2006 IEEE International Conference on Systems, Man, and Cybernetics
October 8-11, 2006, Taipei, Taiwan

A Watermarking Scheme Based on SVM and Tolerable Position Map

Shwu-Huey Yen, Chia-Jen Wang, and Yang-Ta Kao

Abstract—**This paper presents a novel digital watermarking technique based on Support Vector Machines (SVMs) and Tolerable Position Map (TPM). The purpose of SVMs is two folds in this study. One is using SVM to identify tolerable positions for watermark embedding on the host image, and the other is using SVM to embed and extract watermarks. By simulating common image attacks on the host image, pixels which are invincible or vulnerable are identified and used for positive or negative samples for training an SVM. Apply this SVM can create a TPM for the host image. To embed and extract watermarks, we use a known binary sequence to train an SVM such that this SVM can be applied for embedding and extracting the watermark. In the proposed scheme, to improve robustness of attacks and image imperceptibility, the watermark is embedded according to the TPM and by asymmetrically tuning blue channels of the central and neighbor pixels. To further reducing extraction errors, the embedded watermark bits are re-modified if necessary according to classifying result of the trained SVM. Our scheme uses only 128 bits in training both SVMs, thus it is time efficient. Experiments show that the proposed scheme provides high PSNR of a watermarked image, low extraction error rate, and extremely robust to common image attacks.**

1. INTRODUCTION

The emergence of digital imaging and networks has made manipulation and transmission of original artwork very convenient. However, unrestricted copying and media manipulation cause considerable financial loss and become an issue of intellectual property rights. In order to solve this problem, digital watermarking techniques have become an active research area. In general, digital watermarking is a practice of embedding a digital signal, called a watermark, into a host media. The embedded watermark can be detected or extracted later for verifying the ownership.

There are two common approaches of performing watermarking: one in spatial domain, and the other in transformed domain. Each technique has its own advantages and disadvantages. In the spatial domain, the watermark is embedded into the host image by directly modifying the pixel value of the host image. The main advantage of the spatial domain watermarking schemes is that less computational cost is required. On the other hand, transformed domain watermarking schemes perform the domain transformation procedure by transformation functions such as Discrete Wavelet Transformation (DWT), Discrete Cosine Transformation (DCT), Discrete Fourier Transformation (DFT), etc. Then, the transformed frequency coefficients are modified to embed watermark bits. Finally, the inverse transformation function of the specific one used in the forward transformation procedure is performed. The main advantage of the frequency domain watermarking schemes is that they are more robust than the spatial domain schemes. However, they generally consume more computational cost because additional forward transformation and inverse transformation must be performed.

There are quite a few image watermarking researches in transformed domains because of the robustness consideration. In [3], DWT transformation is first applied to the host image, then selected wavelet coefficients from subbands LL_1 and HH_1 are modulated respectively. At the same time, the JND value is computed for each selected coefficient to provide the maximum strength of transparency for the watermark. Ni *et al.* [4] introduce another transform domain scheme. In this scheme, the fractal dimension and parameter α are first calculated to determine whether a given block is a feature block. The watermark is embedded into the medium frequency coefficients of feature blocks in zigzag scanning order after DCT transformation is performed. There are also researches [5, 13] using genetic algorithms (GA) to train frequencies on the transformed domain to embed watermarks. By this way, the image quality of the watermarked image usually can be preserved well. However many researches demonstrated that such approach provides not enough capacity.

There are also several spatial domain watermarking schemes have been proposed. The color quantization scheme [1], proposed by Tsai *et al.*, introduces an approach for image watermarking by modifying the color index table. When the pixel mapping procedure for color quantization is performed, the watermark is embedded at the same time. But, to enhance the robustness of the scheme, the distribution of colors in the palette of host image must be uniform. Wu *et al.* [2] consider human visual effects to adaptively adjust the embedding watermark bits. The number of watermark bits for embedding in this scheme is determined by the visual effect of the pixel values in the host image. Kutter *et al.* [14] propose a method based on amplitude modulation. In their method, robustness is improved by multiply embedding a watermark and adaptive threshold for extracting from two reference

S. H. Yen is with the PRIA Lab., Department of Computer Science and Information Engineering, Tamkang University, Tamsui, Taipei, Taiwan (Tel: 886-2-26215656, ext. 3305; fax: 886-2-26209749; e-mail: shyen@cs.tku.edu.tw).

