Multibit Embedding Algorithm for Steganography of Palette-Based Images

Shoko Imaizumi and Kei Ozawa

Graduate School of Advanced Integration Science, Chiba University
1–33 Yayoicho, Inage-ku, Chiba-shi, Chiba 263-8522, Japan
imaizumi@chiba-u.jp

Abstract. We propose a high-capacity data hiding scheme for palette-based images that does not seriously degrade the image quality in this paper. The proposed scheme can embed a multiple-bit message within the unit of a pixel matrix by using Euclidean distance, while some conventional schemes can embed only a one-bit message per pixel. The stego-images created by using our scheme offer a better quality compared to those by the conventional scheme. Moreover, we have obtained these results with low implementation cost. The experimental results show that the proposed scheme is efficient.

Keywords: Steganography, data hiding, palette-based image, Euclidean distance, pixel matrix.

1 Introduction

Steganography [1,2] is a data hiding scheme that communicates a secret messages by imperceptibly embedding them into a cover data, such as an image or audio, etc. When the cover data is an image, the image is called a cover image. A cover image that possesses a secret message actually forms a stego-image. The stego-image can be transmitted through open channels without suspicion since the secret message is generally embedded into the cover image without creating noticeable artifacts. The authorized recipient can extract the embedded message from the stego-image, while others are unaware of the existence of the message behind the stego-image.

Palette-based images, which generally use no more than 256 palette entries (simply called entries hereafter), are frequently used as cover images. This is because palette-based images can be conveniently distributed and found through communication channels, even if the channel is quite narrow. Entries that compose palette-based images are stored in the palette. Each pixel in a palette-based image possesses an index that points to the entry.

There are two types in steganographic schemes for palette-based images embedding a message by controlling entries in the palette. One of them changes the colors of the entries in order to embed a message with only slight degradation [3–6]. Niimi's scheme [3], for instance, embeds a message based on the Green values of the colors for the target pixels. The other retains the colors of

R. Klette, M. Rivera, and S. Satoh (Eds.): PSIVT 2013, LNCS 8333, pp. 99–110, 2014.

the entries and may reorder the entries in the palette [7–13]. EZ stego scheme [7] sorts the palette by luminance and embeds message bits into the LSBs of the indices. Fridrich [8] presented a steganographic scheme for hiding message bits into the parity bit of close colors. The former schemes [3–6] create some new entries in the palette, and removes the same number of entries as the new entries. Thus, the schemes should increase the computational cost of the calculation for adding and removing entries. Our scheme adopts the latter schemes [7–13].

We propose a high-capacity steganographic scheme for palette-based images in this paper. The proposed scheme can embed a multiple-bit message within the unit of a pixel matrix, while some conventional schemes can embed only a one-bit message per pixel. This scheme forms better quality stego-images using simple implementation than the conventional scheme. A performance analysis validated our scheme.

2 Related Work

We review three conventional steganographic schemes for palette-based images [7,8,13] in this section.

2.1 EZ Stego Scheme [7]

EZ stego scheme is one of the most famous steganographic schemes for embedding a one-bit message within the unit of a pixel by changing its entry. Assume that there are X entries in the palette and that the length of the embedded message is $\varGamma$ bits. This scheme embeds a secret message using the following steps.

Step 1 Sort entries E_i $(i = 0, 1, \dots, X)$ in the palette in ascending order according to the luminance L_i, which is represented as

$$L_i = 0.299 r_i + 0.587 g_i + 0.144 b_i, \tag{1}$$

where r_i, g_i, and b_i are the red, green, and blue values of an entry E_i, respectively.

Step 2 Select the γ-th target pixel $(\gamma = 1, 2, \dots, \varGamma)$ containing index i of entry E_i from the cover image.

Step 3 Find index i' of entry E_i in the reordered palette.

Step 4 Replace the LSB of index i' with the γ-th one-bit message to be embedded, and then obtain index i'_{n} of the neighboring entity $E_{i_{\mathrm{n}}}$. Note that if the one-bit message is equal to the LSB of index i', leave index i' unchanged and return to **Step 2**.

