Steganography Using Multiple-Base Notational System and Human Vision Sensitivity

Xinpeng Zhang and Shuozhong Wang

Abstract—**This letter proposes a novel steganographic scheme that employs human vision sensitivity to hide a large amount of secret bits into a still image with a high imperceptibility. In this method, data to be embedded are converted into a series of symbols in a notation system with multiple bases. The specific bases used are determined by the degree of local variation of the pixel magnitudes in the host image so that pixels in busy areas can potentially carry more hidden data. Experimental results are given to show the advantage of this adaptive technique.**

Index Terms—**Human vision system (HVS), information hiding, steganography.**

I. Introduction

As an application of information hiding, steganography aims to send secret messages under the cover of a carrier signal [1], [2]. A steganographic technique should generally possess two important properties: good visual/statistical imperceptibility and a sufficient payload. The first is essential for the security of hidden communication and the second ensures that a large quantity of data can be conveyed.

While many digital watermarking techniques use characteristics of human vision sensitivity [3], (HVS)-based steganographic techniques are also developed to embed a large amount of secret bits into a still image with high imperceptibility, in which more data are inserted into busy areas. For example, in the bit-plane complexity segmentation (BPCS) method [4], blocks with a bit plane of high complexity, defined as the number of transitions from 1 to 0 or from 0 to 1, are replaced with the secret data. Another technique, termed the pixel-value differencing (PVD) method [5], segments the cover image into nonoverlapping blocks containing two connecting pixels and modifies the pixel difference in each block (pair) for data embedding. A larger difference in the original pixel values allows a greater modification.

In these methods, data embedding is performed in a block-wise fashion. In other words, the numbers of bits of secret data carried by individual pixels in the same block are made identical. In fact, however, even two adjacent pixels may have different tolerance for steganographic modifications in terms of visual and statistical detectability. This can be exploited to accommodate more secret data without introducing additional detectable traces. This letter proposes an alternative steganographic method in which the data to be hidden are re-expressed based on a multiple-base notational system and then embedded into pixels according to the different degree of pixel value variation in the immediate neighborhood. Embedding strength is varied over the entire host image on a pixel-by-pixel basis, allowing more secret data to be carried in busy areas. On the receiving side, the original image is not needed for recovering the embedded message.

II. Notational System With Multiple Bases

In this section, we introduce a notational system with multiple bases, which can be used to re-express a secret message to be hidden. It is well known that the decimal system is convenient in daily life and the binary system in computer operations. The bases in these systems, 10 and 2, respectively, are constant throughout. In most cases, the secret message is a binary stream, and the amount of information contained in each symbol is exactly one bit. In order to embed more data into busy areas, the message can be expressed as an integer number using a variable base system. In other words, the message is converted into a series of symbols with different information-carrying capability due to different bases used. The greater the base, the more information is contained in the corresponding symbol.

Assuming that an integer number is expressed in a variable base system

$$\begin{aligned} &x = (d_{n-1}d_{n-2}\ldots d_2 d_1 d_0)_{b_{n-1}b_{n-2}\ldots b_2 b_1 b_0} \\ &0 \le d_i < b_i \; (i = 0, 1, \ldots, n-1) \end{aligned} \tag{1}$$

where $b_0, b_1, b_2, \ldots, b_{n-2}$ and b_{n-1} denote the different bases corresponding, respectively, to the symbols $d_0, d_1, d_2, \ldots, d_{n-2}$ and d_{n-1}, the decimal value of x can be calculated as follows:

$$x = d_0 + \sum_{i=1}^{n-1} \left(d_i \cdot \Pi_{j=0}^{i-1} b_j\right). \tag{2}$$

If the decimal value of x and the bases, $b_0, b_1, b_2, \ldots, b_{n-2}$, and b_{n-1}, are given, one can convert x into the multiple-base notational system

$$d_0 = \bmod (x, b_0) \tag{3}$$

$$d_k = \bmod \left\{ \frac{1}{\prod_{j=0}^{k-1} b_j} \left[x - d_0 - \sum_{i=1}^{k-1} \left(d_i \cdot \Pi_{j=0}^{i-1} b_j\right) \right], b_k \right\}, \quad k \ge 1. \tag{4}$$

Manuscript received March 16, 2004; revised May 20, 2004. This work was supported by the National Natural Science Foundation of China (no. 60372090). The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Mark Hasegawa-Johnson.

The authors are with the School of Communication and Information Engineering, Shanghai University, Shanghai 200072, China (e-mail: xzhang@staff.shu.edu.cn; shuowang@staff.shu.edu.cn).