C. J. Wang is also with the PRIA Lab., Department of CS & IE, Tamkang University (e-mail: 121718@mail.tku.edu.tw).

Y. T. Kao was with the PRIA Lab., Department of CS & IE, Tamkang University. (e-mail: ydkao@ms39.hinet.net).

watermark bits. The idea of amplitude modulation is further developed by combining SVM in [10]. Yu *et al.* [10] propose an SVM-based color image watermarking algorithm. The watermark bits and additional 1024 training bits are embedded in the blue channels of pixels. For extraction phase, the 1024 embedded training bits are employed as training samples of the SVM. When the SVM is trained, it is used for extracting the watermark. Our method of embedding and extracting watermarks based on SVM is inspired by [10], a comparison and discussion is given in Section 4.

The rest of this paper is organized as follows. In Section 2, the fundamental of SVM will be introduced. In Section 3, concepts of TPM and the improved watermarking embedding and extraction method are given. The experimental results of the proposed scheme are shown in Section 4. Finally, conclusions will be given in Section 5.

2. SUPPORT VECTOR MACHINES (SVMs)

Support vector machine (SVM) is a statistical classification method proposed by Vapnik in 1995 [12]. Given a labeled training set:

$$S=\{(x_i,y_i)\mid x_i\in R^n, y_i\in\{-1,1\}, i=1\ldots m\} \tag{1}$$

where x_i stands for input vector i and y_i is the desired category, positive or negative, SVM can generate a separation hyperplane **H** that separates the positive and negative examples. Since SVM has the maximum generalization ability to separate data into two classes, thus it is naturally suitable for detecting a given bit to be zero or one (watermark bit). If any point $\boldsymbol{x}$ lies on a hyperplane, $\boldsymbol{x}$ must satisfy $w\cdot x+b=0$, where w is normal to this hyperplane and b is the bias. Finally, the optimal hyperplane $\mathrm{H}: w_0\cdot x+b_0=0$ can be determined by

$$w_0=\sum_{i=1}^{m}\alpha_i y_i x_i \tag{2}$$

where α_i and b_0 are Lagrange multipliers and the bias determined by SVM's training algorithm. In Eq. (2), those points x_i with $\alpha_i = 0$ can be ignored and those with $\alpha_i > 0$ are called "support vectors." After completion of training SVM, **H** is thus determined, any data x is classified according to the sign of the decision function. The decision function is defined as:

$$d(x)=\mathrm{sgn}\left(\sum_{i=1}^{m}\alpha_i y_i K(x_i,x)+b_0\right) \tag{3}$$

where $K(x_i,x)$ is the kernel function which maps the training or testing samples to a higher dimensional feature space as shown in Fig. 1. Three kinds of kernel functions are commonly adopted in SVMs as indicated in Table 1. In general, a RBF network is preferred to train the classifier, because it is more powerful and more efficacious than Polynomial and Two-layer [11].

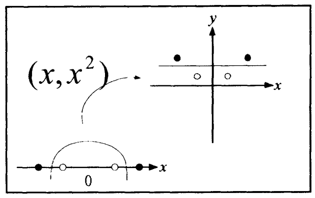


Fig. 1 A mapping function from input space to feature space.

Table 1 Common Kernel Functions

$K(x_i,x)=((x_i\cdot x)+1)^p$	Polynomial learning machine
$K(x_i,x)=\exp(-\lVert x_i-x\rVert^2/2\sigma^2)$	Radial-basis function network
$K(x_i,x)=\tanh(\kappa(x_i\cdot x)+\delta)$	Two-layer perception

3. THE PROPOSED SCHEME

The details of the scheme including TPM construction, watermarks embedding and extracting will be explained in the following.

