Embedding Information in Chinese Calligraphy Images

Zhi-Hui Wang[1] Chin-Chen Chang[2, 3, *] Ying-Hsuan Huang[4]
Hai-Rui Yang[1] Gwoboa Horng[4]

[1] School of Software, Dalian University of Technology
Dalian, Liaoning, China
{wangzhihui1017, hr.dlut}@gmail.com

[2, *] Department of Information Engineering and Computer Science, Feng Chia University
Taichung 40724, Taiwan, R.O.C.
alan3c@gmail.com

[3] Department of Computer Science and Information Engineering, Asia University
Taichung 41354, Taiwan, R.O.C.
alan3c@gmail.com

[4] Department of Computer Science and Engineering, National Chung Hsing University
Taichung 40227, Taiwan, R.O.C.
{phd9807, gbhorng}@cs.nchu.edu.tw

Received 18 June 2005; Revised 19 February 2006; Accepted 20 February 2006

Abstract. In this work, a vector quantization-based (VQ-based) watermark embedding method for binary calligraphy images is proposed in order to achieve copyright protection. Our method does not allow watermark bits to be embedded in the smooth regions of the image, so the visual effect of the image is maintained. Rather, complex regions are used to embed watermark bits. In the watermark embedding phase, the complex regions are modified according to similar codewords and the embedded watermark bits. The modification rule indicated that our method did not distort the images significantly. Moreover, the experimental results indicated that our method can resist some image processing attacks, e.g., salt and pepper noise, image cropping, and image blurring.

Keywords: vector quantization, watermark embedding, calligraphy images, image processing attacks

1 Introduction

Copyright protection is becoming increasingly important, and one method used to provide that protection is the use of digital watermarks [1]. In addition, the watermark can be used to authenticate whether the image suffered from the tampering or not, and then recover the tampered region of the image [2-5]. Most watermark techniques can be categorized into two domains, i.e., the spatial domain [6] and the frequency domain [7]. In the spatial domain, the pixels in the cover image are directly altered according to the watermark [6]. In the frequency domain, the cover image is processed by some pixel transformation equations, after which the watermark bits are embedded into the frequency coefficients [7]. In recent years, there has been increased interest in vector quantization (VQ), and some watermark schemes have been proposed based on VQ [8-16].

In 2000, Lu *et al*. presented a VQ watermark method that partitioned the codebook into a quantity of non-overlapping groups using a secret key and then embedded the watermark bits by modifying the VQ indices [9]. This method must have the original image to extract the watermark. Later, Lu *et al*. expanded the codebook to eliminate this limitation [10].

In 2002, Jo and Kim [11] proposed a new watermark scheme that paired two VQ indices with high similarity and then dispersed them into two different clusters for embedding the watermark bit, either "0" or "1." However, the difference of some VQ index pairs is large, so they are not used to embed the watermark bits.

In 2005, Wu and Chang proposed a watermark method based on the VQ encoding approach [12]. In their method, the VQ codebook was partitioned into two groups according to the similarity between the codewords. Then, the watermark bits were embedded into the VQ indices. This method can extract the watermark correctly even if the watermarked image has been subjected to some image-processing attacks.

In 2007, Wang *et al.* combined a genetic algorithm (GA) and the VQ-based embedding algorithm to embed one watermark [13]. Before embedding the watermark, the VQ codebook was partitioned into two groups according to a pre-determined key, and the groups were used to represent the two types of watermark bits (i.e., "0" and "1"). The GA changed the key continuously until the quality of the watermarked image was satisfactory. However, using the GA in this way incurred very high computational costs.

In 2010, Shen and Ren improved Wu and Chang's method [14] by using the VQ encoding algorithm to compress both the cover image and the watermark to derive two VQ index tables. The association rules in the watermark's VQ index table were embedded into the association rules of the cover image's VQ index table. Since the embedded data comprise the association rules rather than the watermark, the method can embed more watermarks than other methods. However, their method required extensive quantities of extra data to reconstruct the watermark.