Digital Object Identifier 10.1109/LSP.2004.838214

This way, $d_0, d_1, d_2, \ldots, d_{n-2}$, and d_{n-1} can be obtained successively. For example, $49 = (1301)_{3532}$ and $158 = (3142)_{6254}$. With a sequence of bits in hand, we can first interpret it as a positive integer using (2), and then re-express it in a multiple-base notational system by using (3) and (4) with a set of given bases.

III. Steganographic Scheme

In the proposed scheme, a secret message is embedded into the cover image by modifying pixel values in a particular order derived from a key. A rule of thumb is that the more the variation of pixel-values in the vicinity of a pixel, the more the pixel can tolerate steganographic modification, allowing a greater change to be introduced. As such, we let each pixel carry one symbol of the secret message in a multiple-base notational system, with the corresponding base being proportional to the degree of variation in the pixel's immediate neighborhood. Thus, pixels in busy areas carry more information and statistically undergo more modification than those in smooth areas. On the extraction side, a receiver can retrieve all the symbols and bases from the stego-image to recover the embedded message as will be explained in the following.

A secret key, which is shared by the message hider and the receiver, determines a specific path of pseudorandom walk over the pixels. This is achieved in the following manner.

1) Let S_0 be a set made up of pixels in the top-most row and the left-most column, and S_1 be a set of the remaining pixels. Assume that H is a sequence of pixels that is initially empty.
2) Select a pixel $p(i,j)$ from S_1 that meets the condition $p(i-1,j)$, $p(i-1,j-1)$ and $p(i,j-1) \in S_0$, and append it to the end of H. If more than one pixel meet the condition, the element to be appended to H is decided according to the key. Update S_0 and S_1 by adding $p(i,j)$ to S_0 and removing $p(i,j)$ from S_1, i.e., $S_0 \leftarrow S_0 + \{p(i,j)\}$ and $S_1 \leftarrow S_1 - \{p(i,j)\}$.
3) Iteratively repeat Step 2 until S_1 becomes empty.

Thus the final sequence H contains all the pixels $p(i,j)$ $(2 \le i \le M, 2 \le j \le N)$ to be modified, where M and N are the numbers of rows and columns of the cover image, and the secret data will be embedded into all pixels except the top-most row and the left-most column following a path indicated by H derived from the key. The first element in H must be $p(2,2)$ and the last $p(M,N)$. Any pixel in the image can only be processed after its left, top and top-left neighbors have been processed previously. A simple example of such a walk over the pixels in a host image sized 4×8 is shown in Fig. 1. The sequence H in this case is therefore $\{p(2,2)p(2,3)p(3,2)p(2,4)p(3,3)\ldots p(4,8)\}$.

The detailed embedding procedure is designed as follows:

1) Pixel values in the original and stego images are denoted respectively as $p(i,j)$ and $p'(i,j)$. In this method, pixels in the top-most row and the left-most column are not used for data embedding. In other words,

$$p'(i,j) = p(i,j), \quad i = 1 \text{ or } j = 1. \tag{5}$$

	1	2	4	6	7	11	16
	3	5	9	12	15	18	19
	8	10	13	14	17	20	21

Fig. 1. An example of pixel order for data insertion.

2) Divide the secret bit stream into segments, each comprising l bits (for example, $l = 32$).
3) Convert each binary segment to a positive integer x and set the initial value of a parameter $u = 1$, which is used to determine the total number of symbols needed for representing this secret segment in a multiple-base notational system.
4) Embed the secret message sequentially into the pixels directed by H. Note that embedding in a pixel (i,j) must be done after pixels $(i-1,j)$, $(i-1,j-1)$ and $(i,j-1)$ so that modification to $p(i,j)$ does not destroy the data that have previously been inserted. Denote the standard deviation of the three stego-values $p'(i-1,j)$, $p'(i-1,j-1)$ and $p'(i,j-1)$ as $\sigma(i,j)$ and calculate the corresponding base

$$b(i,j) = \min\left(\left\lceil \frac{\sigma(i,j)}{\Delta} \right\rceil, 16\right) \tag{6}$$

where Δ is a constant and the operator $\lceil \bullet \rceil$ takes the nearest integer toward infinity. A small Δ leads to a large b so that more data can be embedded with greater distortion to the cover, and vice versa. The base $b(i,j)$ is proportional to $\sigma(i,j)$ unless the base is clipped to 16.
5) If $b(i,j) \le 1$, pixel (i,j) is not eligible to carry any data. Skip the pixel and go back to Step 4. Otherwise, compute the corresponding digit in the multiple-base notational system

$$d(i,j) = \mathrm{mod}[x, b(i,j)]. \tag{7}$$

6) Modify the value of pixel (i,j) to embed $d(i,j)$

$$p'(i,j) = \mathop{\arg\min}_{v \in [0,255],\, \mathrm{mod}[v,b(i,j)] = d(i,j)} |v - p(i,j)|. \tag{8}$$

