DOI: 10.23977/cpcs.2022.060204
ISSN 2371-8889 Vol. 6 Num. 2

A High Capacity Reversible Data Hiding Technique with Improved Pixel Value Ordering Method

Ke Luo, Renzhi Yuan

Institute of Library, Shaoyang University, Shaoyang, Hunan, China

Keywords: Pixel Value Order, Reversible Data Hiding, Speed Dial, Block Subdivision

Abstract: In order to fully utilize each pixel within an image block, improve the embedding capability, and achieve high hiding quality of reversible data hiding algorithm, a pixel-ordered information hiding scheme that incorporates speed dial embedding strategy and block subdivision technique is proposed. Firstly, the principle of HiSB and LoSB is used to split the original image into quotient image and remainder image, and then the quotient image is divided into 3×3 quotient blocks, and the quotient blocks are subdivided into blocks with the speed dial embedding strategy, and the difference is calculated after sub-block sorting, and the secret data is embedded according to the predicted difference. The experimental results show that each image block can be embedded with a maximum of 8 bits of data, and its embedding block ratio reaches 89.77% on average, which effectively improves the embedding capacity while the quality of the stego image remains relatively stable, and is also robust against unintended attacks on image processing.

1. Introduction

Reversible data hiding technique is a method to hide secret data in the stego image by only slightly changing the stego image pixels, and its biggest advantage is to restore the original image completely and also extract the secret data from the stego image without loss [1], which is suitable for medical image processing [2-4], military image processing, satellite maps, and other sensitive fields that require zero-tolerance image distortion [5].

The reversible data hiding methods proposed in recent years are mainly divided into three categories: Difference Expansion (DE) [6], Histogram Shifting (HS) [7], and Prediction Error Expansion (PEE) [8], in which the Prediction Error Expansion method is the most commonly used, and the principle mainly uses the difference of pixel values to embed secret data, and its embedding amount is twice higher than that of the Difference Expansion method.

Given the highly structured nature of images, scholars usually choose images as stego for data hiding. In order to make full use of the highly correlated neighboring pixel values of images, Li proposed the first reversible data hiding algorithm based on Pixel Value Ordering (PVO) [9-10] method in 2013 in combination with the prediction error method. The stego image is divided into non-overlapping equal-sized sub-blocks, and the pixels within the blocks are ordered to calculate the maximum prediction error and minimum prediction error values, and secret data can be embedded if it falls at +1 or -1 peak point, and is not used if it is 0. Peng found that the prediction error value of 0 is more common in histograms, and the embedding of secret data can be performed when the peak point is 0. He proposed an improved PVO algorithm (IPVO) [11]. Ou improved on IPVO and proposed the PVO-K [12,13] method, which performs better than PVO in terms of embedding volume and is suitable for smoothed images. Wang proposed a quadratic approach [15 13], which allows more secret data to be embedded in the smoothed sub-blocks. In 2015, Qu proposed a reversible pixel ranking method with high realism (pixel-based pixel value ordering, PPVO) [14], which uses a floating window approach to embed data block by block, and the average maximum hiding capacity of PPVO is 13,500 bits higher than that of PVO, and the image embedding capacity is improved to 2.3 dB on average, but only up to 1 bit of secret data can be embedded each time the floating window is executed.

In order to effectively utilize the pixel space of the stego image to improve the embedding capability of reversible data hiding technique, this paper proposes an idea based on PVO block repartitioning and speed dial embedding strategy, which makes full use of the connectivity between pixels to divide a commercial image block into four sub-blocks and embed data in each block according to the prediction error value, which improves the embedding capacity of the algorithm on the basis of maintaining the visual quality of the stego image.

2. Related Work

2.1. Reversible Hiding Techniques Based on Pixel Value Ordering

The pixel value ordering technique (PVO) [10] slices the image into multiple equal-sized sub-blocks and sorts the pixel values within the blocks to calculate the difference between the predicted maximum pixel value and the next largest pixel value (or the smallest pixel value and the next smallest pixel value), which is calculated according to Eq. (1) and Eq. (2). Embedding data according to the maximum or minimum pixel value greatly increases the embedding capacity of PVO.

$$PE'_{max} = \begin{cases} PE_{max} & if \quad PE_{max} = 0 \\ PE_{max} + b & if \quad PE_{max} = 1 \\ PE_{max} + 1 & if \quad PE_{max} > 1 \end{cases} \tag{1}$$

$$PE'_{min} = \begin{cases} PE_{min} & if \quad PE_{min} = 0 \\ PE_{min} - b & if \quad PE_{min} = -1 \\ PE_{min} - 1 & if \quad PE_{max} < -1 \end{cases} \tag{2}$$

where is secret data, $b \in \{0,1\}$.

When the prediction error value PE_{max} equals to 1 or PE_{min} equals to -1, the embedding of the secret data can be performed. If when PE_{max} , the maximum value is added to the secret data b to obtain the disguised pixel value; if then $PE_{min} = -1$, the minimum value is subtracted from the secret data b to obtain the disguised pixel value.