Most of the proposed watermark schemes mentioned above have been applied to grayscale images rather than binary calligraphy images. Thus, we proposed a VQ-based watermark embedding method for binary calligraphy images. In the embedding phase, the smooth regions in the image remain unchanged to retain the good visual quality of the binary calligraphy image. Complex regions can embed the watermark by the VQ encoding method and our pre-established substitution rule. In the watermark extraction and authentication phases, most watermark bits can be extracted successfully, even if the image has been subjected to some image-processing attacks.

2 Proposed Method

In the following sections, we propose a VQ-based watermark scheme for binary calligraphy images. Our method was inspired by the concept presented in [11], in which Jo and Kim skillfully paired all of the codewords in a VQ codebook into closest pairs, and then dispersed all of the components in the same pairs into two different sets for the use of embedding watermark bits. The proposed method contains three phases, i.e., generating the codebook, embedding the watermark, and extracting the watermark.

2.1 Generating the Codebook

The VQ compression method was proposed by Linde *et al.* in 1980 [17]. In this method, one $H \times W$ binary image is divided into $(H \times W)/(n \times n)$ non-overlapping blocks, where $n \times n$ denotes the block size. Thus, one block B_i consists of $n \times n$ bits, i.e., $B_i = \{b_1, b_2, \ldots, b_{n\times n}\}$, where i denotes the ID number of the block. In these blocks, K blocks are selected randomly as the candidate blocks for use in classifying all of the blocks into K groups. The details of the classification are described below.

Let the bits in the candidate block C_l be $\{c_1, c_2, \ldots, c_{n\times n}\}$, where $1 \le l \le K$. The Hamming distances, h_l, between the block B_i and all of the candidate blocks are calculated, i.e., $h_l = \sum_{x=1}^{n\times n} b_x \otimes c_x$. Let the j^{th} Hamming distance is the shortest of the K Hamming distances, e.g., $h_j = \min\{h_l\}$; then, block B_i belongs to the j^{th} group. After each block is classified, the centroid of each group is calculated as a new candidate block, i.e., $C'_l = \{c'_1, c'_2, \ldots, c'_{n\times n}\}$. The above procedures are repeated until the centroid of each group is constant.

2.2 Embedding the Watermark

Before embedding the watermark, each set of two similar codewords, in which their Hamming distance equals the threshold T, is paired and put into two different groups (G_1, G_2), respectively, where the initial value of the threshold T is one. Note that each codeword is only paired one time. According to the above rules, most codewords can be paired successfully, but some codewords cannot be paired because their Hamming distance is greater than T; thus, we let $T = T + 1$ and repeat the above procedure until all codewords have been paired.

To enhance security, the watermark is encrypted by a user key. Let the encrypted watermark bit be w, where $w \in \{0, 1\}$. The watermark embedding procedure is described as follows. One binary calligraphy image is resolved into $(H \times W)/(n \times n)$ blocks, e.g., $B_i = \{b_1, b_2, \ldots, b_{n\times n}\}$. A block B_i is labeled as non-embeddable by setting one authentication bit to be "–1", i.e., $a_i = -1$ if all pixels in the block are white or black, i.e., $\sum_{x=1}^{n\times n} b_x = n\times n$ or $\sum_{x=1}^{n\times n} b_x = 0$. This is because the non-embeddable blocks are located in the very smooth region of the binary image, which does not allow for any modification to avoid reducing the visual quality. Other embeddable blocks are compressed using the VQ encoding algorithm to generate VQ indices I_i. The VQ index I_i is modified according to its group ID number and the watermark bit w. The modification formula is

$$I_i' = \begin{cases} I_i, & \text{if } I_i \in G_1 \text{ and } w = 0, \\ s(I_i), & \text{if } I_i \in G_1 \text{ and } w = 1, \\ s(I_i), & \text{if } I_i \in G_2 \text{ and } w = 0, \\ I_i, & \text{if } I_i \in G_2 \text{ and } w = 1, \end{cases} \qquad (1)$$

where $s(I_i)$ denotes another VQ index in the same pair. Note that the embeddable block cannot be paired with the entirely white block or the entirely black block. Block B_i is replaced by the codeword of the modified VQ index I_i'. Moreover, the authentication bit a_i of block B_i is set to w, i.e., $a_i = w$.