Step 5 Find index i_{n} of entry $E_{i_{\mathrm{n}}}$ in the original palette.

Step 6 Replace index i in the target pixel with index i_{n}.

Step 7 Repeat **Steps 2** to **6** until $\gamma = \varGamma$.

The recipient should only collect the LSBs by using the location map when extracting the message. However, EZ stego may occasionally replace an entry in a pixel with a totally different entry, because it reorders the palette entries according to the luminance given by Eq. (1).

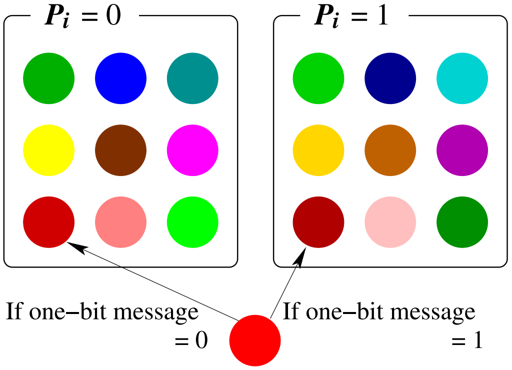


Fig. 1. Embedding one-bit message bit into parity in Fridrich's scheme [8]

2.2 Fridrich's Scheme [8]

To solve the problem of EZ stego scheme, Fridrich proposed another scheme that embeds a Γ-bit message into the parities of close entries. The embedding procedure is described as follows.

Step 1 Calculate parities P_i for all entries $E_i\,(r_i, g_i, b_i)$ in the palette, which is defined as

$$P_i = (r_i + g_i + b_i) \mod 2, \quad i = 0, 1, \ldots, X. \tag{2}$$

Step 2 Select the γ-th target pixel ($\gamma = 1, 2, \ldots, \Gamma$) with entry E_i, whose parity is P_i, in the cover image. Note that if the γ-th one-bit message M_γ to be embedded is equal to P_i, leave the γ-th pixel unchanged and repeat **Step 2**.

Step 3 Find the closest entry E_{i_c} to the entry E_i of the target pixel in the set of entries whose parities $p\,(p = 0$ or $1)$ are equal to M_γ, by using the Euclidean distance D_{i_0,i_1}, as shown in Fig. 1. The D_{i_0,i_1} between two entries $E_{i_0}\,(r_{i_0}, g_{i_0}, b_{i_0})$ and $E_{i_1}\,(r_{i_1}, g_{i_1}, b_{i_1})$ is given by

$$D_{i_0,i_1} = \sqrt{(\Delta r_{i_0,i_1})^2 + (\Delta g_{i_0,i_1})^2 + (\Delta b_{i_0,i_1})^2}. \tag{3}$$

Step 4 Replace entry E_i of the target pixel with the closest entry E_{i_c}.

Step 5 Repeat **Steps 2** to **4** until $\gamma = \Gamma$.

The message can be easily recovered by collecting the parity bits for the entries of the target pixels according to the location map. Although this scheme can avoid replacing the colors of the target pixels with completely different ones, it can only embed a one-bit message per pixel.

(a) Original (b) Original (partially magnified) (c) Stego (partially magnified)

Fig. 2. Example of message embedding using Tanaka's scheme [13]

2.3 Tanaka's Scheme [13]

Tanaka presented a high-capacity steganography that is based on Fridrich's scheme. This scheme can embed a k-bit message per pixel without increasing the degradation of the image quality compared to that for Fridrich's scheme. The main contribution of this scheme is assigning a k-bit parity to each of X entries. Parity P_i is assigned to entry E_i using the following steps.

Step 1 Find initial entry E_{i_0}, that is

$$E_{i_0} = \arg\min_{i} \left(256^2 r_i + 256^1 g_i + 256^0 b_i\right). \tag{4}$$

Assign 0 to parity P_{i_0} for entry E_{i_0}.

