sum_{i=1}^{m+n+1} c_i$
To determine the flat threshold α of the given image and secret information, we need to do iteratively. If $T(I_i) \le \alpha$ then the block Ii will be considered smooth and

III. THE PROPOSED SCHEME

Scheme [19] is considered to be highly efficient, so the proposed scheme applies this algorithm to embed 1 bit at the smallest prediction error. The new point of the paper is to propose a 2-bit embedding algorithm at the largest prediction error. The scheme proposed to divide the subblock by 3 pixels and use the idea of block selection of the scheme [24] to get the better imperceptibility. The embedding and extracting scheme on a block and the whole image are presented below.

1. Embedding algorithm in a block

Given subblock by three pixel $I_i = (x_1, x_2, x_3)$, threshold α and secret data $B = (b_1, b_2, b_3)$. Assume that there are no valuable pixels in the set $\{0{,}253{,}254{,}255\}$ and $T(I_i) \le \alpha$ (T is calculated by the formula (1)), The process of embedding information on this block will be done as follows:
Step 1: Sort (x_1, x_2, x_3) in ascending order to get the sequence $(x_{\sigma(1)}, x_{\sigma(2)}, x_{\sigma(3)})$, $x_{\sigma(1)} \le x_{\sigma(2)} \le x_{\sigma(3)}$, where $\sigma : \{1, 2, n\} \rightarrow \{1, 2, n\}$ is the reversible mapping.
Step 2: Compute the largest prediction error $d_{max} = x_{\sigma(3)}) - x_{\sigma(2)}$ and the smallest prediction error $d_{min} = x_{\sigma(1)} - x_{\sigma(2)}$
Step 3: Embed the secret data
Embed bit b_1 into the smallest prediction error d_{min} by using the IPVO method (section II). Embed b_2, b_3 into the largest prediction error d_{max} as follows: If $d_{max} > 1$ then data is not embedded and $x'_{\sigma(3)} = x_{\sigma(3)} + 3$. $d'_{max} > 4$.
If $d_{max} = 1$ then data is embeded as follows:
If $b_2 = 0, b_3 = 0$ then$x'_{\sigma(3)} = x_{\sigma(3)}$. d'_{max} will equal to 1.
If $b_2 = 0, b_3 = 1$ then $x'_{\sigma(3)} = x_{\sigma(3)} + 1$. d'_{max}will equal to 2.
If $b_2 = 1, b_3 = 0$ then $x'_{\sigma(3)} = x_{\sigma(3)} + 2$ d'_{max} will equal to 3.
If $b_2 = 1, b_3 = 1$ then $x'_{\sigma(3)} = x_{\sigma(3)} + 3$. d'_{max} will equal to 4.
It is easy to see that $x_{\sigma(2)} = x'_{\sigma(2)}$ and $x'_{\sigma(1)} \le x'_{\sigma(2)} \le x'_{\sigma(3)}$
Step 4: Rearrange $(x'_{\sigma(1)}, x'_{\sigma(2)}, x'_{\sigma(3)})$ to get (y_1, y_2, y_3).
The following example describes the embedding process of the current tile (colored tile). In case of being able to embed in Fig. 2 and in case of not being able to embed in Fig. 3.

No embedded data case:

2. Extracting and restoring algorithm in a block

Given a block of 3 pixels $I_i = (y_1, y_2, y_3)$ and threshold α. Assume the location map of the block is 0 which means this block contains secret information and $T(I'_i) \le \alpha$ then the data and the host image recovery is performed as

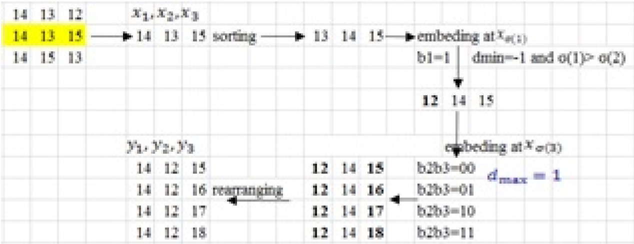

Figure 2. Example of embedded case

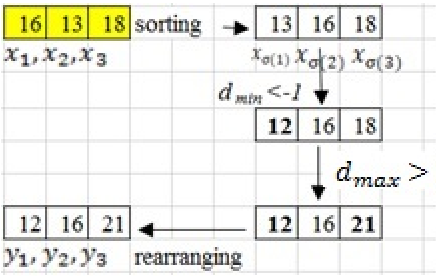

Figure 3. Example of non-embedded case

follows:

Step 1: Sort (y_1, y_2, y_3) to get an ascending sequence$(y_{\sigma(1)}, y_{\sigma(2)}, y_{\sigma(3)})$

Step 2: Compute $d'_{max} = y_{\sigma(3)} - y_{\sigma(2)}$ and $d'_{min} = y - y_{\sigma(2)}$

Step 3: Restore the data Restore bit b_1 and pixel x_1 based on the smallest predictor d'_{min} as the IPVO method presented in Section II.

Based on the largest predictor d'_{max} to recover the remaining b_2, b_3 and 2 pixels x_2, x_3 as follows:

If $d'_{max} > 4$ then no secret information is extracted, $x_{\sigma(3)} = y_{\sigma(3)} - 3$, otherwise, perform data recovery as follows:

If $d'_{max} = 1$ then $b_2 = 0, b_3 = 0$ and $x_{\sigma(3)} = y_{\sigma(3)}$.

If $d'_{max} = 2$ then $b_2 = 0, b_3 = 1$ and $x_{\sigma(3)} = y_{\sigma(3)} - 1$.

If $d'_{max} = 3$ then $b_2 = 1, b_3 = 0$ and $x_{\sigma(3)} = y_{\sigma(3)} - 2$.

If $d'_{max} = 4$ then $b_2 = 1, b_3 = 1$ and $x_{\sigma(3)} = y_{\sigma(3)} - 3$.

Step 4: Rerange $(x_{\sigma(1)}, x_{\sigma(2)}, x_{\sigma(3)})$ to get the original image (x_1, x_2, x_3).

Below is an illustration of the process of restoring the original image and embedding information. In case there is embedded data Fig. 4 and in case there is no embedded information in Fig. 5. An example of restoring the original image in case of unembedding is as follows

3. Data embedding procedure

Input: The host image I, threshold α the secret data B

Output: The stego image I'

Embedding secret information B into image I is done as follows:

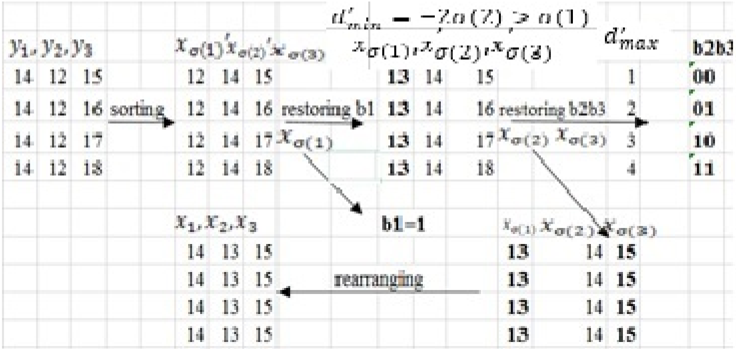

Figure 4. An example of the case of recovering block containing information

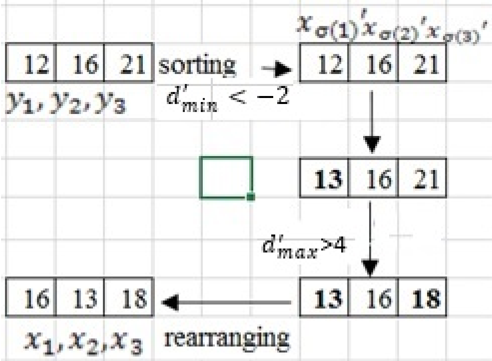

Figure 5. An example of the case of recovering block not containing information

Step 1: Divide the image into nonoverlapping subblocks of length 3 pixels (browsing the image in order from left to right and from top to bottom).