2.3 Extracting the Watermark

The watermark is extracted from the image, if the copyright is being questioned, and then the watermark is verified by the preserved authentication bits. The watermarked image is partitioned into a number of $n \times n$ non-overlapping blocks, i.e., $B_i' = \{b_1', b_2', \ldots, b_{n\times n}'\}$. No watermark bit w' exists in block B_i' if $a_i = -1$. Otherwise, if $a_i > -1$, then one watermark bit w' will be extracted by the following procedure.

First, the block is compressed using the VQ encoding algorithm to obtain one VQ index I_i'. Then, according to its group ID number, one watermark bit, w', is extracted by

$$w' = \begin{cases} 0, \text{ if } I_i' \in G_1, \\ 1, \text{ otherwise}. \end{cases} \qquad (2)$$

The extracted watermark bit w' can be matched with the authentication bit a_i to achieve copyright authorization. The image copyright is valid if most of the extracted watermark bits are equal to the authentication messages.

3 Experimental Results

Figs. 1(a) through 1(e) show five binary calligraphy images [18-22] and one watermark [23] that have sizes of 988×1200, 1401×1422, 1000×522, 2554×885, 1144×1160, and 56×59, respectively. In the experiments, the block size was set to 4×4, and the codebook size was set to 64. All of the test images are used to generate a VQ codebook. Figs. 2(a) through 2(e) show that the visual appearance of the watermarked image is similar to that of the original image because our method does not modify the smooth regions in the binary image. Fig. 2(f) shows the extracted watermark without any distortion.

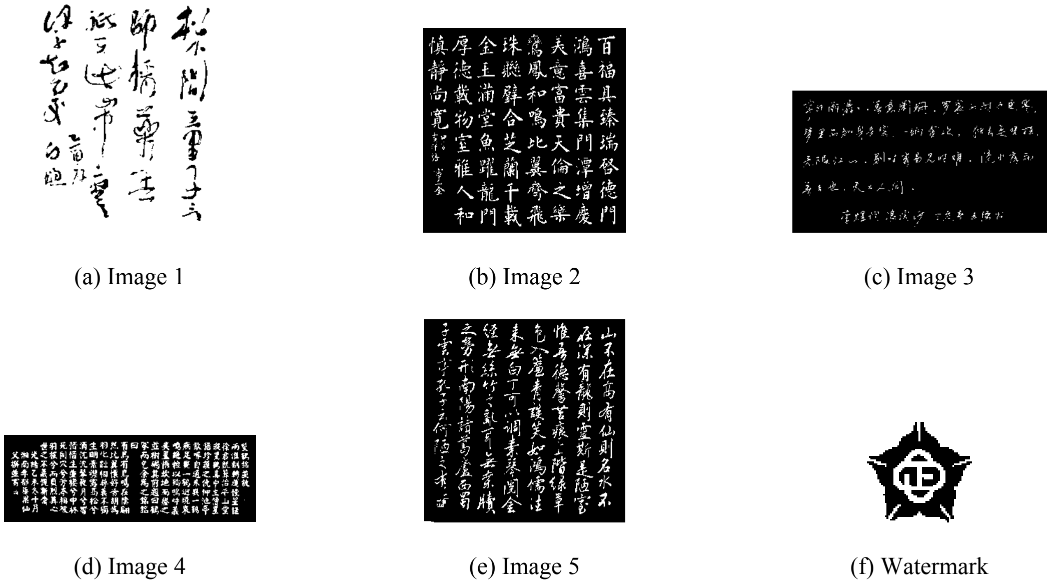
Fig. 3(a) shows our peak signal-to-noise ratio (PSNR) values by using different codebook sizes, i.e., 64, 128, 256, 512, and 1024. The greater the codebook size was, the greater the PSNR values became; this occurred because, the larger codebook has more representative codewords than the smaller codebook, so they achieve greater PSNR values. However, these typical codewords cannot be modified, even slightly. Thus, after adding the same 5% salt and pepper noise, the normalized correlation (NC) value of the extracted watermark with the larger size codebook is lower than that of the smaller size codebook, see Fig. 3(b). The equation of the NC value is