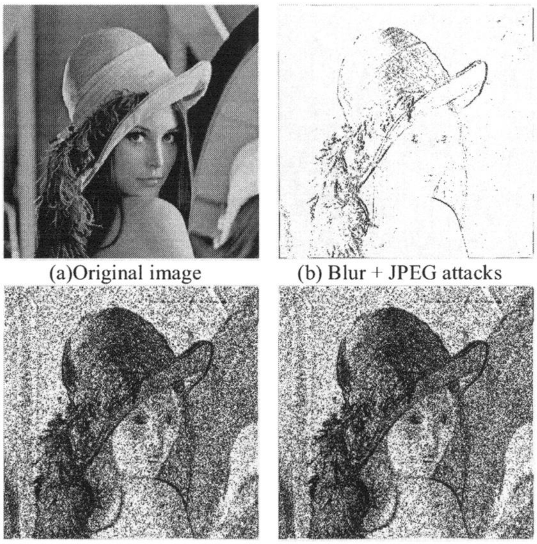
3.1 Tolerable Position Map (TPM)

To promote the robustness of watermarks, the candidate positions for embedding are selected carefully. We first simulate the common image attacks on the host image. Comparing the original image and the simulated image, pixels are identified as invincible or vulnerable to attacks. Using the abovementioned pixels as positive and negative samples respectively, an SVM is trained to obtain a tolerable position map (TPM) for the host image. The detailed procedure of the TPM is described in the following.

- Apply image attacks or processing to the host image, and call it the simulated image.
- Evaluate hue values on all pixels of the host image and simulated image by Eq. (4). The hue differences on pixel-wise position between these two images are calculated and sorted. Note that any position which has small hue difference indicating this pixel is invincible to the image attacks and vulnerable otherwise.

$$H=\cos^{-1}\left[\frac{1/2[(R-G)+(R-B)]}{\sqrt{(R-G)^2+(R-B)(G-B)}}\right] \tag{4}$$

- Training the SVM. Positive samples are selected randomly from pixels with small hue differences, and negative samples are selected from pixels with large hue differences. Each sample together with features in Eq. (10 -13) is used for training the SVM. In our experiment, 64 positive and 64 negative samples are used.
- Use the trained SVM to find out pixels robust to image attacks in the host image. The image consisting of all these pixels is called the tolerable position map (TPM). Fig. 2 shows various TPMs under different attacks. White pixels indicate positions that are suitable for watermark embedding and black pixels indicate vulnerable positions that are not suitable for watermark embedding.



(a)Original image (b) Blur + JPEG attacks

(c) Blur + Sharpen attacks (d) Sharpen + JPEG attacks

Fig. 2 Tolerable Position Maps.

3.2 Embedding algorithm

On behalf of training the second SVM for embedding and extracting watermarks, an additional binary sequence is needed. We assume that the watermark $\boldsymbol{W}$ is binary and concatenated from two sequences T and S as $W = TS = t_0t_1t_2...t_{N-1}s_0s_1s_2...s_{M-1}$. The first binary sequence $T = t_0t_1t_2...t_{N-1}$ denotes the training information of length N which is generated by pseudo-random number

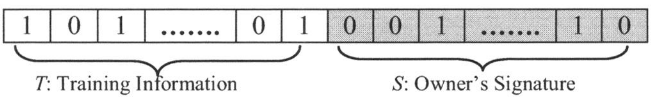


Fig. 3 A binary sequence of a watermark $\boldsymbol{W}$.

generator (PRNG) with seed$_1$. In our experimental, N is equal to 128. The sequence $S = s_0s_1s_2...s_{M-1}$ represents the owner's digital signature of length M. It may be a binary sequence or a binary image (logo). The binary sequence of watermark $\boldsymbol{W}$ is shown in Fig. 3.