Step 2: The location map is created according to the formula below:

$$LM(I_i) = \begin{cases} 1, & \text{if } \exists x_j \in \{0, 223, 224, 225\}, j = 1, 2, 3 \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

Step 3: Compress the map in a lossless manner to obtain a compressed map denoted by CLM (assuming the CLM has length L).

Step 4: Specify extra information to be embedded into the image to be used for image recovery purposes as follows:

Step 5: Set the data embedded in the image as: secret data $B + LSB54$

where $LSB54$ is $(54+L)$ low bits of the embedded element data ta for extracting is explained in Step 6.

Step 6: Embed $(54 + L)$ bit B into I by traversing each sub-block in turn, assuming the i-th sub-block is denoted by I_i. If $LM(I_i) = 1$ then skip and browse to the next block, otherwise embedding in this block according to the

TABLE I
PSNR OF THE STEGO IMAGES IN MAXIMAL EMBEDDING CASE

	Information	Size of bit
1	Block size	6 bits
2	Threshold α	8 bits
3	Low bit data embedding start block position (BO)	16 bits
4	The size of compressed location map	24 bits
5	The compressed location map CLM	L

algorithm mentioned in section 3.1 get I_J. Take the low $54+L$ bits of the first pixels of I_J to get $LSB54$. Continue embedding until the end B. Locate the the first block which is embedded $LSB54(BO)$, continue to embed all $LSB54$ according to the algorithm in section3.1.
Step 7: Embeds extra information by low bit insertion into the first $54+L$ pixels. The result is stego image I'.

4. Data extracting procedure

Input: stego image I'
Output: The host image I, threshold α the secret data B
Step 1: Using the low-bit insertion method LSB extracts 6, 8, 16, and 24 bits respectively to get the tile size, threshold α, starting block position BO and compressed map size L. Continue using LSB with Next L bits to get the CLM compressed map. Extract the CLM to get the LM location map.
Step 2: Divide the image into sub-blocks of 3 pixels (1×3 or 3×1) as embedding. From the block location BO, refer to the values of $LM(j)(j > BO)$, perform $(54+L)LSB54$ bit extraction and recover the original tiles according to the algorithm mentioned in section 3.2, in blocks j has $LM(j) = 0$. Insert bits from $LSB54$ into the first $54+L$ pixels by LSB method.
Step 3: Reference $LM(i)(i >= 1)$, extract data bits from the first blocks to the $BO-1$ block corresponding to blocks with $LM(i) = 0$ in the message containing image B and recover the original image according to the algorithm described in section 3.2. By on embedded data B and the original image I fully restored.

IV. EXPERIMENTAL RESULTS

Grayscale images[1] with size 512×512 as Fig. 6 below are used to investigate the embedding ability and image quality of the proposed schema compared to schema has the same method. The test tool is Matlab2016, windows10. Below are the results of the embedding capacity survey of the methods.

Tables II and III above are the results of embedding testing when dividing 3×1 and 1×3 blocks, eliminating the compressed map storage space of the proposed schema and related schemas. The proposed method has a higher embedding capacity than PVO, IPVO, PVOK, and GePVOK methods by 19%, 15%, 18%, and 15%, respectively, If the subblock is divided as 3×1.
Next, we use $PSNR$ to test the imperceptibility of the proposed scheme and several widely PVO based schemes. $PSNR$ is calculated by using the following Equation:

$$PSNR = 10\log_{10}\frac{255^2}{MSE}, \quad (3)$$

where I, I' are the original image and the image containing data, respectively with M rows N columns, where the mean square error (MSE) between the original and embedded image is defined as:

$$MSE = \frac{1}{MN}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}(I(i,j) - I'(i,j))^2, \quad (4)$$

The results are shown in the figures, from Fig. 7 to Fig. 15. The above figures show that the $PSNR$ value of the GePVO scheme is the lowest, and the second-lowest is PVOK, in most cases then the proposed scheme has the highest PSNR value. Particularly in Figure 9 then the $PSNR$ of the IPVO and the proposed schemes equal approximately. Table 3 below is the result of recording the $PSNR$ value of the methods in the case of maximum embedding.