$$\frac{\text{Number of the correctly watermark bits}}{\text{Number of watermark bits}}. \qquad (3)$$

Therefore, the higher NC value implies that the proposed method can achieve high robustness.

Figs. 4(a) through 4(d) show that our proposed method can extract one visible watermark when the watermarked image has 5% to 20% of salt and pepper noise added. Furthermore, Fig. 5 shows that the proposed method can extract one identifiable watermark even if half of the watermarked image is cropped. This is because some significant watermark bits in the watermark are embedded into the other half of the image. Thus, these significant watermark bits can be extracted correctly, thereby revealing the watermark. Fig. 6 shows that the watermark can be extracted even after being subjected to 11% Gaussian noise. These experimental results indicated that our robust watermark algorithm can effectively resist some image-processing attacks, such as salt and pepper noise, image cropping, and image blurring.

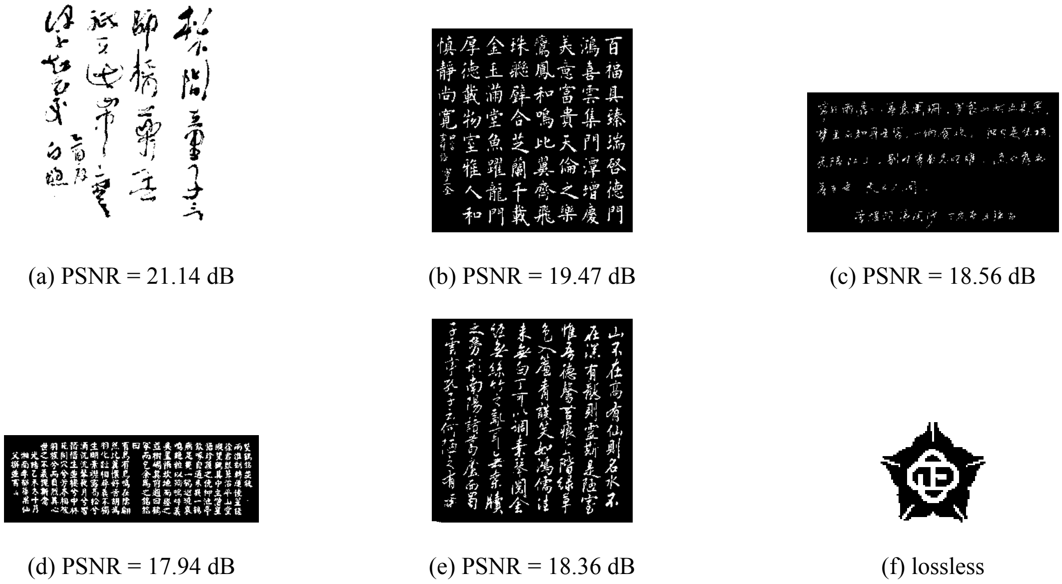
Fig. 7 shows the embedding capacity for each of our binary calligraphy images. Our method fits complex images (images 1 and 3) because the smooth blocks in the image are not used to embed watermark bits. Our method provides great robustness, and the embedding capacity is almost as high as that of Wu and Liu's method [24], which is purely a data hiding method with the same level of quality of the stego image.