Clearly, $p'(i,j)$ is closest to the original $p(i,j)$ among all values having a residue modulo $b(i,j)$ of $d(i,j)$. In fact, $p'(i,j)$ can only take two values, i.e.,

$$p'_1 = \left\lfloor \frac{p(i,j) - d(i,j)}{b(i,j)} \right\rfloor \cdot b(i,j) + d(i,j) \tag{9}$$

or

$$p'_2 = \left\lfloor \frac{p(i,j) - d(i,j)}{b(i,j)} + 1 \right\rfloor \cdot b(i,j) + d(i,j) \tag{10}$$

where the operator $\lfloor \bullet \rfloor$ takes the nearest integer toward minus infinity. Therefore, the stego-value is simply chosen between p'_1 and p'_2.

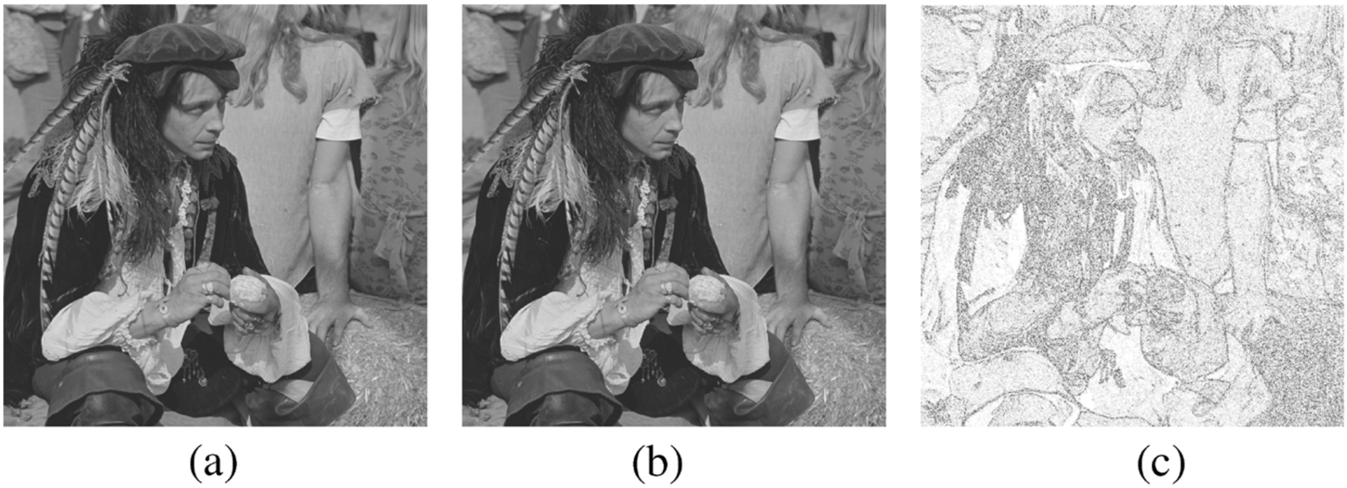


Fig. 2. (a) Original image. (b) Stego-image with $\Delta = 0.75$. (c) Enhanced error image.

TABLE I
PERFORMANCE OF THE PROPOSED MBNS STEGANOGRAPHY WITH DIFFERENT Δ VALUES (COVER IMAGE: "MAN")

Δ	1.50	1.25	1.00	0.75	0.50
Length of embedded payload (bits)	4.2×10^5	4.6×10^5	5.2×10^5	6.1×10^5	7.4×10^5
Embedding rate	0.20	0.22	0.25	0.29	0.35
PSNR (dB)	43.5	42.5	41.3	39.9	38.1
Watson metric	0.0290	0.0328	0.0383	0.0460	0.0591
Q	0.9994	0.9992	0.9990	0.9986	0.9978

TABLE II
VALUES OF THE THREE QUALITY METRICS AVERAGED OVER 100 TEST IMAGES WITH DIFFERENT EMBEDDING RATES

Quality metrics	PSNR			Watson metric			*Q*		
Embedding rate	0.20	0.25	0.30	0.20	0.25	0.30	0.20	0.25	0.30
Average value	43.4	41.2	39.9	0.029	0.039	0.047	0.9994	0.9990	0.9986

7) Update the parameter u

$$u \leftarrow u \cdot b(i,j). \tag{11}$$

If $u < 2^l$, meaning that the secret segment has not been completely represented, go to Step 4 after updating the value of x

$$x \leftarrow \frac{x - d(i,j)}{b(i,j)}. \tag{12}$$

Otherwise, go to Step 3 to embed another binary segment, until all segments of the secret bit stream are embedded. In this way, each binary segment is embedded into several pixels in the cover image.