Step 2 Find entry E_{i_x}, that is

$$E_{i_x} = \arg\min_{i \in \alpha} D_{i_{x-1}, i}, \tag{5}$$

where D is defined by Eq. (3). Note that α is a set of i in which each entry E_i has not been assigned a parity yet.

Step 3 Set parity $P_{i_x} = P_{i_{x-1}} + 1$.

Step 4 Repeat **Steps 2** and **3** until $x = 2^k - 1$.

Step 5 Find entry E_{i_x}, which is given by Eq. (5).

Step 6 Find the 2^k closest entries $E_{i_c^{(p)}}$ to E_{i_x} in the sets $\alpha^{(p)}$ on each parity p $\left(p = 0, 1, \ldots, 2^k - 1\right)$, which are given by

$$E_{i_c^{(p)}} = \arg\min_{i^{(p)} \in \alpha^{(p)}} D_{i_x, i^{(p)}}. \tag{6}$$

Step 7 Set parity P_{i_x} as

$$P_{i_x} = \arg\max_{p} D_{i_x, i_c^{(p)}}. \tag{7}$$

Step 8 Repeat **Steps 5** to **7** until $x = X - 1$.

The embedding procedure is the same as that for Fridrich's scheme. When a message is embedded into an cover image as shown in Fig. 2(a), part of the image, such as that shown in Fig. 2(b), for instance, is degraded like that shown in Fig 2(c). If the entry of the target pixel has no close entries in the palette, the target pixel is changed to a totally different color, and the stego-image is seriously damaged.

3 Proposed Scheme

In this section, we present a high-capacity steganographic scheme for palette-based images that has less degradation than the conventional scheme. Our scheme is composed of simple operations. This scheme is based on embedding the message into the parities. The new approach uses 3 × 3 pixel matrices to inhibit the serious degradation of a stego-image. Note that the arbitrary size of the matrix can be adopted to the proposed scheme.

3.1 Reordering Palette Entries

Assume that we embed a k-bit message, where $k = 1, 2,$ or 3, into each 3 × 3 pixel matrix. First, we reorder all X entries in the palette for a cover image using following steps.

Step 1 Find initial entry E_{i_0} in the original palette using the following equation.

$$E_{i_0} = \arg\min_{i} \left(256^2 r_i + 256^1 g_i + 256^0 b_i\right). \tag{8}$$

Step 2 Set index $I_0 = 0$ to E_{i_0}.
Step 3 Find entry E_{i_x}, that is

$$E_{i_x} = \arg\min_{i \in \alpha} D_{i_{x-1}, i}, \tag{9}$$

where D is defined by Eq. (3). Note that α is a set of i, where each entry E_i has not been assigned an index yet.

Table 1. Reordered palette in proposed scheme

Index I_x	Entry E_{i_x}
0	$\arg\min_i \left(256^2 r_i + 256 g_i + b_i\right)$
1	$\arg\min_{i\in\alpha} D_{i,0}$
2	$\arg\min_{i\in\alpha} D_{i,1}$
...	...
$X-2$	$\arg\min_{i\in\alpha} D_{i,X-3}$
$X-1$	$E_i \ (i \in \alpha)$

$t_{0(\beta)}$	$t_{1(\beta)}$	$t_{2(\beta)}$
$t_{3(\beta)}$	$t_{4(\beta)}$	$t_{5(\beta)}$
$t_{6(\beta)}$	$t_{7(\beta)}$	$t_{8(\beta)}$

Fig. 3. Pixels $t_{j(\beta)}$ in 3×3 matrix

Step 4 Set index $I_x = x$ to E_{i_x}.
Step 5 Repeat **Steps 2** and **4** until $x = X - 1$.

The neighboring entries possess similar colors to each other when using the above mentioned steps. The reordered palette is formed in the way shown in Table 1.

3.2 Embedding Procedure

We divide embedded message M into B of k-bit blocks, whose values are represented as M_β ($M_\beta = 0, 1, \ldots, 2^k - 1$ and $\beta = 1, 2, \ldots, B$). The k-bit message M_β is embedded into the pixels $t_{j(\beta)}$ $(j = 0, 1, \ldots, 8)$ in the β-th 3×3 matrix, which are shown in Fig. 3. The embedding procedure is as follows.