TABLE II
COMPARISON OF EMBEDDING CAPACITY WITH SUBBLOCK SIZE 1×3

Image	PVO	IPVO	PVOK	GePVO	Proposed scheme
1	28040	32681	30640	35536	49215
2	15952	22872	16776	18096	34541
3	30736	40333	35396	47222	60298
4	47155	55311	52468	61375	82427
5	43185	55395	49526	64298	82510
6	29530	34538	32730	37651	52168
7	33449	36821	35931	38350	54948
8	6328	42782	6954	17655	63918
9	29506	31899	31453	33228	47937
10	30757	32691	32462	33652	49114
TB	29464	38532	32434	38706	57708

TABLE III
COMPARISON OF EMBEDDING CAPACITY WITH SUBBLOCK SIZE 3×1

Image	PVO	IPVO	PVOK	GePVO	Proposed scheme
1	30876	36826	33678	39974	55364
2	13586	18879	14185	15198	28480
3	34332	47135	39336	54488	70712
4	47335	59182	53388	66299	88304
5	45557	57199	51898	65663	85053
6	27407	31734	30145	34392	47867
7	39620	45263	42781	47012	67694
8	6121	42219	6659	17710	63005
9	27178	29831	28799	30460	44677
10	31658	33576	33303	34540	50313
TB	30367	40184	33417	40574	60147