On the extraction side, the secret key and the system parameter Δ are used to recover the embedded message. After obtaining the sequence H from the key, the bases $b(i,j)$ can be orderly calculated from the stego-image using (6), and the value of $d(i,j)$ is obtained when $b(i,j)$ is larger than 1

$$d(i,j) = \mathrm{mod}[p'(i,j), b(i,j)]. \tag{13}$$

Take a series of $b(i,j)s$ larger than 1 consecutively in an order indicated by H, denoted $b(t)$, $t = 1, 2, \ldots, T$, where $T \leq (M-1)(N-1)$. Rewrite $\{b(1), b(2), \ldots, b(T)\}$ as $\{b(1), b(2), \ldots, b(T_1), b(T_1+1), b(T_1+2), \ldots, b(T_2), \ldots, b(T_k+1), b(T_k+2), \ldots, b(T_{k+1}), \ldots\}$ under the following condition:

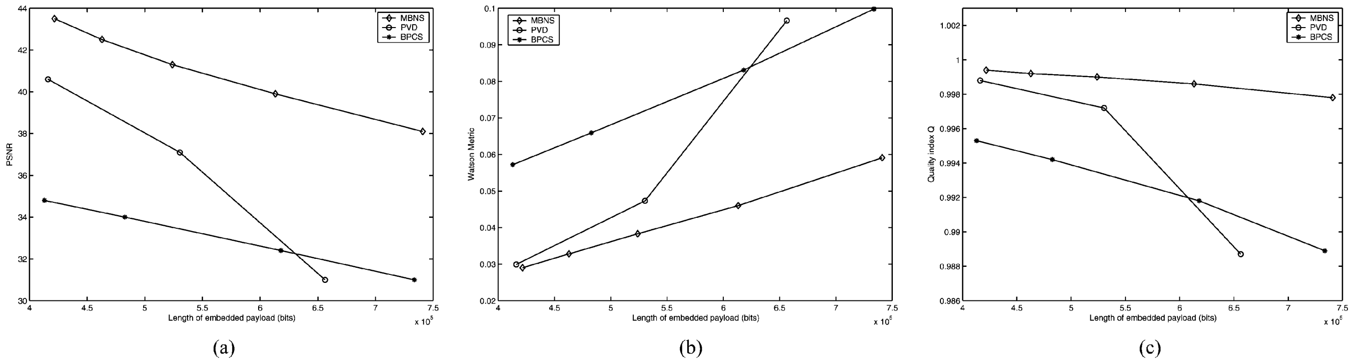
$$\prod_{t=T_k+1}^{T_{k+1}-1} b(t) < 2^l$$
$$\prod_{t=T_k+1}^{T_{k+1}} b(t) \geq 2^l, \quad k = 0, 1, 2, \ldots; \ T_0 = 0. \tag{14}$$

Then, every embedded binary segment can be extracted from the corresponding symbols $d(T_k+1), d(T_k+2), \ldots, d(T_{k+1})$.

IV. EXPERIMENTAL RESULTS

Using a 512 × 512 8-bit grayscale image "Man" as the cover and choosing $\Delta = 0.75$, a total of 6.1 $\times 10^5$ secret bits were embedded. In this case, the embedding rate, which is a ratio between the bit numbers of embedded data and the host image, is 0.29. The original and stego images are shown in Fig. 2. Also shown is the error image, which has been enhanced by a 20-time gray-level stretch for the purpose of display. Since the modifications were mainly in texture areas and on edges, high imperceptibility was achieved.

Three quality metrics were used to measure the distortion induced by data embedding: PSNR, the Watson metric, and a universal quality index Q. While PSNR simply indicates the en-



(a)

(b)

(c)

Fig. 3. Performance comparison between PVD, BPCS, and MBNS techniques at different payload, using "Man" as the cover image. (a) PSNR. (b) Watson metric. (c) Quality index Q. It is shown that the MBNS technique provides better invisibility with higher PSNR, lower Watson metric and higher Q.