Step 1 Select the β-th target matrix with nine pixels $t_{j(\beta)}$ $(j = 0, 1, \ldots, 8)$.
Step 2 Take summation S_β over nine indices $I_{j(\beta)}$ of $t_{j(\beta)}$.

$$S_\beta = \sum_{j=0}^{8} I_{j(\beta)}. \tag{10}$$

Step 3 Calculate parity P_β for the β-th pixel matrix given as

$$P_\beta = S_\beta \mod 2^k, \tag{11}$$

where $k = 1, 2$, or 3.
Step 4 Calculate minimal error R_β between M_β and P_β, as shown in Fig. 4.

$$R_\beta = \begin{cases} \min\left(P_\beta - M_\beta, M_\beta - P_\beta + 2^k\right), & \text{if } M_\beta < P_\beta \\ \min\left(M_\beta - P_\beta, P_\beta - M_\beta + 2^k\right), & \text{if } M_\beta > P_\beta. \end{cases} \tag{12}$$

If $R_\beta = 0$, i.e., $M_\beta = P_\beta$, leave the β-th matrix unchanged and return to **Step 1**.
Step 5 Extract the R_β of pixels $t_{j(\beta)}$ in ascending order corresponding to the Euclidean distance $D_{j(\beta)}$ between entry $E_{j(\beta)}$ $\left(E_{j(\beta)} = E_{i_x}\right)$ for pixel $t_{j(\beta)}$ and entry $E_{i_{x-1}}$ or $E_{i_{x+1}}$, which is given as

$$D_{j(\beta)} = \begin{cases} D_{i_x, i_{x-1}}, & \text{if } R_\beta < 0 \\ D_{i_x, i_{x+1}}, & \text{if } R_\beta > 0. \end{cases} \tag{13}$$

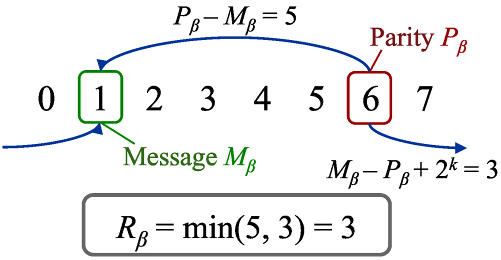


Fig. 4. Decision of minimal error R_β ($k = 3$)

Step 6 Replace the R_β of indices $I_{j(\beta)}$ $\left(I_{j(\beta)} = I_x\right)$ of target pixels $t_{j(\beta)}$ to I_{x_i-1} or I_{x_i+1}.

Step 7 Repeat **Steps 1** to **6** until $\beta = B$.

3.3 Extracting Procedure

The algorithm for extracting the embedded message is quite simple. After reordering the palette entries according to Section 3.1, the extracting procedure works in the following way.

Step 1 Select the β-th target matrix with nine pixels $t_{j(\beta)}$ ($j = 0, 1, \ldots, 8$) according to the location map.

Step 2 Take summation S_β over nine indices $I_{j(\beta)}$ of $t_{j(\beta)}$.

$$S_\beta = \sum_{j=0}^{8} I_{j(\beta)}. \tag{14}$$

Step 3 Calculate parity P_β for the β-th pixel matrix given as

$$P_\beta = S_\beta \mod 2^k, \tag{15}$$

where $k = 1, 2$, or 3.

Step 4 Assign P_β to M_β.

Step 5 Repeat **Steps 1** to **Step 4** until $\beta = B$.

Step 6 Concatenate the B of the k-bit messages M_β in ascending order of β, and read the extracted message M.

Note that the recipient has to receive the location map in order to extract the message.