[1]taken from https://sipi.usc.edu/database/

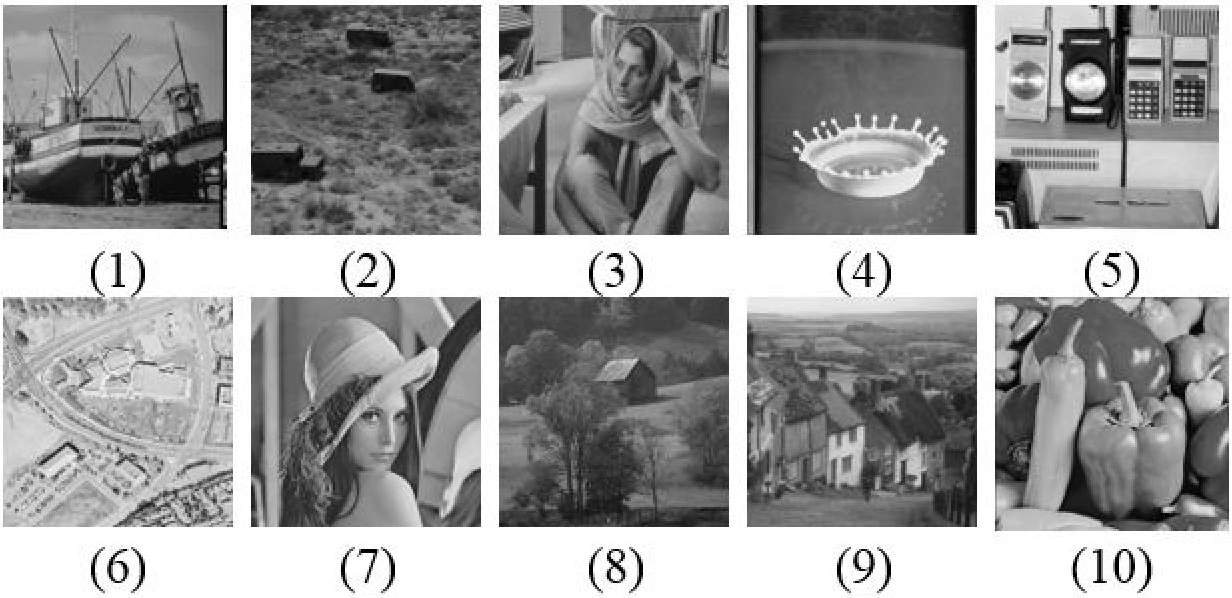

Figure 6. Experimental images

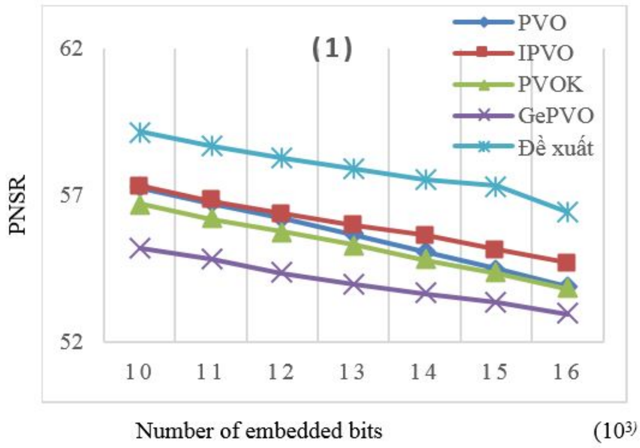

Figure 7. Comparisons in terms of quality image between methods for image(1)

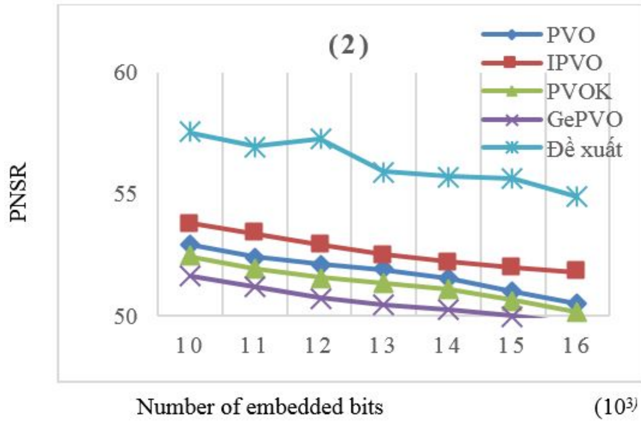

Figure 8. Comparisons in terms of quality image between methods for image(2)

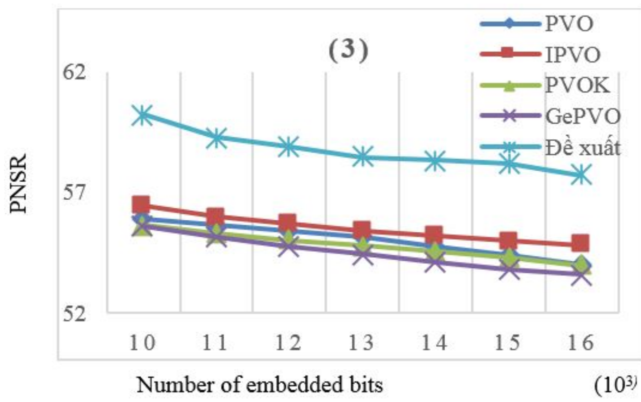

Figure 9. Comparisons in terms of quality image between methods for image(3)

With the number of embedding bits of the proposed scheme higher than that of PVO, PVOK, IPVO, and GePVO schemes at 19%, 15%, 18%, and 15%, respectively, the image quality of the proposed scheme is still high. ($PSNR$ of the proposed scheme is above 40). Therefore, it can be concluded that the proposed scheme gets good image quality.

V. CONCLUSION

The article has researched and improved the reversible hiding method based on the improved pixel value orderring method, which is considered as one of the efficient reversible data hiding technique. The proposed technique

