MVA2005 IAPR Conference on Machine VIsion Applications, May 16-18, 2005 Tsukuba Science City, Japan

13-30

An Image Watermarking Method Based on Mean-Removed Vector Quantization for Multiple Purposes

Zhe-Ming Lu
Department of Automatic Test and Control,
Harbin Institute of Technology, Harbin 150001,
China
zhemingl@yahoo.com

Zhen Sun
Department of Automatic Test and Control,
Harbin Institute of Technology, Harbin 150001,
China

Abstract

Digital watermarking technique has been presented to solve copyright protection, copy protection, and content authentication issues in the digital world. Conventional watermarking algorithms are mostly designed for only one purpose. In recent years, some multipurpose digital watermarking methods based on discrete wavelet transform (DWT) and discrete Fourier transform (DFT) have been presented to achieve the goal of both content authentication and copyright protection. In this paper, we present a novel multipurpose digital image watermarking method based on a mean-removed vector quantizer structure. In the proposed method, the fragile watermark and the robust watermark are embedded in mean indices and residual indices respectively. Simulation results demonstrate the effectiveness in terms of robustness and fragility.

1 Introduction

The explosive growth of digital multimedia techniques and digital network communications has created a pressing demand for techniques used for copy protection, copyright protection, and content authentication. Over the last decade, digital watermarking has been presented to complement cryptographic processes. Digital watermarking is a technique to insert a secret signal in digital data, which enables one to establish ownership or identify a buyer. Most of existing invisible watermarking schemes are designed for either copyright protection or content authentication. Invisible watermarks can be broadly classified into two types, *robust* and *fragile* watermarks. Robust watermarks [1,2] are generally used for copyright protection and ownership verification because they are robust to nearly all kinds of image processing operations. In comparison, fragile watermarks [3,4] are mainly applied to content authentication and integrity attestation because they are completely fragile to any modifications. To fulfill multipurpose applications, several multipurpose watermarking algorithms based on wavelet transform [5] and fast Fourier transform [6] have been presented. Recently, some robust image watermarking techniques based on vector quantization (VQ) [7]-[13] have been presented. In this paper, we present a novel multipurpose watermarking method based on mean-removed vector quantization. In the proposed algorithm, the robust watermark is embedded in the quantized mean indices by using the embedding method presented in [12], and the fragile watermark is embedded in the residual codeword indices by using a novel index constrained method.

2 Previous VQ Watermarking Algorithms

2.1 Codebook partition based

The main idea of the VQ-based digital watermarking schemes presented in [7]-[10] is to carry secret copyright information by codeword indices. The aim of the codebook partition is to classify the neighboring codewords into the same cluster. Before the embedding process, the original image is first divided into blocks. For each block, the index of the best match codeword is found. The watermarked codeword index is then obtained by modifying the original codeword index according to the corresponding watermark bits. The modification is under the constraint that the modified index and the original one is in the same partition such that the introduced extra distortion is less than the given distortion threshold. In the decoding phase, not the original but the watermarked codeword is used to represent the input image block.

2.2 Index properties based

To enhance the robustness to rotation operations and VQ compression operations, some image watermarking algorithms [12,13] based on the properties of neighboring indices have been proposed. In [12], the original watermark $\boldsymbol{W}$ with size $A_w \times B_w$ is first permuted by a predetermined key, key_1, to generate the permuted watermark $\boldsymbol{W}_P$ for embedding. The original image $\boldsymbol{X}$ with size $A \times B$ is then divided into vectors $\boldsymbol{x}(h,l)$ with size $(A/A_w)\times(B/B_w)$, where $\boldsymbol{x}(h,l)$ denotes the image block at the position of (h,l). After that, each vector $\boldsymbol{x}(h,l)$ finds its best codeword $\boldsymbol{c}_i$ in the codebook C and the index i is assigned to $\boldsymbol{x}(h,l)$, we can then obtain the indices matrix $\boldsymbol{Y}$ with elements $y(h,l)$, which can be represented by

$$\boldsymbol{Y} = \mathrm{VQ}(\boldsymbol{X}) = \bigcup_{h=0}^{\frac{A}{A_w}-1} \bigcup_{l=0}^{\frac{B}{B_w}-1} \mathrm{VQ}(\boldsymbol{x}(h,l)) = \bigcup_{h=0}^{\frac{A}{A_w}-1} \bigcup_{l=0}^{\frac{B}{B_w}-1} y(h,l) \quad (1)$$