1). Training information embedding: For security reason, a pseudo-random number generator (PRNG) is used to protect the information of these embedding positions. Assume N positions $P_i = (x_i, y_i)$ on the TPM of the host image are generated by PRNG with seed$_2$, and the embedding algorithm is described as followed.

For $i = 0$ to N-1

- Compute the luminance L_{P_i} at position P_i by

$$L_{P_i} = 0.299 R_{P_i} + 0.587 G_{P_i} + 0.114 B_{P_i} \tag{5}$$

where $R_{P_i}, G_{P_i}, B_{P_i}$ represent red, green and blue channel values of the pixel at P_i position. Based on the consideration that human eyes are less sensitive to changes in very dark and very bright regions, we have modified luminance factor as in Eq. (6).

$$L'_{p_i} = \begin{cases} L_{p_i} & L_{p_i} \geq 128 \\ 255 - L_{p_i} & \text{otherwise} \end{cases} \tag{6}$$

- The training information bit t_i is embedded into the host image by modifying the blue channels B_{P_i} and $B_{P_i'}$, at positions P_i and its 4-neighbors P_i', according to the luminance L'_{p_i} . The formula is shown in Eq. (7).

$$\begin{cases} B_{P_i} = B_{P_i} + \alpha_1(2t_i - 1)L'_{p_i} \\ B_{P_i'} = B_{P_i'} - \alpha_2(2t_i - 1)L'_{p_i} \end{cases} \tag{7}$$

where α_1, α_2 are positive constants that determine the watermark strength. In our study, α_1, α_2 are set to be 0.15 and 0.05 respectively.

Observe that when position P_i is selected, its surroundings should not be selected again. If any of surrounding position is selected, the value of B_{P_i} or $B_{P_i'}$ will be re-modified and cause the inaccuracy in the retrieval phase. Fig. 4 shows $P_i = (x_i, y_i)$ (the center one), its 4-neighbors P_i', with corresponding watermark strength constants, and the surroundings of P_i (12 colored positions as indicated). After training information T is embedded, an SVM will be trained by the set of training features F which is defined in (8).

$$F = \{TF_i = (V_i, d_i), i = 0 \ldots N-1\} \tag{8}$$

where TF_i is the training feature obtained from bit t_i, and d_i represents the class type of t_i, 0 or 1. Inspired by [10], V_i is defined as in (9).

$$V_i = (\delta^1_{P_i}, \delta^2_{P_i}, \delta^3_{P_i}, \delta^4_{P_i}) \tag{9}$$

The feature values $\delta^k_{P_i}$, defined in Eq. (10) – (13), are the difference between the blue channels of the central pixel and the average from corresponding neighbors. When training procedure for an SVM is completed, an optimal hyperplane can be determined. Now this trained SVM will be used to embed the owner's signature S.

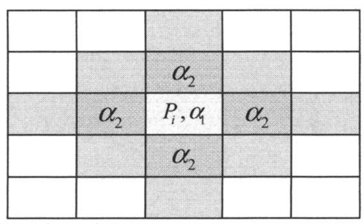


Fig. 4 $P_i = (x_i, y_i)$, its 4-neighbors, and the 12 corresponding surroundings.

$$\delta^1_{P_i} = B_{P_i} - \frac{1}{8}\left(\sum_{k=-1}^{1}\sum_{j=-1}^{1} B_{(x_{i+k}, y_{i+j})} - B_{(x_i, y_i)}\right) \tag{10}$$

$$\delta_{P_i}^2 = B_{P_i} - \frac{1}{8}\left(\sum_{k=-2}^{2} B_{(x_{i+k},y_i)} + \sum_{j=-2}^{2} B_{(x_i,y_{i+j})} - 2B_{(x_i,y_i)}\right) \tag{11}$$

$$\delta_{P_i}^3 = B_{P_i} - \frac{1}{8}\left(\sum_{k=-2}^{2} B_{(x_{i+k},y_{i+k})} + \sum_{j=-2}^{2} B_{(x_{i-j},y_{i+j})} - 2B_{(x_i,y_i)}\right) \tag{12}$$

$$\delta_{P_i}^4 = B_{P_i} - \frac{1}{8}\left(\sum_{k=-2}^{0}\sum_{j=0}^{2} B_{(x_i+k+j,y_i-k-2+j)} - B_{(x_i,y_i)}\right) \tag{13}$$