Take Fig. 1 as an example, suppose the original image *X*= {35, 40, 45, 46}, after sorting the pixel values first, use the maximum value minus the next largest value to get, if the secret data is embedded, add the maximum value to the secret data to get the pixel value containing secret data $X' = 46 + 1 = 47$.

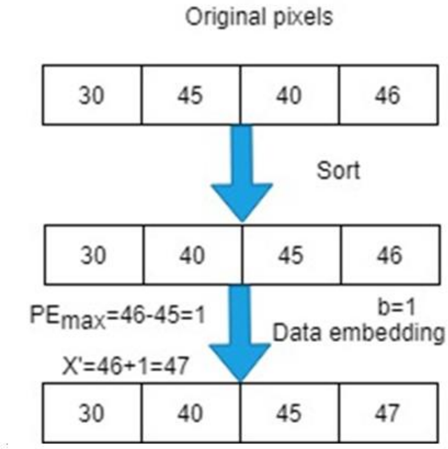

Figure 1: Demonstration diagram of PVO method embedding

2.2. Significant-Bit-Difference Expansion

Wang proposed the reversible data hiding method of Significant-bit-difference expansion (SBDE) [12 15], which decomposes the pixels in an image into two parts, the higher significant bits (HiSB, called the quotient image) and the lower significant bits (LoSB, called the residual image). It uses the higher quotient image in pixel value as the secret information embedding object, and embeds the secret data into the quotient image by calculating the difference between neighboring pixels. Since general image processing or attacks make modifications in the lower residual image, hiding the data in I_{HiSB} preserves the integrity of the data content better and enhances the robustness of the embedding method.

2.2.1. Embedding Method for Significant-Bit-Difference Expansion

Step 1: Split the image into I_{HiSB} and I_{LoSB}.

Step 2: Calculate the difference between adjacent pixels of the whole image $v_{i,j}$ and set the peak points pp_1 and pp_2.

Step 3: The secret data is embedded in I'_{HiSB} along with the difference according to Equation (3) and Equation (4).

$$I'_{HiSB(i,j)} = \begin{cases} I_{HiSB(i,j+1)} + v_{i,j} + s & if \quad v_{i,j} = pp_1 \\ I_{HiSB(i,j+1)} + v_{i,j} - s & if \quad v_{i,j} = pp_2 \\ I_{HiSB(i,j+1)} + v_{i,j} + 1 & if \quad v_{i,j} > pp_1 \\ I_{HiSB(i,j+1)} + v_{i,j} - 1 & if \quad v_{i,j} < pp_2 \\ I_{HiSB(i,j+1)} + v_{i,j} & otherwise \end{cases} \tag{3}$$

$$I'_{HiSB(i,1)} = \begin{cases} I_{HiSB(i+1,1)} + v_{i,1} + s & if \quad v_{i,1} = pp_1 \\ I_{HiSB(i+1,1)} + v_{i,1} - s & if \quad v_{i,1} = pp_2 \\ I_{HiSB(i+1,1)} + v_{i,1} + 1 & if \quad v_{i,1} > pp_1 \\ I_{HiSB(i+1,1)} + v_{i,1} - 1 & if \quad v_{i,1} < pp_2 \\ I_{HiSB(i+1,1)} + v_{i,1} & otherwise \end{cases} \tag{4}$$

Step 4: Confirm that there is no overflow problem according to equation (5) and equation (6).

$$I'_{HiSB(i,j)} = \begin{cases} I_{HiSB(i,j+1)} + v_{i,j} + s & if \quad I_{HiSB(i,j+1)} + v_{i,j} < MAX - 1 \\ I_{HiSB(i,j+1)} + v_{i,j} - s & if \quad I_{HiSB(i,j+1)} + v_{i,j} > 1 \\ I_{HiSB(i,j+1)} + v_{i,j} + 1 & if \quad I_{HiSB(i,j+1)} + v_{i,j} < MAX - 1 \\ I_{HiSB(i,j+1)} + v_{i,j} - 1 & if \quad I_{HiSB(i,j+1)} + v_{i,j} > 1 \end{cases} \tag{5}$$

$$I'_{HiSB(i,j)} = \begin{cases} I_{HiSB(i+1,1)} + v_{i,1} + s & if \quad I_{HiSB(i+1,1)} + v_{i,1} < MAX - 1 \\ I_{HiSB(i+1,1)} + v_{i,1} - s & if \quad I_{HiSB(i+1,1)} + v_{i,1} > 1 \\ I_{HiSB(i+1,1)} + v_{i,1} + 1 & if \quad I_{HiSB(i+1,1)} + v_{i,1} < MAX - 1 \\ I_{HiSB(i+1,1)} + v_{i,1} - 1 & if \quad I_{HiSB(i+1,1)} + v_{i,1} > 1 \\ I_{HiSB(i+1,1)} + v_{i,1} & otherwise \end{cases} \tag{6}$$

Step 5: After embedding all the secret data, the obtained I'_{HiSB} is merged with the original I_{LoSB} to obtain the disguised pixel I' .