(a) Image 1 (b) Image 2 (c) Image 3

(d) Image 4 (e) Image 5 (f) Watermark

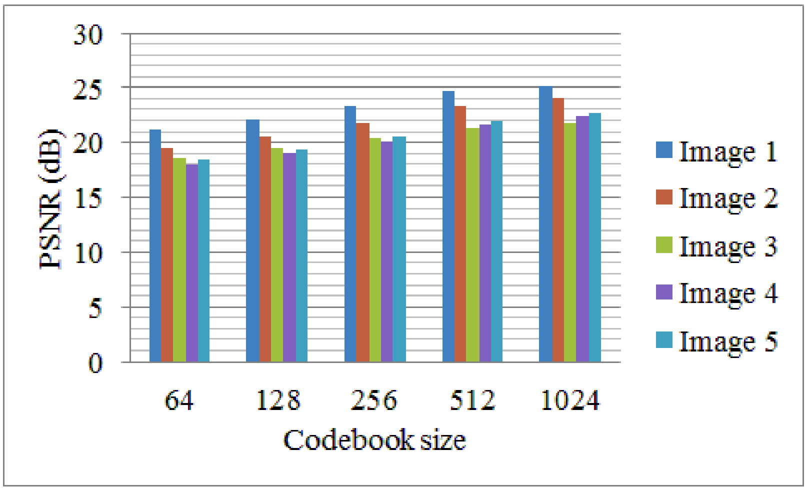
Fig. 1. Five binary calligraphy images and one watermark



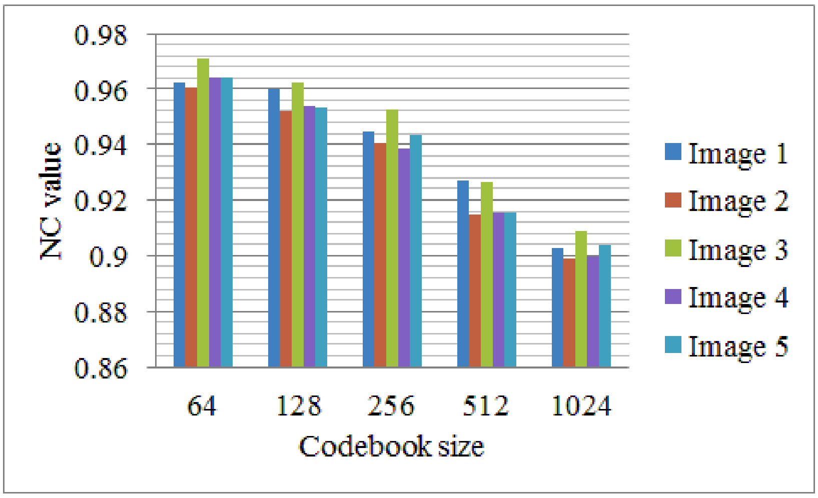
(a) PSNR = 21.14 dB (b) PSNR = 19.47 dB (c) PSNR = 18.56 dB

(d) PSNR = 17.94 dB (e) PSNR = 18.36 dB (f) lossless

Fig. 2. Five watermarked images and one extracted watermark



(a) PSNR values

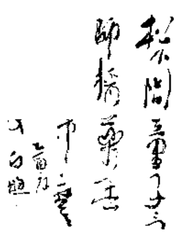


(b) NC values

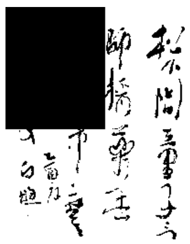
Fig. 3. Performance comparison of five watermarked images with variant codebook sizes

(a) NC = 0.965 (b) NC = 0.925 (c) NC = 0.883 (d) NC = 0.842

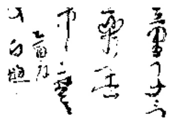
Fig. 4. Extracted watermarks, which suffer from 5%, 10%, 15% and 20% salt and pepper noise attacks, respectively