After calculating the variances of $y(h,l)$ and the indices of its surrounding blocks with

$$\sigma^2(h,l) = \left(\frac{1}{9}\sum_{i=h-1}^{h+1}\sum_{j=l-1}^{l+1} y^2(i,j)\right) - \left(\frac{1}{9}\sum_{i=h-1}^{h+1}\sum_{j=l-1}^{l+1} y(i,j)\right)^2 \quad (2)$$

We can obtain the polarities $\boldsymbol{P}$ as follows

$$\boldsymbol{P} = \bigcup_{h=0}^{\frac{A}{A_w}-1} \bigcup_{l=0}^{\frac{B}{B_w}-1} p(h,l) \qquad (3)$$

$$p(h,l) = \begin{cases} 1 & \text{if } \sigma^2(h,l) \geq T \\ 0 & \text{otherwise} \end{cases} \qquad (4)$$

For convenience, we set the threshold T to be half of the codebook size, N/2. We are then able to generate the final embedded watermark or the secret key, key_2, with the exclusive-or operation as $key_2 = \boldsymbol{W_P} \oplus \boldsymbol{P}$. After the inverse-VQ operation, both the reconstructed image $\boldsymbol{X}'$ and the secret key, key_2, work together to protect the ownership of the original image. In the extraction process, we first calculate the estimated polarities $\boldsymbol{P}'$ from $\boldsymbol{X}'$, and then obtain an estimate of the permuted watermark as $\boldsymbol{W}'_{\boldsymbol{P}} = key_2 \oplus \boldsymbol{P}'$. Finally, we can perform the inverse permutation operation with key_1 to obtain the extracted watermark $\boldsymbol{W}'$.

The above algorithm has the following problems: First, we can also extract the watermark from the original image without watermark at all. Secondly, the codebook should be used as a key, because if the user possesses the same codebook, he can also embed his watermark in the watermarked image without any modification.

3 Proposed Multipurpose Algorithm

3.1 The embedding process

Before describing the proposed algorithm, we make some assumptions. Let $\boldsymbol{X}$ be the original image with size A×B, let $\boldsymbol{W}_R$ and $\boldsymbol{W}_F$ be the binary robust and fragile watermarks with size A_w×B_w, respectively. Here, a small visually meaningful binary image $\boldsymbol{V}$ with size a×b is replicated periodically to obtain the binary fragile watermark $\boldsymbol{W}_F$ with size A_w×B_w that is large enough for embedding. In the proposed algorithm, only one bit is embedded in the mean or residual index of each image block (or vector), so the dimension of each input vector or codeword is k=(A/A_w)×(B/B_w). Assume that the mean codebook is C_m={$\hat{m}_0, \hat{m}_1$, …, $\hat{m}_{N_m-1}$} with size N_m=2^{n_m} and the residual codebook is C_r={$\hat{\boldsymbol{r}}_0$, $\hat{\boldsymbol{r}}_1$, …, $\hat{\boldsymbol{r}}_{N_r-1}$} with size N_r=2^{n_r}, where n_m and n_r are natural numbers. Thus a binary number with n_m+n_r bits, in which the first n_m bits stand for the mean index and the last n_r bits denote the residual index, can represent the overall index. The overall codeword can be selected from the equivalent product codebook C={$\boldsymbol{c}_0$, $\boldsymbol{c}_1$, …, $\boldsymbol{c}_{N\text{-}1}$} with size N=N_m×N_r. In other words, if the index in codebook C_m is i and the index in codebook C_r is j, then the equivalent overall index in the product codebook C is j+i×N_r. In what follows, we describe the two embedding processes separately.