TABLE III

PERFORMANCE COMPARISON OF THE THREE METHODS WITH THE SIZE OF ALL COVER IMAGES USED BEING 512 × 512, AND THE PAYLOAD 4.4 $\times 10^5$ BITS (EMBEDDING RATE $= 0.21$), INDICATING THAT THE MBNS METHOD PROVIDES THE HIGHEST PSNR, LOWEST WATSON METRIC AND HIGHEST Q IN ALL CASES

Cover	PSNR (dB)			Watson metric			Q		
Image	PVD	BPCS	MBNS	PVD	BPCS	MBNS	PVD	BPCS	MBNS
Lena	41.7	34.6	44.3	0.0299	0.0592	0.0292	0.9989	0.9947	0.9994
Baboon	36.2	26.8	41.4	0.0447	0.1266	0.0275	0.9968	0.9718	0.9994
Bridge	37.8	28.4	42.5	0.0404	0.1620	0.0315	0.9982	0.9778	0.9995
Peppers	41.2	33.5	45.0	0.0317	0.0727	0.0296	0.9990	0.9941	0.9996

ergy of distortion caused by data hiding, the Watson metric is designed by using characteristics of the human visual system and measures the total perceptual error, which is DCT-based and takes into account three factors: contrast sensitivity, luminance masking and contrast masking [6]. [7] provides a source code for calculating the Watson metric. Additionally, the quality index Q works in the spatial domain, as a combination of correlation loss, luminance distortion, and contrast distortion [8].

With the cover image "Man", the length of payload, embedding rate, PSNR, Watson metric, and Q corresponding to different Δ are listed in Table I. Table II shows values of the three quality metrics averaged over 100 test images with different embedding rates. The 100 host images containing landscape and people were captured with a digital camera. Higher PSNR, lower Watson metric, or higher Q means better imperceptibility. In Fig. 3, performance comparison between the three methods: BPCS, PVD and the proposed MBNS embedding is presented showing that the MBNS technique provides better invisibility measured by all the three metrics. Experiments on other images give similar results. Some results are listed in Table III, with a length of embedded bit-stream being 4.4 $\times 10^5$ in all cases.

V. CONCLUSION

In the proposed steganographic scheme, the amount of information carried by individual pixels is adapted to the gray value variation in the immediate neighborhood, realized by using a novel multiple-base notational system. As more data are embedded in busy areas and on edges that can tolerate more changes, the method provides a good imperceptibility with a large quantity of embedded data. Simulation experiments show that the proposed MBNS method has a better performance than the bit-plane complexity segmentation and pixel-value differencing techniques.

Using this method, smooth regions are unchanged, and variation in busy areas is slightly increased in such a way that the modification is approximately proportional to the local fluctuation, in the hope that statistical analyses are difficult. On the other hand, the order of pixels used for data hiding is derived from a secret key. Therefore, any third party will be unable to detect or extract the embedded message. However, the method is not designed to resist lossy compression or active attacks as any such operations will prevent correct extraction of the bases and symbols.

REFERENCES

[1] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Information hiding—A survey," *Proc. IEEE*, vol. 87, pp. 1062–1078, 1999.

[2] H. Wang and S. Wang, "Cyber warfare—Steganography vs. Steganalysis," *Commun. ACM*, vol. 47, no. 10, pp. 76–82, 2004.

[3] M. Kutter and S. Winkler, "A vision-based masking model for spread-spectrum image watermarking," *IEEE Trans. Image Processing*, vol. 11, pp. 16–25, Jan. 2002.

[4] H. Noda, J. Spaulding, M. N. Shirazi, and E. Kawaguchi, "Application of bit-plane decomposition steganography to JPEG2000 encoded images," *IEEE Signal Processing Lett.*, vol. 9, no. 12, pp. 410–413, Dec. 2002.

[5] D.-C. Wu and W.-H. Tsai, "A steganographic method for images by pixel-value differencing," *Pattern Recognit. Lett.*, vol. 24, pp. 1613–1626, 2003.

[6] A. Mayache, T. Eude, and H. Cherifi, "A comparison of image quality models and metrics based on human visual sensitivity," in *Proc. Int. Conf. Image Processing (ICIP'98)*, vol. 3, Chicago, IL, Oct. 1998, pp. 409–413.

[7] [Online]. Available: http://watermarking.unige.ch/Checkmark/

[8] Z. Wang and A. C. Bovik, "A universal image quality index," *IEEE Signal Processing Lett.*, vol. 9, no. 3, pp. 81–84, Mar. 2002.