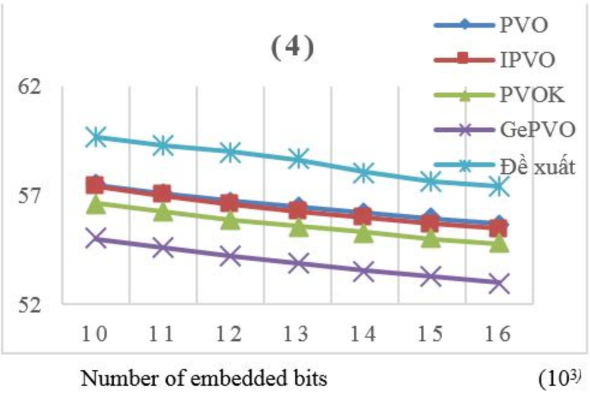

Figure 10. Comparisons in terms of quality image between methods for image(4)

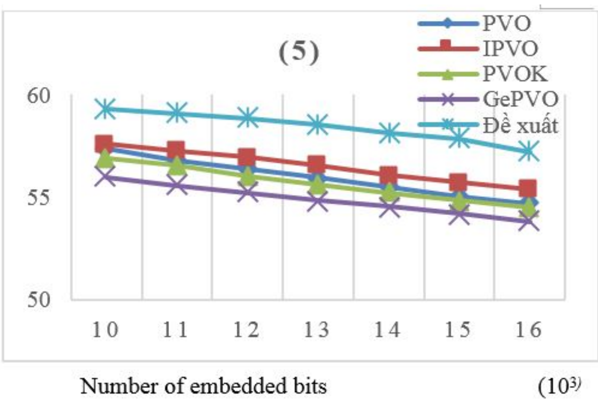

Figure 11. Comparisons in terms of quality image between methods for image(5)

image(6).

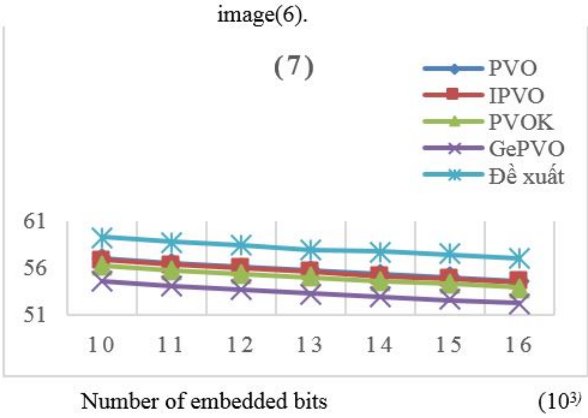

Figure 12. Comparisons in terms of quality image between methods for image(6)

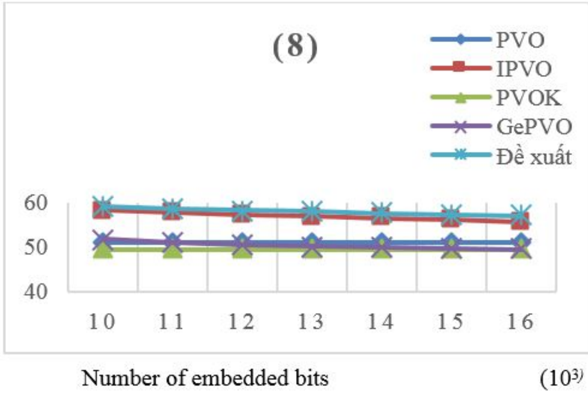

Figure 13. Comparisons in terms of quality image between methods for image(7)

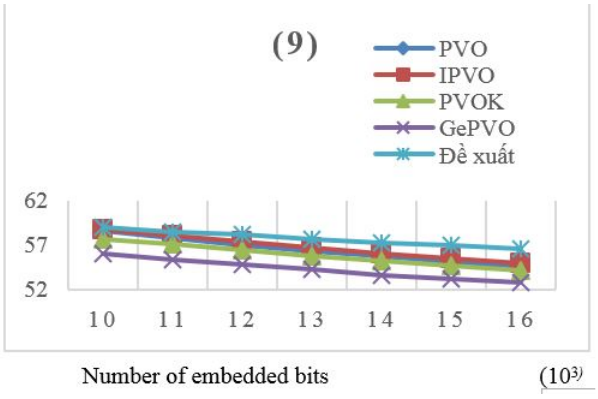

Figure 14. Comparisons in terms of quality image between methods for image(8)