The robust watermark embedding process In the proposed algorithm, we adopt the method [12] based on index properties to embed the robust watermark in the mean codeword indices. For convenience of description, the mean scalar quantization here is looked upon as the mean vector quantization (VQ$_m$), where all components of a mean vector (or codeword) are equal to its mean value. The original watermark $\boldsymbol{W}_R$ is first permuted by a predetermined key, key_1, to generate the permuted watermark $\boldsymbol{W}_{RP}$ for embedding. The polarities $\boldsymbol{P}$ can then be calculated with (1)-(4). Finally, we generate the final embedded watermark or the secret key, key_2, with the exclusive-or operation (5). After the robust embedding, we can obtain the reconstructed image $\boldsymbol{X}'$ and the residual image $\boldsymbol{X}_r$ as follows

$$\boldsymbol{X}' = \mathrm{VQ}_m^{-1}[\mathrm{VQ}_m[\boldsymbol{X}]] \qquad (5)$$

$$\boldsymbol{X}_r = \boldsymbol{X} - \boldsymbol{X}' \qquad (6)$$

According to Section 2.2, we know that this method has two problems. However, in our algorithm, these two problems can be automatically solved, which will be discussed later in the extraction process.

The fragile watermark embedding process To embed one bit in each residual index, we can adopt an index constrained vector quantization (ICVQ) encoding scheme. Because each index has n_r bits, we can select an embedding position from n_r candidate positions. Assume that we select Position key_3, which is considered as a key, to embed the watermark bit, where $0 \leq key_3 \leq n_r\text{-}1$. Unlike the normal VQ encoder, the embedding process for each watermark bit can be performed by searching the best match codeword $\hat{\boldsymbol{r}}_p$ for each input residual vector under the constraints that the key_3-th bit of index p is equal to the watermark bit to be embedded. After the normal VQ decoder, we can obtain the reconstructed residual image as follows

$$\boldsymbol{X}'_r = \mathrm{VQ}_r^{-1}[\mathrm{ICVQ}_r[\boldsymbol{X}_r]] \qquad (7)$$

And then we obtain the final watermarked image by

$$\boldsymbol{X}_W = \boldsymbol{X}' + \boldsymbol{X}'_r \qquad (8)$$

3.2 The extraction process

To enhance the security of our embedding process, we use the equivalent product codebook C in the extraction process, that is to say, the mean and residual codebooks are used as secret keys while the product codebook is open for user. In addition, because the users don't know the mean and residual codebook sizes used in mean-removed VQ either, how to segment the overall index into the mean index and the residual index is also a secret key, key_5, to users. In order to make the embedding algorithm more secretly, we can also permute the product codebook and then publicize the permuted codebook C_u for users. The extraction process can be performed without the original image and can be described as follows: Firstly, perform the inverse permutation operation with key_4 on Codebook C_u to obtain the product codebook C. Secondly, the watermarked image $\boldsymbol{X}_W$ is divided into blocks or vectors. Thirdly, the normal VQ encoder performs the nearest neighbor codeword search on all input vectors to obtain the encoded overall indices. Fourthly, according to the two codebook sizes, each overall index is segmented into two indices. One is for robust watermark extraction; the other is for fragile watermark extraction. Finally, the robust and fragile watermarks are extracted independently. For the robust watermark extraction, we first compute the polarities $\boldsymbol{P}$ from the mean indices, and then perform XOR operation between $\boldsymbol{P}$ and key_2 to obtain the extracted permuted robust watermark $\boldsymbol{W}_{EPR}$, and finally perform inverse permutation operation with key_1 to obtain the extracted robust watermark $\boldsymbol{W}_{ER}$. For the fragile watermark extraction, we can simply check the key_3-th bit of each residual index to obtain the extracted watermark bit, where key_3 is just the watermarking position, and then piece all extracted bits together to form the extracted fragile watermark $\boldsymbol{W}_{EF}$.

In Section 2.2, we point out two problems of the robust embedding technique [12]. However, in our algorithm, these two problems can be automatically solved. Detecting the inexistence of the fragile watermark in the original image can solve the first problem. Using not the mean and residual codebooks but the equivalent product codebook to extract the watermarks can solve the second one.