Taking Fig. 2 as an example, suppose an original image I, first set n = 3 to divide the original image I into 2 planes, which are images I_{HiSB} and images I_{LoSB}. Then the I_{HiSB} is used as a stego for embedding secret data, calculating the I_{HiSB} individual adjacent pixels in the first column to calculate the difference is $I_{HiSB(i,1)}$-$I_{HiSB(i+1,1)}$, the remaining difference is calculated as $I_{HiSB(i,j)}$-$I_{HiSB(i,j+1)}$, and in order to be reversible, the first pixel on the upper left of the image does not change, and after calculating the difference, the peak point is set as pp_1=1 and pp_2=0. Embedding is possible when the difference is equal to 0 or 1, and displacement is performed if it is greater than 1 or less than 0. The secret data to be embedded is S= $(1101001)_2$, and the secret data is embedded when the difference is 0 or 1, while the rest is shifted by 1 bit to produce I'_{HiSB}, and then Combining I'_{HiSB} and I_{LoSB} are merged to produce the image I'.

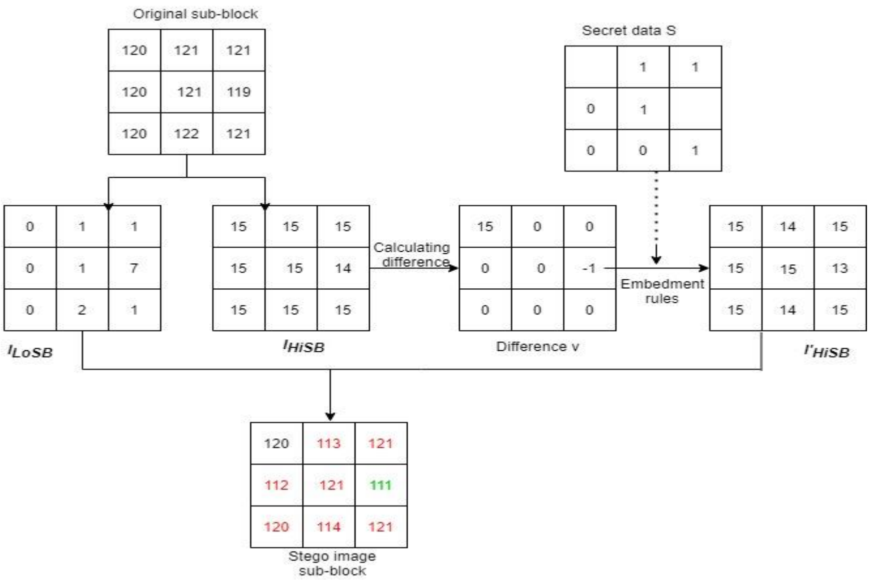

Figure 2: The SBDE embedding process

3. Proposed SDPVO-Based Reversible Information Hiding Method

The proposed SDPVO (Speed Dial PVO) method based on the nine-box grid, the idea of which originates from the nine-box grid number game passed down in ancient China, can generate four blocks of pixels in a 3×3 size sub-block with the center point as the reference, treating the three pixel values of each block as a group, adopting the prediction error expansion method of Li scholars PVO method, calculating the prediction error after sorting these three pixels and embed the secret data at the extreme value point of +1 and -1, then use the concept of Qu's PPVO floating window to execute the four groups within this sub-block sequentially, and put them back into the sub-block after finishing embedding, so that the pixel values can be fully used.

3.1. SDPVO Embedding Algorithm

Input: original image I and secret data S

Output: the stego images I'

Step 1: Receive the original image I and the secret data S to be embedded.

Step 2: Split the original image I into two images: the quotient image (I_{HiSB}) and the remainder image (I_{LoSB}), where pixel values of the image I_{LoSB} are $\{x|0\leq x\leq 32\}$.

Step 3: Split image I_{HiSB} into non-overlapping blocks of 3×3 size.

Step 4: Number and group the pixel values in the sub-block by raster scan.

The pixel values within the sub-block are numbered $A=[a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9]$, and the pixel values that are not adjacent to each other are taken as a group with a_5 as the center point, and four groups can be obtained, respectively: $A_1=[a_1, a_5, a_7]$ 、 $A_2=[a_2, a_5, a_6]$ 、 $A_3=[a_3, a_5, a_9]$ 、 $A_4=[a_4, a_5, a_8]$.