2). Owner's signature embedding: The overview of the owner's signature embedding is shown in Fig. 5. Similarly, M positions are generated on the TPM by PRNG with seed$_3$. For each position P_i, the feature vector V_i is constructed according to Eq. (9). The signature embedding algorithm is described as follows.

For $i = 0$ to M-1

- Classify the feature vector V_i by the trained SVM and let $s_i' = d(V_i)$ be the retrieved sign from the decision function in Eq. (3) such that

$$s_i' = \begin{cases} 0, & \text{if } d(V_{N+i}) < 0 \\ 1, & \text{if } d(V_{N+i}) \geq 0 \end{cases}. \tag{14}$$

- Compare the signature bit s_i with s_i':
  - $s_i = s_i'$: reinforce the blue channel value of the central pixel only, i.e., change B_{P_i} of the pixel at position P_i according to Eq. (15), again here α_1 is set to be 0.15.

$$B_{P_i} = B_{P_i} + \alpha_1(2t_i - 1)L_{p_i}' \tag{15}$$

  - $s_i' \neq s_i$: not only reinforce the blue channel value of the central pixel, but also modify the 4-neighbors', i.e., change B_{P_i} and $B_{P_i'}$, at position P_i and its 4-neighbors P_i' as in Eq. (16) where α_1, α_2 are set to be 0.15 and 0.05 respectively.

$$\begin{cases} B_{P_i} = B_{P_i} + \alpha_1(2t_i - 1)L_{p_i}' \\ B_{P_i'} = B_{P_i'} - \alpha_2(2t_i - 1)L_{p_i}' \end{cases} \tag{16}$$

After modification, V_i will be recalculated and classified again. To strike a balance between robustness and imperceptibility, Eq. (16) will be repeated one more time if the retrieved sign s_i' is still not the same with the signature bit s_i.

3.3 Extraction algorithm

When the recipient receives the watermarked image, he must train the SVM first then use this SVM to extract the signature. At same time, some secret information is needed: seed$_1$, seed$_2$, and seed$_3$. Additional information TPM is also needed, because the embedding positions are obtained by PRNG on the TPM.

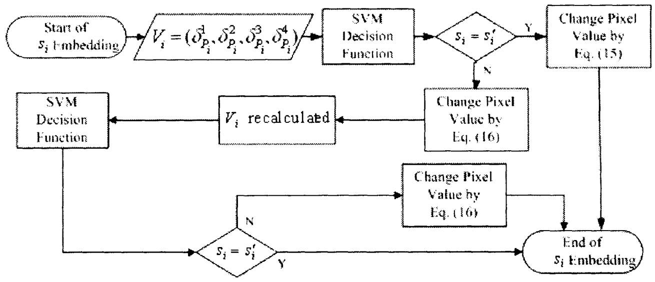


Fig. 5 Owner's signature embedding procedure.

1). SVM training for signature extraction: In order to train the SVM for extraction we need to recover training information T and corresponding embedding positions. They are determined by PRNG with the seed$_1$ and seed$_2$ as well as the TPM, the training feature vectors are evaluated from watermarked image. Similar to Eq. (8), the training feature set F' is defined as:

$$F' = \{TF_i' = (V_i', d_i), i = 0 \ldots N-1\} \tag{17}$$

All training features TF_i' are used to train a new SVM. When the training procedure is completed, this trained SVM will be used to extract the owner's signature S.