4 Experimental Results

To evaluate the performance of the proposed method, the 512×512 Lena image with 8bits/pixel resolution is used for multipurpose watermarking. The Lena image is divided into 16384 blocks of size 4×4 for VQ encoding. A binary image of size 32×32 is replicated for 16 times to obtain a binary watermark $\boldsymbol{W}_F$ with size 128×128 for fragile watermarking. Another binary watermark $\boldsymbol{W}_R$ with size 128×128 is used for robust watermarking. The original Lena image and two watermarks are shown in Fig. 1 (a)(c)(d). The mean codebook Cm with size 16 and the residual codebook C_r with size 256 are obtained by the well-known LBG algorithm [14], which corresponds to 4+8=12 bits per overall index. If we embed the fragile watermark in the residual index, then we can randomly select the watermarking position key_3 ranged from 0 to 7 for the fragile watermarking. Before extraction, the equivalent product codebook C with size 16×256=4096 can be generated by the Cartesian product $C_m \times C_r$. Fig. 1(b) shows the watermarked image with PSNR=30.398dB obtained by the proposed method.

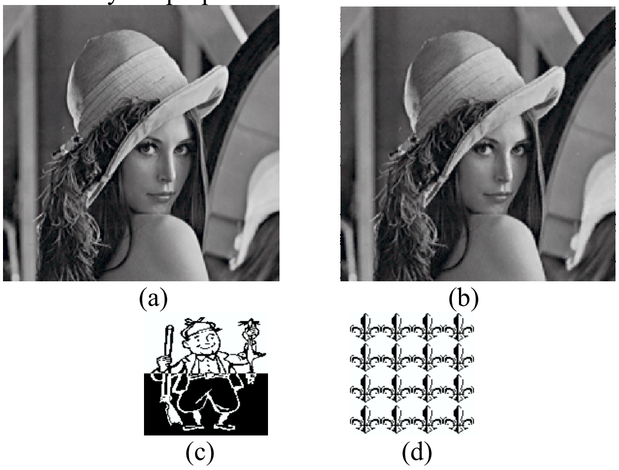

Figure 1. Original and watermarked images and original watermarks.

Here, we employ the normalized Hamming Similarity, NHS, to evaluate the effectiveness of the proposed algorithm. The NHS between the embedded binary watermark $\boldsymbol{W}$ and the extracted one $\boldsymbol{W}'$ is defined as

$$\mathrm{NHS} = 1 - \frac{\mathrm{HD}(\boldsymbol{W}, \boldsymbol{W}')}{A_w \times B_w} \tag{9}$$

Where HD(· , ·) denotes the Hamming distance between two binary strings, i.e., the number of bits different in the two binary strings. Results show that the robust and fragile watermarks extracted from the watermarked image without any attack are both with NHS=1.0.

To check the robustness and fragility of our algorithm, we perform several attacks on the watermarked image, including JPEG compression, VQ compression, spatial image processing and rotation. We perform JPEG compression with different quality factors (QF) on the watermarked image with QF=100%, 80%, 50% and 30%, respectively. The extracted watermarks and NHS values are depicted in Fig. 2. From these results, we can see that the proposed algorithm is robust to JPEG compression. For the case that QF is larger than 80%, the extracted watermarks, both robust and fragile, are similar to the embedded ones. For all cases, the extracted robust watermarks are with relatively high NHS values.

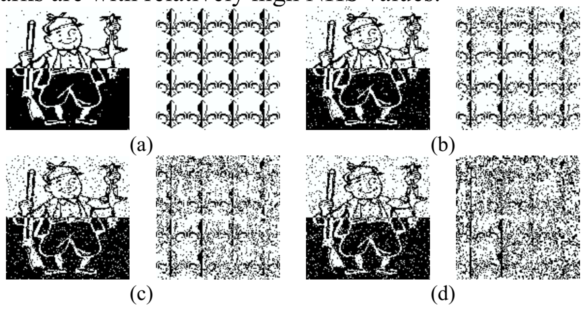

Figure 2. Watermarks extracted from JPEG compressed watermarked images. (a) QF=100%, robust NHS=0.991 and Fragile NHS=0.988. (b) QF=80%, robust NHS=0.935 and fragile NHS=0.828. (c) QF=50%, robust NHS=0.883 and fragile NHS=0.605. (d) QF=30%, robust NHS=0.854 and fragile NHS=0.415.