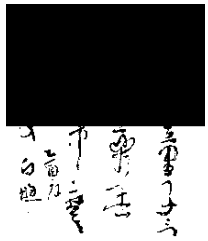
(a) 25% cropping attack



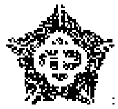
(b) 25% cropping and pasting attacks



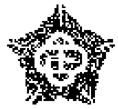
(c) 50% cropping attack



(d) 50% cropping and pasting attacks

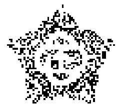


(e) NC = 0.876



(f) NC = 0.876

(g) NC = 0.757



(h) NC = 0.757

Fig. 5. Four cropped and pasted images and four extracted watermarks

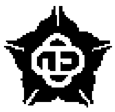


Fig. 6. Extracted watermark

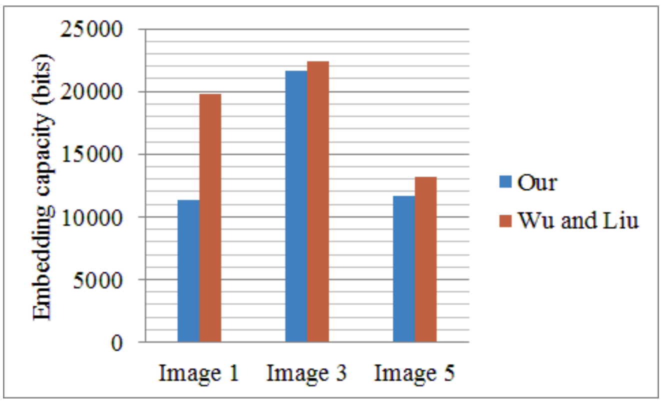


Fig. 7. Comparison of embedding ability for our method and previous method [24]

4 Conclusions

The proposed method can embed watermarks into binary calligraphy images. In the embedding phase, all smooth regions in the image remain unchanged to avoid decreasing the visual quality of the image. Complex regions are modified according to similar codewords and the embedded watermark bits. Thus, the binary image is modified only slightly, which is the reason the PSNR value is high in our scheme. The experimental results showed that our method can embed massive numbers of watermark bits and resist some image-processing attacks.

However, in our method, the searching process of the most similar VQ index increases the time complexity and requires more memory spaces. In the future, we will try to combine the proposed method with the SMVQ technique [25] to reduce the computational cost.

Acknowledgement

This work was supported by National Science Foundation of China under Grant No. 61272374 and Innovation Leading Foundation of Software School, Dalian University of Technology.

References

[1] M.S. Hwang, C.C. Chang, K.F. Hwang, "A Watermarking Technique Based on One-Way Hash Functions," *IEEE Transactions on Consumer Electronics*, Vol. 45, No. 2, pp. 286-294, 1999.

[2] C. Qin, C.C. Chang, T.J. Hsu, "Effective Fragile Watermarking for Image Authentication with High-Quality Recovery Capability," *KSII Transactions on Internet and Information Systems*, Vol. 7, No. 11, pp. 2941-2956, 2013.

[3] C.C. Chang, C.C. Lin, L.T. Liao, "A DCT-Based Dual Watermark Scheme for Image Authentication and Recovery," *Advance in Information Science and Service Science*, Vol. 5, No. 12, pp. 56-69, 2013.

[4] C. Qin, C.C. Chang, P.Y. Chen, "Self-Embedding Fragile Watermarking with Restoration Capability Based on Adaptive Bit Allocation Mechanism," *Signal Processing*, Vol. 92, No. 4, pp. 1137-1150, 2012.

[5] C.C. Chang, K.N. Chen, C.F. Lee, L.J. Liu, "A Secure Fragile Watermarking Scheme Based on Chaos-and-Hamming Code," *Journal of Systems and Software*, Vol. 84, No. 9, pp. 1462-1470 , 2011.