4 Experimental Results

We present the experimental results of the proposed scheme and compare them with those of Tanaka's scheme [13]. We performed our experiments on 11

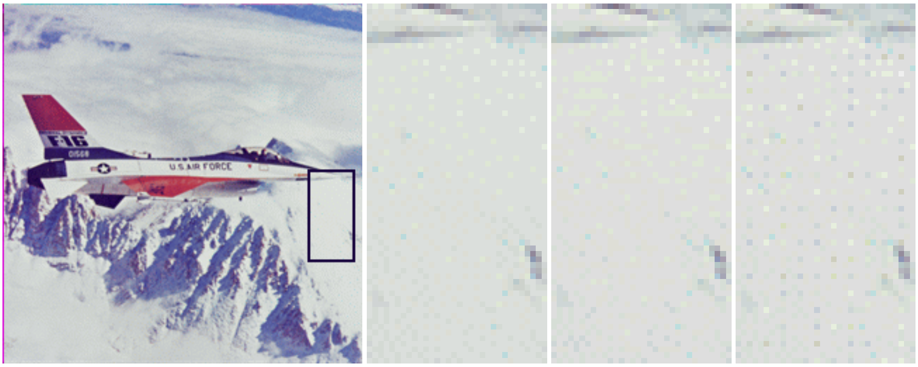
(a) Proposed ($k = 1$) (b) Proposed ($k = 2$) (c) Proposed ($k = 3$)

(d) Tanaka's ($k = 1$) (e) Tanaka's ($k = 2$) (f) Tanaka's ($k = 3$)

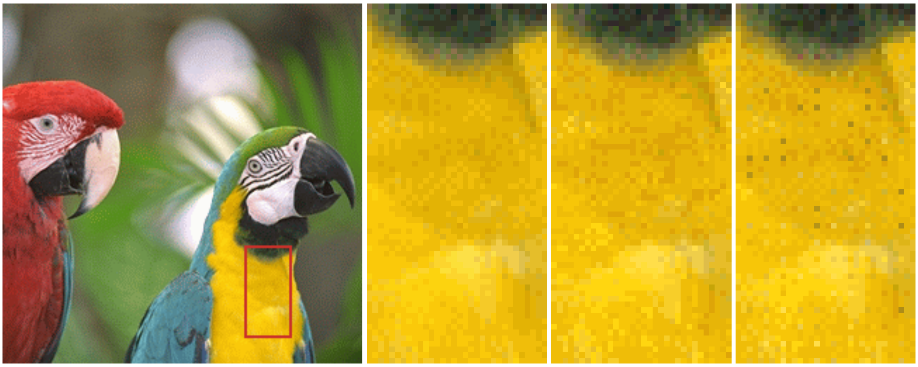
(g) Proposed ($k = 1$) (h) Proposed ($k = 2$) (i) Proposed ($k = 3$)

(j) Tanaka's ($k = 1$) (k) Tanaka's ($k = 2$) (l) Tanaka's ($k = 3$)

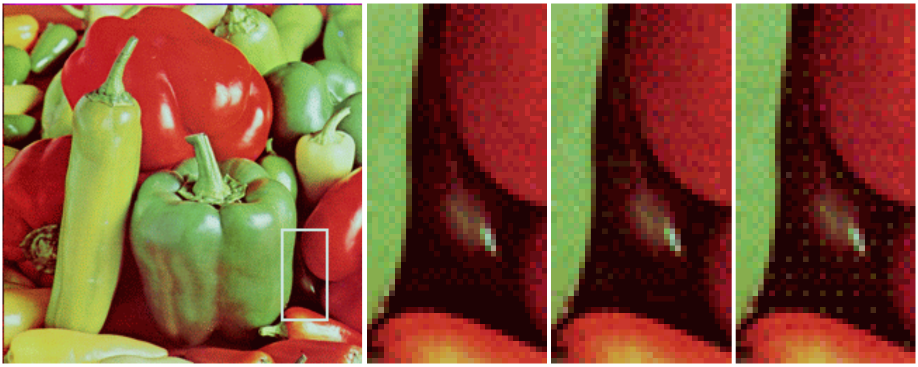
Fig. 5. Stego-images with a $7,000 \times k$-bit message (Airplane and Parrots)



(a) Airplane (b) Original (c) Proposed (d) Tanaka's [13]