We use four different codebooks to compress the watermarked image. Codebook 1 is the product codebook used in our method. Codebook 2 with size 8192 and Codebook 3 with size 256 are both trained from the Lena image. Codebook 4 with size 4096 is trained from the Pepper image. Fig. 3 shows the watermarks extracted from these images. From these results, we can see that the proposed algorithm can extract the same watermarks as the embedded ones from the VQ compressed watermarked image with the product codebook. The reason is that the watermarked image isn't modified under the VQ compression with the product codebook. For other cases, the robust watermark can tolerate the VQ compression, while the fragile watermark cannot. The higher the codebook performance is, the larger the NHS value of the fragile watermark is.

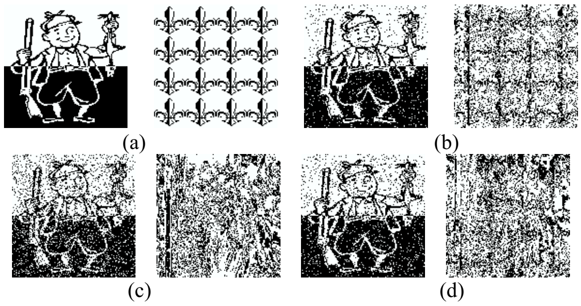

Figure 3. Watermarks extracted from VQ compressed watermarked images. (a) Codebook 1, robust NHS=1.0 and fragile NHS=1.0. (b) Codebook 2, robust NHS=0.893 and fragile NHS=0.566. (c) Codebook 3, robust NHS=0.720 and fragile NHS=0.173. (d) Codebook 4, robust NHS=0.834 and fragile NHS=0.287.

Several spatial-domain image processing techniques, including image cropping, median filtering, blurring, sharpening, contrast enhancement, adding Guassian noise are performed on the watermarked image. The extracted watermarks are depicted in Fig. 4. For each case, the robust watermark can successfully survive with NHS>0.77. For the case of image cropping in the upper-left corner, the extracted fragile watermark can locate the cropping position. For each case, the fragile watermark can be used to verify the watermarked image.

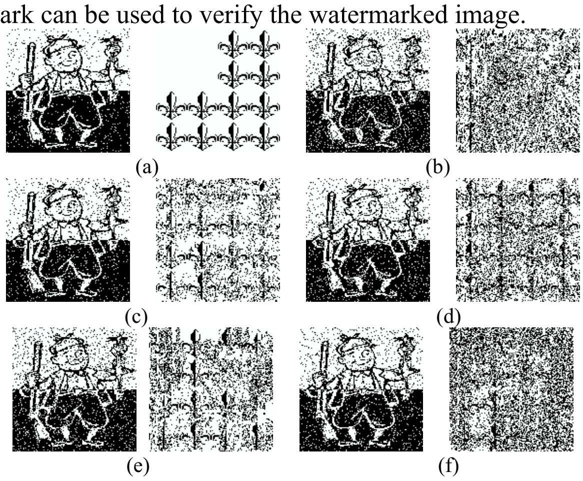

Figure 4. Watermarks extracted from spatial-domain-attacked watermarked images. (a) Image cropping in the upper-left corner, robust NHS=0.902 and fragile NHS=0.895. (b) Median filtering with the radius of 2 pixels, robust NHS=0.776 and fragile NHS=0.254. (c) Blurring with radius=1.0 and threshold=10.0, robust NHS=0.859 and fragile NHS=0.632. (d) Sharpening, robust NHS=0.825 and fragile NHS=0.514. (e) Contrast Enhancement by 10%, robust NHS=0.800 and fragile NHS=0.245. (f) Adding Guassian noise by the amount of 4%, robust NHS=0.833 and fragile NHS=0.222.

With StirMark, we perform the geometric attack by rotating the watermarked image with some angles. We rotate the watermarked image by 0.5^{o} and 1^{o} in clockwise and counter-clockwise directions, and the extracted watermarks are shown in Fig. 5. From these results, we can show the robustness of the robust watermark and the fragility of the fragile watermark to rotation operations.