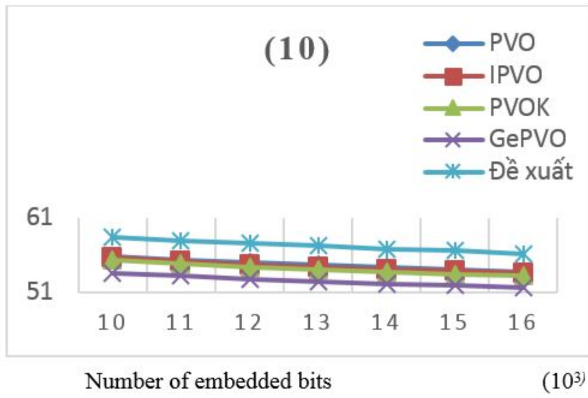

Figure 15. Comparisons in terms of quality image between methods for image(9)

TABLE IV
PSNR OF THE STEGO IMAGES IN MAXIMAL EMBEDDING CASE

Image	PVO	IPVO	PVOK	GePVO	Proposed scheme
1	57.24	57.30	56.69	55.20	51.05
2	56.71	56.79	56.16	54.79	50.55
3	56.22	56.38	55.74	54.33	50.07
4	55.64	55.98	55.29	53.97	49.66
5	55.06	55.61	54.79	53.64	49.37
6	54.47	55.16	54.33	53.33	49.04
7	51.56	52.23	51.11	50.28	45.83
8	52.93	53.82	52.46	51.66	47.28
9	52.42	53.37	51.93	51.19	46.85
10	51.91	52.53	51.36	50.49	46.12
Average	54.41	54.92	53.99	52.89	48.58

of embedding three bits per subblock with three pixels in the article has significantly enhanced embedding ability compared with schemes with the same method. The experiment results also show that the embedding capacity of the proposed method is higher than that of PVO, PVOK, IPVO, GePVO methods by $19\%, 15\%, 18\%$ and 15%, respectively. The image quality of the proposed scheme is also high. The next goal is to improve the map to increase the space for embedding purposes, thereby continuing to improve the embedding capacity. Besides, we study the direction of reducing image variability to enhance image quality.

REFERENCES

[1] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, *Digital watermarking and steganography*. Morgan kaufmann, 2007.

[2] D. Wu and W. Tsai, "A steganographic method for images by pixel-value differencing," *Pattern recognition letters*, vol. 24, no. 9-10, pp. 1613–1626, 2003.

[3] L. Mielikainen, "Lsb matching revisited," *signal processing lettersl*, vol. 13, no. 5, pp. 285—-287, 2006.

[4] J. Tian, "Reversible data embedding using a difference expansion," *IEEE transactions on circuits and systems for video technology*, vol. 13, no. 8, pp. 890–896, 2003.

[5] A.M.Alattar, "Reversible watermark using the difference expansion of a generalized integer transform," *IEEE transactions on image processing*, vol. 13, no. 8, pp. 1147–1156, 2004.

[6] Y. Hu, H. Lee, and J. Li, "De-based reversible data hiding with improved overflow location map," *IEEE Trans. Circuits Syst. Video Technol*, vol. 19, no. 2, pp. 250–260, 2009.

[7] C. T. Luyen and P. V. At, "An efficient reversible watermarking method and its application in public key fragile watermarking," *Applied Mathematical Sciences*, vol. 11, no. 30, pp. 1481–1494, 2017.

[8] D. Thodi and J. Rodriguez, "Expansion embedding techniques for reversible watermarking," *IEEE transactions on image processing*, vol. 16, no. 3, pp. 721–730, 2007.

[9] W. Hong, T.-S. Chen, and C.-W. Shiu, "Reversible data hiding for high quality images using modification of prediction errors," *Journal of Systems and Software*, vol. 82, no. 11, pp. 1833–1842, 2009.

[10] D. Coltuc, "Improved embedding for prediction-based reversible watermarking," *IEEE Transactions on Information Forensics and security*, vol. 6, no. 3, pp. 873–882, 2011.