Step 5: Then, the pixel values in each group are sorted from small to large, which satisfying Equation (7).

$$A_{\pi(n)}=\{a_{\pi(1)},a_{\pi(2)},a_{\pi(3)}\,|\,n=1\sim 4\}$$
$$a_{\pi(1)}\leq a_{\pi(2)}\leq a_{\pi(3)} \tag{7}$$

Step 6: Establish the location map (LM) of $A_{\pi(n)}$, and record whether the group overflows.

$$\begin{cases} a_{\pi(3)}-a_{\pi(2)}<1 \quad or \quad a_{\pi(3)}\neq 32 \\ a_{\pi(1)}-a_{\pi(2)}>-1 \quad or \quad a_{\pi(1)}\neq 0 \end{cases} \tag{8}$$

Determine the overflow bit or underflow bit according to Formula (8). If the grouping all meet the above conditions, it means that the maximum value adjustment will produce overflow bits, or the minimum value adjustment will produce underflow bits, and there will be no adjustment at both ends of the sub-block, giving sub-block LM = 1. In addition to that, which means that at least one end of the group can be embedded with secret data or shifting, thus giving sub-block LM = 0.

Step 7: The prediction error values are obtained by calculation.

When there is no pixel value for the over/under bit problem (LM = 0), embedding of secret data and expansion of the prediction error value can be performed. As in Equation (9), the maximum prediction error $PE_{\max}$ is obtained by subtracting the middle pixel value $a_{\pi(2)}$ from the maximum pixel value $a_{\pi(3)}$ in the sorted $a_{\pi(n)}$ group; the minimum prediction error $PE_{\min}$ is obtained by subtracting the middle pixel value $a_{\pi(2)}$ from the minimum pixel value $a_{\pi(1)}$.

$$\begin{cases} PE_{\max}=a_{\pi(3)}-a_{\pi(2)} \\ PE_{\min}=a_{\pi(1)}-a_{\pi(2)} \end{cases} \tag{9}$$

Step 8: Embed secret data.

Adjust the prediction error by embedding the secret data in the extreme values, in which the extreme points are set to 1 and -1. On the histogram of the prediction error, the value on either side of the extreme point is expanded by one unit, and one bit of secret data can be embedded in the extreme point, or if the value is 0, no adjustment is done.

$$PE'_{\max} = \begin{cases} 0, if \quad PE_{\max} = 0 \\ s, if \quad PE_{\max} = 1 \\ 1, if \quad PE_{\max} > 1 \end{cases} \tag{10}$$

$$PE'_{\min} = \begin{cases} 0, if \quad PE_{\min} = 0 \\ -s, if \quad PE_{\min} = -1 \\ -1, if \quad PE_{\min} < -1 \end{cases} \tag{11}$$

Step 9: Adjust the pixel values in the sub-block.

With $PE'_{\max}$ and $PE'_{\min}$ acting as the prediction error values for the embedded secret data, the maximum pixel value $x_{\pi(3)}$ and the minimum pixel value $x_{\pi(1)}$ in the $A_{\pi(n)}$ group are modified, and the middle pixel values remain unchanged. The adjustment is as indicated in Equation (12).

$$\begin{cases} x'_{\pi(3)} = x_{\pi(3)} + PE'_{\max} \\ x'_{\pi(1)} = x_{\pi(1)} + PE'_{\min} \\ x'_{\pi(2)} = x_{\pi(2)} \end{cases} \tag{12}$$

Step 10: The modified pixel $x'_{\pi(1)}, x'_{\pi(2)}, x'_{\pi(3)}$ obtained by modifying the value in the original position corresponding to A_n to obtain the modified A'_n group. Place A'_n back into the sub-block and take the next A_n -group and repeat steps 5 to 9 until completing the four groups of the sub-block.

Step 11: Producing a stego quotient image.

When the modified sub-block is put back into the image, the next sub-block is taken and steps 3 to 9 are performed to produce the stego quotient containing image I'_{HiSB}.

Step 12: The additional information LM is embedded and the stego quotient image (I_{HiSB}) is synthesized with the remainder image (I_{LoSB}) as the dense image I'.

3.2. Secret Data Extraction and Image Rrestore Algorithm

Input: stego image I'

Output: original image I, secret data S

Step 1: Remove the relevant additional information LM from the stego image I'.

Step 2: Split the stego image into non-repeating sub-blocks of size 3×3.

Step 3: Number and group the pixel values in the sub-blocks by raster scan.

The pixel values in the sub-block are numbered $A = [a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9]$, and the pixel values that are not adjacent to each other are grouped together with a_5 as the center point, and four groups can be obtained. The embedding step is executed in the order of A1 to A4, and a_5 may be adjusted several times, so the secret data should be taken out from A4 to A1 by reverse operation.

Step 4: Then, the pixels within each group are sorted from smallest to largest to obtain $A'_{\pi(n)}$, which satisfies Equation (13)

$$A'_{\pi(n)} = \{a'_{\pi(1)}, a'_{\pi(2)}, a'_{\pi(3)} \mid n = 1 \sim 4\}$$
$$a'_{\pi(1)} \le a'_{\pi(2)} \le a'_{\pi(3)} \tag{13}$$

Step 5: Judge whether secret data is embedded based on the location map LM.