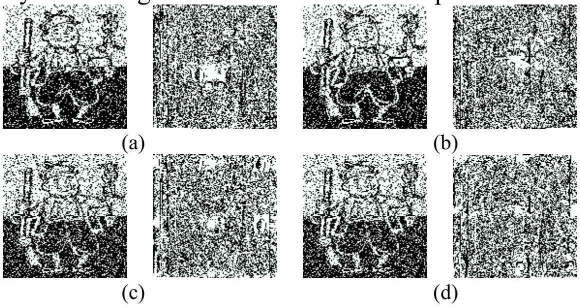

Figure 5. Watermarks extracted from rotated watermarked images. (a) Rotation by 0.5^{o} in the clockwise direction, robust NHS=0.698 and fragile NHS=0.129. (b) Rotation by 0.5^{o} in the counter-clockwise direction, robust NHS=0.692 and fragile NHS=0.136. (c) Rotation by 1^{o} in the clockwise direction, robust NHS=0.608 and fragile NHS=0.147. (d) Rotation by 1^{o} in the counter-clockwise direction, robust NHS=0.610 and fragile NHS=0.132.

5 Summary and Conclusion

An efficient multipurpose watermarking algorithm based on mean-removed VQ has been presented. Experimental results demonstrate that the proposed method can be used for copyright protection by extracting the robust watermark, and it can also be used for image authentication by extracting the fragile watermark.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under grant 60272074.

References

[1] J. J. K. O'Ruanaidh, W. J. Dowling, and F. M. Boland: "Watermarking digital images for copyright protection," *IEE Proceedings-Vision, Image and Signal Processing*, vol. 143, no.4, pp. 250–256, 1996.

[2] S. Pereira, and T. Pun: "An iterative template matching algorithm using the Chirp-Z transform for digital image watermarking," *Pattern Recognition*, vol. 33, no. 1, pp.173–175, 2000.

[3] Y. Wang, J. F. Doherty, and R. E. Van Dyck: "A wavelet-based watermarking algorithm for ownership verification of digital images," *IEEE Trans. Image Processing*, vol. 11, no. 2, pp. 77–88, 2002.

[4] D. Kundur, and D. Hatzinakos: "Digital watermarking for telltale tamper proofing and authentication," *Proceedings of the IEEE*, vol. 87, no. 7, pp.1167–1180, 1999.

[5] C. S. Lu, and H. Y. M. Liao: "Multipurpose watermarking for image authentication and protection," *IEEE Trans. Image Processing*, vol. 10, no. 10, 2001, pp. 1579–1592.

[6] C. S. Lu, H. Y. M. Liao, and L. H. Chen: "Multipurpose audio watermarking," in *Proc. 15th Int. Conf. Pattern Recognition*, vol. 3, pp. 282–285, 2000.

[7] 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.

[8] Z. M. Lu, J. S. Pan, and S. H. Sun: "VQ-based digital image watermarking method," *Electronics Letters*, vol. 36, no. 14, pp. 1201–1202, 2000.

[9] Z. M. Lu, C. H. Liu, and S. H. Sun: "Digital image watermarking technique based on block truncation coding with vector quantization," *Chinese Journal of Electronics*, vol. 11, no. 2, pp. 152–157, 2002.

[10] J. Minho, and K. HyoungDo: "A digital image watermarking scheme based on vector quantisation," *IEICE Trans. Information and systems*, vol. E85-D, no.6, 1054–1056, 2002.

[11] A. Makur, and S. S. Selvi: "Variable dimension vector quantization based image watermarking," *Signal Processing*, vol. 81, no.4, pp. 889–893, 2001.

[12] H. C. Huang, F. H. Wang, and J. S. Pan: "A VQ-based robust multi-watermarking algorithm," *IEICE Trans. Fundamentals*, vol. E85-A, no. 7, pp. 1719–1726, 2002.

[13] H. C. Huang, F. H. Wang, and J. S. Pan: "Efficient and robust watermarking algorithm with vector quantisation," *Electronics Letters*, vol. 37, no. 13, pp. 826–828, 2001.

[14] Y. Linde, A. Buzo, and R. M. Gray: "An algorithm for vector quantizer design," *IEEE Trans. Communications*, vol. 28, no.1, pp. 84–95, 1980.