[11] G. Xuan, C. Yang, Y. Zhen, Y. Shi, and Z. Ni, "Reversible data hiding using integer wavelet transform and companding technique," in *Lecture Notes in Computer Science book series*, vol. 3304. Springer, 2004, pp. 115–124.

[12] F. Peng, X. Li, and B. Yang, "Adaptive reversible data hiding scheme based on integer transform," *Signal Process*, vol. 92, pp. 54–62, 2012.

[13] M. Arsalan, S. Malik, and A. Khan, "Intelligent reversible watermarking in integer wavelet domain for medical images," *J Syst Softw*, vol. 18, no. 4, pp. 883–894, 2012.

[14] F. Li, Q. Mao, and C. C. Chan, "A reversible data hiding scheme based on iwt and the sudoku method," *J Syst Softw*, vol. 18, no. 3, pp. 410–419, 2016.

[15] Z. Ni, Y. Q. Shi, N. Ansari, and W. Su, "Reversible data hiding," *Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 3, pp. 354–362, 2006.

[16] X. Wang, X. Li, B. Yang, and Z. Guo, "Efficient generalized integer transform for reversible watermarking," *IEEE Signal Process. Lett*, vol. 17, no. 6, pp. 567–570, 2010.

[17] H. Wu and J. Huang, "General framework to histogram shifting based reversible data hiding," *Signal Processl*, vol. 22, no. 6, pp. 2181–2191, 2013.

[18] X. Li, J.Li, B. Li, and B. Yang, "High fidelity reversible data hiding scheme based on pixel value ordering and prediction error expansion," *Signal Process*, vol. 93, pp. 198–205, 2013.

[19] F.Peng, X.Li, and B.Yang, "Improved pvo-based reversible data hiding," *Digital Signal Processing*, vol. 25, pp. 255–265, 2014.

[20] X.Qu and H.J.Kim, "Pixel-based pixel value ordering predictor for high-fidelity reversible data hiding," *Signal Processing*, vol. 111, pp. 249–260, 2015.

[21] J.Li, Y.H.Wu, C.F.Lee, and C.C.Chang, "Generalized pvo-k embedding technique for reversible data hiding." *Int. J. Netw. Secur.*, vol. 20, no. 1, pp. 65–77, 2018.

[22] L.Q.Hoa, C.T.Luyen, N.K.Sao, and P.V.At, "An improved reversible watermarking based on pixel value ordering and prediction error expansion," in *Asian Conference on Intelligent Information and Database Systems*. Springer, 2020, pp. 582–592.

[23] Z. Pan and E. Gao, "Reversible data hiding based on novel embedding structure pvo and adaptive block-merging strategy," *Multimedia Tools and Applications*, vol. 78, no. 18, pp. 26 047–26 071, 2019.

[24] N.T.Phuc and C.T.Luyen, "A reversible data hiding based on generalized pixel value ordering," *Transport and Communications Science Journal*, vol. 72, no. 2, pp. 193–203, 2021.

Luyen Cao Thi Born on April 28, 1979 in Hung Yen. Graduated from the University in 2001, master in 2005 from the University of Science and Technology and received a doctorate in 2018 from the Military Institute of Science and Technology. Lecturer at Faculty of Information Technology - University of Transport and Commutications Areas of interest: Information hiding, digital watermarking, information security.
Email: luyenct@utc.edu.vn.

Phong Pham Dinh received a Master degree in Information Technology and a Doctor of Philosophy degree in Computer Science from University of Engineering and Technology, Vietnam National University, Hanoi in 2011 and 2018, respectively. Now, he is a lecturer of University of Transport and Communications. His research interests include hedge algebras, fuzzy systems, soft computing, data mining and machine learning.
Email: phongpd@utc.edu.vn

Hiep Pham Hoang graduated the High School for Gifted Students, Hanoi University of Science, Vietnam National University in 2022. His research interests include competitive programming, algorithms and machine learning.
Email: phamhoanghiep03092004@gmail.com