Judge whether the group has embedded secret data based on the LM recorded in the extra information. If LM = 0, means the group has embedded secret data and can follow the next step; if LM = 1, means the group does not have embedded secret data, skip the adjustment step and go back to step 4, to executing the next group $A'_{\pi(n)}$.

Step 6: The prediction error value is obtained by calculation.

For the adjusted group (LM = 0) a secret data extraction and reduction step was performed to obtain the maximum and minimum prediction error values in the sorted $A'_{\pi(n)}$ group.

$$\begin{cases} PE'_{\max} = a'_{\pi(3)} - a'_{\pi(2)} \\ PE'_{\min} = a'_{\pi(1)} - a'_{\pi(2)} \end{cases} \tag{14}$$

Step 7: Extract secret data and restore the prediction error.

Knowing that the extreme value points are 1 and -1, take out the secret data and restore the prediction error value according to equations (15) and (16). The maximum prediction error value $PE_{\max}$, if $PE_{\max}$ = 0, means not embedded secret data or displacement, then no need to restore; if $PE_{\max}$ =1, then remove secret data S =0; if $PE_{\max}$ = 2, then remove secret data S=1 and restore $PE_{\max}$ to the maximum prediction error value -1; if $PE_{\max}$ > 2, then shift $PE_{\max}$ to the left by one unit and restore the maximum prediction error value -1. Similarly, the minimum prediction error value $PE_{\min}$, if $PE_{\min}$ =0, means not been; If $PE_{\min}$ = -1, the secret data S=0 is removed; if $PE_{\min} = -2$, the secret data S=1 is removed and $PE_{\min}$ is set to 1 to restore the minimum prediction error; if $PE_{\min}$ < -2, $PE_{\min}$ is shifted one unit to the right to restore the minimum prediction error $PE_{\min}$.

$$PE_{\max} = \begin{cases} 0 & if \quad PE'_{\max} = 0 \\ 0\, and\, S = 0 & if \quad PE'_{\max} = 1 \\ -1\, and\, S = 1 & if \quad PE'_{\max} = 2 \\ -1 & if \quad PE'_{\max} > 2 \end{cases} \tag{15}$$

$$PE_{\min} = \begin{cases} 0 & if \quad PE'_{\min} = 0 \\ 0\, and\, S = 0 & if \quad PE'_{\min} = -1 \\ 1\, and\, S = 1 & if \quad PE'_{\min} = -2 \\ 1 & if \quad PE'_{\min} < -2 \end{cases} \tag{16}$$

The prediction error values after a restoration are $PE_{\max}$ and $PE_{\min}$, the maximum pixel value $x'_{\pi(3)}$ and the minimum pixel value $x'_{\pi(1)}$ in the restored group of $A'_{\pi(n)}$ according to Equation (17), the middle pixel values were not adjusted.

$$\begin{cases} x_{\pi(3)} = x'_{\pi(3)} + PE_{\max} \\ x_{\pi(1)} = x'_{\pi(1)} + PE_{\min} \\ x_{\pi(2)} = x'_{\pi(2)} \end{cases} \tag{17}$$

Step 9: Restoring the group.

First, restore the pixel values in the group, then replace them in the corresponding original positions to obtain the restored group $A'_{\pi(n)}$. Finally, put the $A'_{\pi(n)}$ group back into the sub-block and take the next group and repeat steps 4 to 8 until the four groups of the sub-block are completed, to obtain the restored sub-block. The secret data is removed in the order A4 to A1 and embedded in the order A1 to A4, so the retrieved binary needs to be reordered to obtain the correct secret data.

Step 10: After the restored image block obtained is put back into the stego image, the next sub-block is taken and steps 3 to 9 are performed to obtain the original image I with secret data S.