[6] J.C. Liu and S.Y. Chen, "Fast Two-Layer Image Watermarking without Referring to the Original Image and Watermark," *Image and Vision Computing*, Vol. 19, No. 14, pp. 1083-1097, 2001.

[7] C.S. Shieh, H.C. Huang, F.H. Wang, J.S. Pan, "Genetic Watermarking Based on Transform-Domain Techniques," *Pattern Recognition*, Vol. 37, No. 3, pp. 555-565, 2004.

[8] H.C. Huang, F.H. Wang, J.S. Pan, "A VQ-Based Robust Multi Watermarking Algorithm," *IEICE Transactions on Fundamentals of Electronics*, Vol. E85-A, No. 7, pp. 1719-1726, 2002.

[9] Z.M. Lu and S.H. Sun, "Digital Image Watermarking Technique Based on Vector Quantisation," *Electronics Letters*, Vol. 36, No. 4, pp. 303-305, 2000.

[10] Z.M. Lu, J.S. Pan, S.H. Sun, "VQ-Based Digital Image Watermarking Method," *Electronics Letters*, Vol. 36, No. 14, pp. 1201-1202, 2000.

[11] M. Jo and H.D. Kim, "A Digital Image Watermarking Scheme Based on Vector Quantization," *IEICE Transactions on Information and Systems*, Vol. E85-D, No. 6, pp. 1054-1056, 2002.

[12] H.C. Wu and C.C. Chang, "A Novel Digital Image Watermarking Scheme Based on the Vector Quantization Technique," *Computers & Security*, Vol. 24, No. 6, pp. 460-471, 2005.

[13] F.H. Wang, L.C. Jain, J.S. Pan, "VQ-Based Watermarking Scheme with Genetic Codebook Partition," *Journal of Network and Computer Applications*, Vol. 30, No. 1, pp. 4-23, 2007.

[14] J.J. Shen and J.M. Ren, "A Robust Associative Watermarking Technique Based on Vector Quantization," *Digital Signal Processing*, Vol. 20, No. 5, pp. 1408-1423, 2010.

[15] W.J. Wang, C.T. Huang, C.H. Yang, S.J. Wang, "VQ-Based Algorithms Extended to Non-Embedded Watermarking for Multimedia Ownership Prevention Systems," *Peer-to-Peer Networking and Applications*, Vol. 7, pp. 676-686, 2013.

[16] C.F. Lee, K.N. Chen, C.C. Chang, M.C. Tsai, "A Hierarchical Fragile Watermarking with VQ Index Recovery," *Journal of Multimedia*, Vol. 6, No. 3, pp. 277-284, 2011.

[17] Y. Linde, A. Buzo, R.M. Gray, "An Algorithm for Vector Quantizer Design," *IEEE Transactions on Communications*, Vol. 28, No. 1, pp. 84-95, 1980.

[18] Art Network Union, http://www.art86.cn, 2013.

[19] China Central TeleVision, http://img.blog.cctv.com/attachments/2009/06/835115200906170816241.jpg, 2013.

[20] China Central TeleVision, http://img.blog.cctv.com/attachments/2008/09/835115200809241440271.jpg, 2013.

[21] Calligraphy Presentation, http://www.yac8.com, 2013.

[22] Hudong, http://a3.att.hudong.com/02/04/01200000028610134400049206017.jpg, 2013.

[23] National Chung Cheng University, http://www.ccu.edu.tw, 2013.

[24] M. Wu and B. Liu, "Data Hiding in Binary Image for Authentication and Annotation," *IEEE Transactions on Multimedia*, Vol. 6, No. 4, pp. 528-538, 2004.

[25] C. Qin, C.C. Chang, Y.P. Chiu, "A Novel Joint Data-Hiding and Compression Scheme Based on SMVQ and Image Inpainting," *IEEE Transactions on Image Processing*, Vol. 23, No. 3, pp. 969-978, 2014.