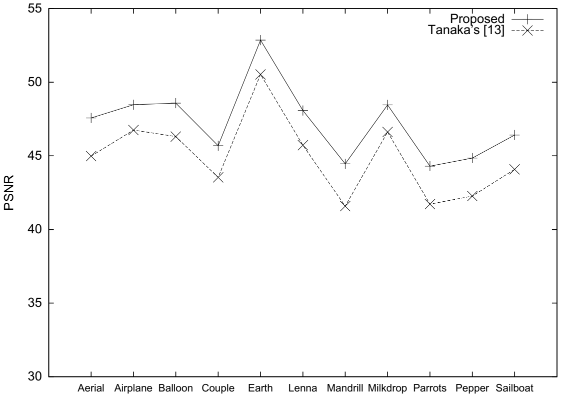


(e) Parrots (f) Original (g) Proposed (h) Tanaka's [13]

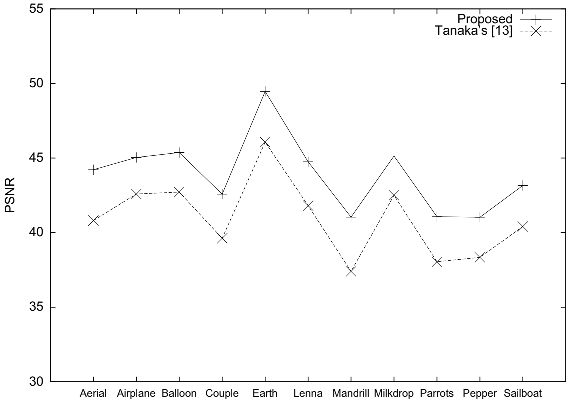


(i) Pepper (j) Original (k) Proposed (l) Tanaka's [13]

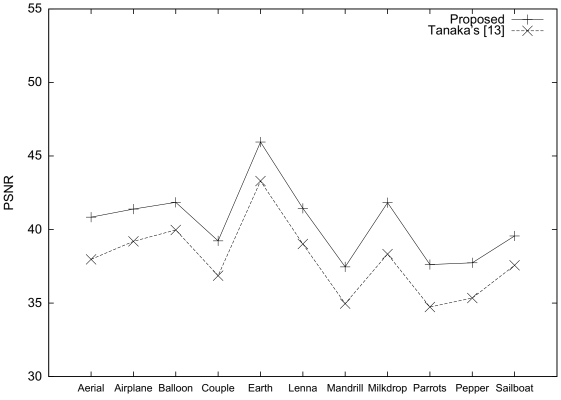
Fig. 6. Comparisons of stego-images between proposed scheme and Tanaka's scheme [13]



(a) $k = 1$



(b) $k = 2$



(c) $k = 3$

Fig. 7. Comparisons of PSNRs between proposed scheme and Tanaka's scheme [13]

Table 2. PSNRs of stego-images

	$k = 1$		$k = 2$		$k = 3$	
	Proposed	Tanaka's [13]	Proposed	Tanaka's [13]	Proposed	Tanaka's [13]
Aerial	47.57	44.98	44.22	40.81	40.84	37.97
Airplane	48.47	46.75	45.04	42.59	41.39	39.20
Balloon	48.58	46.31	45.37	42.72	41.85	39.97
Couple	45.69	43.54	42.58	39.62	39.23	36.86
Earth	52.86	50.53	49.47	46.07	45.95	43.30
Lenna	48.08	45.72	44.75	41.81	41.44	39.02
Mandrill	44.46	41.58	41.04	37.40	37.46	34.96
Milkdrop	48.46	46.62	45.14	42.50	41.83	38.33
Parrots	44.30	41.72	41.07	38.05	37.62	34.73
Pepper	44.85	42.27	41.03	38.34	37.74	35.34
Sailboat	46.41	44.08	43.16	40.41	39.56	37.57

palette-based images from SIDBA [14], that were 256×256 pixels, and embedded a $7,000 \times k$-bit message into each image.