3.3. SDPVO Embedding Example

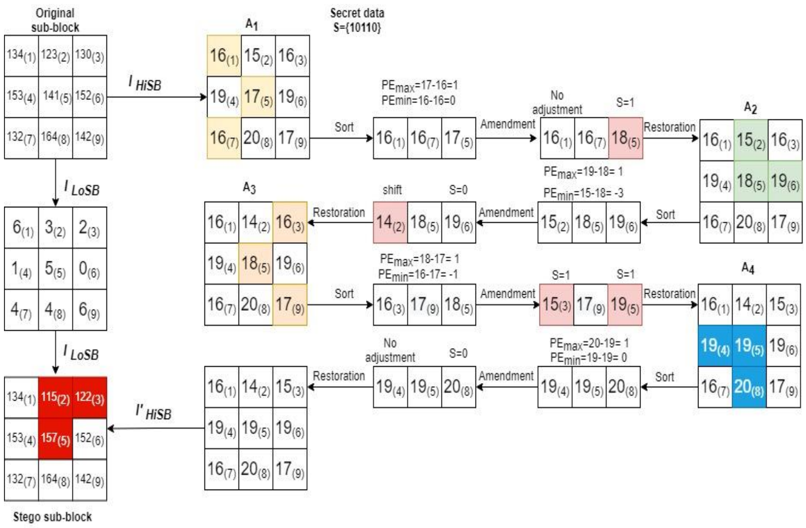

Figure 3: SDPVO method embedding example

First, 3×3 non-overlapping block of original data is first obtained, then the pixels are numbered $A=\{a_i \mid i=1,2,3,\cdots,9\}$, from left to right and top to bottom, and the pixel values are divided into four groups. The pixel values in group A_n are sorted from smallest to largest, and according to equation (8), determine whether they are over/under bit, and then determine the maximum and minimum prediction error values according to equation (9), and determine whether they can be embedded in the secret data or displaced. As shown in Fig. 3, the pixel values within the group A_n = (16(1), 17(5), 16(7)) are obtained, sorted to give, $A_{\pi(1)}$ = (16(1), 16(7), 17(5)), and the maximum prediction error is found to be 1 by equation (10), indicating that 1 bit of secret data can be embedded at the peak point; while the minimum prediction error is 0, indicating that no correction is made. After adjusting the pixel values according to equation (12), the sorting is restored and put back into the original image. Each block has to perform the embedding of the secret data four times, and the steps need to be performed sequentially according to the order of the groupings 1 to 4, where the pixel value a_5 is adjusted several times and even embeds the secret data several times. As can be seen from the example, although the sub-block is shifted and embedded several times, only 3 pixel values are adjusted to embed 5 bits of secret data, and the value of a_5 is adjusted twice.

3.4. SDPVO Extraction Example

After obtaining the stego image and additional information, the receiver cuts the stego image into 3×3 sized non-overlapping sub-blocks, and numbers and groups the pixel values within the blocks using raster scanning; after completing the grouping, it determines whether the group needs to be embedded with secret data according to LM, and then retrieves the secret data and restores the pixel values using equations (13) to (17). Since the embedding step is performed in the order of A1 to A4, and there is a chance that the secret data will be adjusted or embedded more than once, the removal of the secret data should be performed in reverse from group A4 to A1, and each completed group must be put back into the sub-block containing the secret data before the next one is removed to maintain the correctness of the pixel value a_5, the process is shown in Fig. 4.

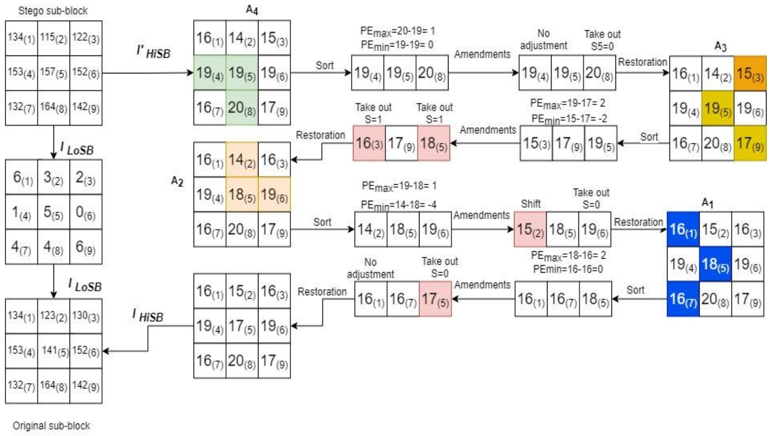

Figure 4: SDPVO method extraction example

4. Simulation Experiments and Analysis of Results

The scheme in this paper was simulated on the MATLAB R2018a platform. The test images were taken from the USC-SIPI database of eight 512×512 tiff distortion-free image format grayscale images, as shown in Fig. 5. The data S in the embedding process is a random sequence consisting of {0, 1}.

Figure 5: Standard images from the USC-SIPI database

In order to compare the performance of existing reversible data hiding schemes based on pixel value ordering with the scheme in this paper, experiments will use the performance of the following three parameters as evaluation criteria.

(1) Embedding Capacity (EC): The amount of secret data that can be embedded in an image.

(2) Peak Signal to Noise Ratio (PSNR): A criterion for determining the image quality of an image after embedding secret data, the higher the PSNR the better, representing a high imperceptibility of the stego image.

(3) Structural similarity index (SSIM): A measure of the degree of similarity between the original image and the distorted image, which is more in line with the human eye's judgement of image quality.

4.1. Comparison of Different Image Conditions at Maximum Embedding Volume

As shown in Table 1, the selected standard images maintain an average PSNR of 32.16db for the stego images at the maximum embedding and a mean SSIM of up to 0.9996, indicating that the level of distortion is still within the acceptable range for the human eye.

Table 1: SDPVO method embedding analysis table

Image	Total blocks	Maximum embedding volume	Embeddable blocks	Embed Ratio	PSNR (dB)	SSIM
Lena	28900	74593	25953	89.80%	32.7397	0.9994
Tiffany	28900	79863	26925	93.17%	32.0152	0.9999
Boat	28900	83522	27280	94.39%	31.7164	0.9996
Bridge	28900	75110	26659	92.25%	31.0013	0.9997
Pirate	28900	76200	26116	90.37%	32.1756	0.9996
Baboon	28900	79995	27277	94.38%	31.4801	0.9995
Peppers	28900	67205	24842	85.96%	32.4557	0.9999
Jetplane	28900	61187	22506	77.88%	33.7161	0.9999
Average value	**28900**	**74709**	**25944**	**89.77%**	**32.1625**	**0.9996**

In a 512×512 image, the image can be divided into 28,900 blocks according to the 3×3 chunking pattern, and the embedding amount varies from one image chunk to another. The Boat map has the highest embeddable block ratio at 94.39%. Analysis of Fig. 6 shows that the blocks with 2 and 3 bits embedded are the most numerous, accounting for over 50% of the blocks. Although, image sub-blocks can be embedded up to 8 bits, the percentage is extremely low.

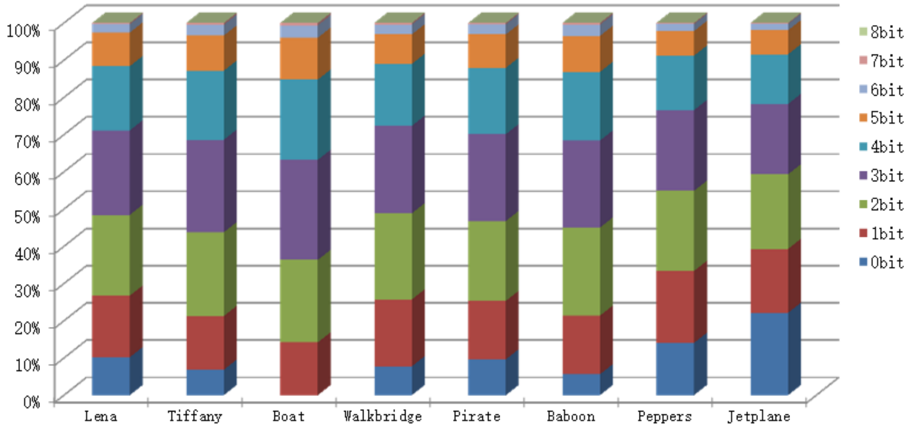

Figure 6: Block embedding comparison for each standard diagram

4.2. Comparison of the Maximum Embedding Volume with Other Methods

As shown in Fig. 7, the maximum embedding of six standard graphs of Lena, Jetplane, Baboon, Tiffany, Boat and Peppers in 512 × 512 images under different methods are compared, and the SDPVO method proposed in this paper basically outperforms PVO [10], IPVO [11], PVO-K [13] in terms of embedding per graph, PPVO [14] and Wang scholars' [15] methods, with the average embedding difference with PPVO being nearly more than a factor of one. In the experimental results it can be obtained that the best performance is obtained with image Baboon and Tiffany, indicating that the embedding of secret data using SDPVO is more suitable in images with high contrast pixel values and complex images.

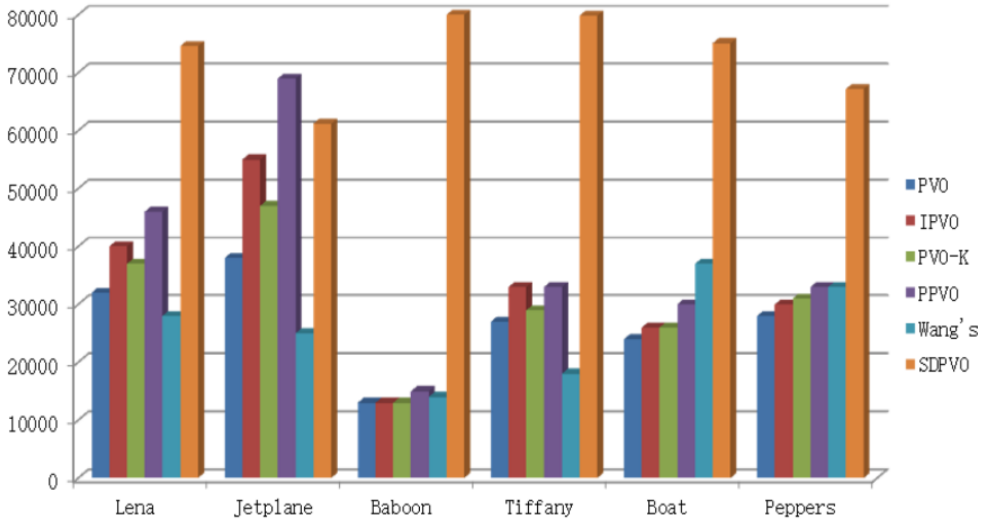

Figure 7: Comparison of the maximum embedding volume for each PVO method

4.3. Comparison of PSNR at Fixed Collections

In Table 2, the PSNR of PVO, IPVO, PVO-K, PPVO at 10,000 and 20,000 bits of embedded volume is compared with that of SDPVO in this study. The experimental results show that the SDPVO method has an average PSNR performance of 42.71 dB vs. 39.76 dB. Although the image quality performance of the proposed method is lower than the other methods, it still remains within the acceptable quality range for the human eye.