Fig. 5 shows the stego-images of Airplane and Parrots from SIDBA with a $7,000 \times k$-bit message, where $k = 1, 2$, or 3, using the proposed scheme and Tanaka's scheme [13], respectively. Figs. 6(b), (f), and (j) are parts of the original images, which are enclosed by the squares in Figs. 6(a), (e), and (i), respectively. Figs. 6(c) and (d), Figs. 6(g) and (h), and Figs. 6(k) and (l) are the same parts of the stego-images when using the proposed scheme and Tanaka's scheme [13], respectively.

We summarized the evaluation using PSNR for the stego-images embedded by the proposed scheme in Table 2 and Fig. 7 to compare them with those for Tanaka's scheme [13]. The maximum difference between those two schemes is 3.64 and the minimum difference is 1.72. These results validate the proposed scheme.

5 Conclusion

We have proposed a high-capacity steganographic scheme for palette-based images that are only slightly degraded. The proposed scheme embeds a multiple-bit message within the unit of a pixel matrix to improve the quality of stego-images, while some conventional schemes can embed only a one-bit message within the unit of a pixel. The implementation of this scheme is simple and straightforward. A performance analysis proved the effectiveness of our scheme. Our future work involves improvement of the way to reorder the entries in the palette.

Acknowledgments. This work was supported by JSPS KAKENHI Grant Number 23800010.

References

1. Kahn, D.: The history of steganography. In: Anderson, R. (ed.) IH 1996. LNCS, vol. 1174, pp. 1–5. Springer, Heidelberg (1996)
2. Chandramouli, R., Kharrazi, M., Memon, N.D.: Image Steganography and Steganalysis: Concepts and Practice. In: Kalker, T., Cox, I., Ro, Y.M. (eds.) IWDW 2003. LNCS, vol. 2939, pp. 35–49. Springer, Heidelberg (2004)
3. Niimi, M., Noda, H., Kawaguchi, E., Eason, R.O.: High capacity and secure digital steganography to palette-based images. In: Proc. of IEEE ICIP, pp. II-917–II-920 (2002)
4. Wu, M.-Y., Ho, Y.-K., Lee, J.-H.: An iterative method of palette-based image steganography. Pattern Recognition Letters 25(3), 301–309 (2004)
5. Wang, X., Yao, Z., Li, C.-T.: A palette-based image steganographic method using colour quantisation. In: Proc. of IEEE ICIP, pp. II-1090–II-1093 (2005)
6. Zhao, H., Wang, H., Khan, M.K.: Steganalysis for palette-based images using generalized difference image and color correlogram. Signal Processing 91(11), 2595–2605 (2011)
7. Machado, R.: EZ Stego, Stego Online (1997), `http://www.stego.com`
8. Fridrich, J.: A new steganographic method for palette-based image. In: Proc. of IS&T PICS, pp. 285–289 (1999)
9. Tzeng, C.-H., Yang, Z.-F., Tsai, W.-H.: Adaptive data hiding in palette images by color ordering and mapping with security protection. IEEE Trans. Commun. 52(5), 791–800 (2004)
10. Zhang, X., Wang, S., Zhou, Z.: Multibit assignment steganography in palette images. IEEE Signal Proc. Lett. 15, 553–556 (2008)
11. Kim, S.-M., Cheng, Z., Yoo, K.-Y.: A new steganography scheme based on an index-color image. In: Proc. of International Conference on Information Technology: New Generations, pp. 376–381 (2009)
12. Wang, C.-T., Liao, C.-H., Chen, R.-M.: High Capacity Image Data Hiding Scheme for Grouping Palette Index. In: Proc. of International Conference on Multimedia and Ubiquitous Engineering, pp. 197–204 (2009)
13. Tanaka, G., Suetake, N., Uchino, E.: A steganographic method realizing high capacity data embedding for palette-based images. In: Proc. of International Workshop on Smart Info-Media Systems in Asia, pp. 92–95 (2009)
14. Onoe, M., Sasaki, M., Inamoto, Y.: SIDBA Standard Image Data Base. MIPC Report 79-1 (1979)