Table 2: Comparison of the PSNR of each method of PVO with an embedding capacity of 10,000 bits and 20,000 bits

	Mathod	**Lena**	**Jetplane**	**Peppers**	**Boat**
10000bits	*PVO*	60.3	62	58.9	58.1
	IPVO	60.5	62.9	59	58.3
	PVO-K	60.6	63.3	59.2	58.2
	PPVO	60.3	63.7	58.8	58.4
	SDPVO	**43.1**	**43.3**	**42.5**	**42.1**
20000bits	*PVO*	56.2	58.1	54.7	53.3
	IPVO	56.5	59	54.7	53.9
	PVO-K	56.6	59.3	54.9	53.7
	PPVO	56.7	59.9	55	54.2
	SDPVO	**39.7**	**40.7**	**39.7**	**38.9**

5. Conclusion

This paper proposes a high-capacity reversible data hiding method based on pixel value ordering, which embeds secret data in the quotient pixel values of grey-scale images to enhance their robustness, and incorporates the idea of chunking and adopts a speed dial embedding strategy to fully embed secret data based on all pixel values within the chunk, so as to increase the embedding volume. The experimental results show that a large amount of secret data can be embedded in a complex image using this method, and the degradation of PSNR tends to increase with the amount of embedded data, and the image quality is maintained within a tolerable impact range.

Acknowledgement

This paper is funded by Education Department of Hunan Province of China fund project "Research on reversible information hiding technology integrating Chinese residual theorem and pixel value sorting method" (21C0595), and Hunan Provincial Philosophy and Social Science fund project "Research on the Application of Artificial Intelligence Technology in Accurate Book Procurement in Universities under Smart Library" (21YBA179).

References

[1] Shi Y Q, Li X, Zhang X. (2016) Reversible Data Hiding: Advances in the Past Two Decades. IEEE Access, 4:3210-3237.

[2] Gouenou, Coatrieux, Clara. (2009) Reversible Watermarking for Knowledge Digest Embedding and Reliability Control in Medical Images. IEEE Trans, 13, 2, 158–165.

[3] Zhang Hongchao. (2020) Reversible Data Hiding Method for Enhancing Medical Image Contrast and Tampering Detection. Modern Information Technology, 42, 3, 55-58, 64.

[4] JIANG Fei, LI Jun, QIAN Zhen-xing. (2015) Reversible Data Hiding in Encrypted Medical Images with Improved Embedding Capacity. JOURNAL OF APPLIED SCIENCES, 33, 6, 644-654.

[5] Menendez O A, Feregrino U C. (2019) A Survey on Reversible Watermarking for Multimedia Content: A Robustness Overview. IEEE Access, 7, 132662-132681.

[6] J. Tian, (2003) Reversible Data Embedding Using a Difference Expansion. IEEE Trans. Circuits Syst. Video Technol, 13: 890–896.

[7] Z. Ni, Y. Q. Shi, N. Ansari, W. Su. (2006) Reversible Data Hiding, IEEE Trans. Circuits Syst. Video Technol. 16: 354–362.

[8] M. Thodi, J. J. Rodr ǵuez. (2007) Expansion Embedding Techniques for Reversible Watermarking. IEEE Trans. Image Processing. 16: 721–730.

[9] B. Ou, X. Li, Y. Zhao, R. Ni, Y. Shi. (2013) Pairwise Prediction-error Expansion for Efficient Reversible Data Hiding. IEEE Trans. Image Process, 22: 5010–5021.

[10] X. Li, J. Li, B. Li, B. Yang. (2013) High-fidelity Reversible Data Hiding Scheme Based on Pixel-Value-Ordering and Prediction-error Expansion. Signal Process. 93, 198–205.

[11] F. Peng, X. Li, B. Yang. (2014) Improved PVO-based Reversible Data Hiding. Digital Signal Processing, 25: 255–265.

[12] Bo. Ou, X. Li, Y. Zhao, R. Ni. (2014) Reversible Data Hiding Using Invariant Pixel-valueordering and Prediction-error Expansion. Signal Process: Image Commun, 29, 760–772.

[13] X. Wang, J. Ding, Q. Pei. (2015) A Novel Reversible Image Data Hiding Scheme Based on Pixel Value Ordering and Dynamic Pixel Block Partition. Information sciences, 310, 16-35.

[14] X. Qu, H. J. Kim. (2015) Pixel-based Pixel Value Ordering Predictor for High-fidelity Reversible Data Hiding. Signal Processing, 111, 249-260.

[15] W. Wang, J. Ye, T. Wang, W. Wang. (2017) Reversible Data Hiding Scheme Based on Significant-bit-difference Expansion. IET Image Process, 11, 1002-1014.