2). Signature extraction: Likewise, PRNG with seed$_3$ is used to generate M positions on the TPM and feature vectors V_{N+i}' are calculated from the watermarked image for $i = 0 \ldots M-1$. The decision function $d(V_{N+i}')$, define in (3), can be used to determine the signature bits s_i'' as in Eq. (14).

4. EXPERIMENTAL RESULTS

These experiments are implemented in an environment of Intel Centrino-Mobile 1.4GHz CPU, 35G HDD, 640M RAM, and Microsoft Windows XP. In the following experiments, a set of color images of 512×512 - *Lena*, *Peppers*, *Baboon*, *Airplane*, *Sailboat*, *House*- are used for host images as shown in Fig. 6(a-f). The binary image, *Rose*, of 64×64 is used as the watermark shown in Fig. 6(g). The TPM is obtained from a host image by blurring and JPEG compression processing. The RBF kernel is employed with σ= 0.2 in training SVMs. We compare the proposed scheme with the schemes proposed by Yu *et al.* [10] and Kutter *et al.* [14]. As in [10], 1024 training samples are used in training SVMs which takes a lot of computation time in the extraction phase. To save time, the training sample size of [10] is reduced to 512 in the experiments. Moreover, unlike our proposed scheme, the owner's signals are embedded into the pixels without re-modification from the classification result of the SVM, so the embedding time in [10] is recorded as zero. Similarly, in [14], there is no training time consumed in either embedding or extracting phases, but a watermark is embedded three times in the proposed method to enhance the accuracy of the extracted watermark. In producing the TPM for a given host image, which only applies to our method, it takes averagely 10 seconds to complete the task including a training of SVM and use it to classify every pixel on the host image. With emphasizing on watermarks embedding and extracting, we did not add this additional time into comparison. To compare the performances, experiments are measured by mean square error (MSE), image quality (PSNR), embedding and extracting time and error rate. The MSE and PSNR values between original and watermarked images are calculated by (18) and (19) respectively where H and W represent the height and width of the image. The error rate of the extracted watermark is defined in (20) with the length of the watermark to be M, s_i and s_i'' to be the ith bit of the original and extracted watermark.

$$MSE = \frac{\sum_{i=0}^{H-1}\sum_{j=0}^{W-1}\left((R_{ij}-\overline{R}_{ij})^2+(G_{ij}-\overline{G}_{ij})^2+(B_{ij}-\overline{B}_{ij})^2\right)}{3\times H\times W} \quad (18)$$

$$PSNR = 10\,log_{10}\frac{255^2}{MSE} \quad (19)$$

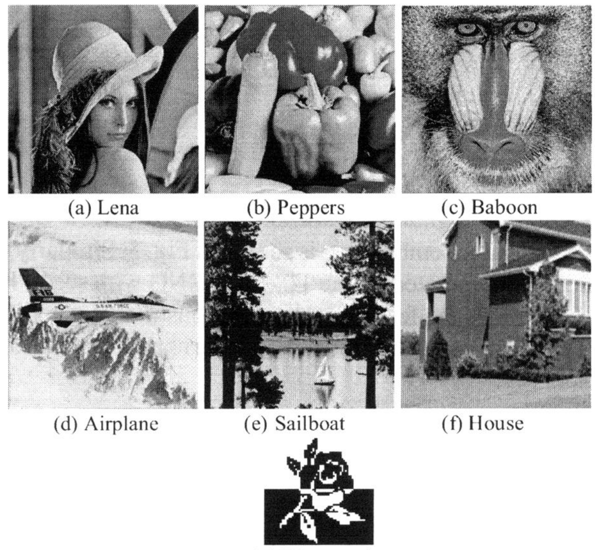
$$Error\ Rate = \frac{\sum_{i=0}^{M-1}|\,s_i - s_i''\,|}{M} \quad (20)$$

Experiments without attacks are summarized in Tables 2 to 7. In images *Baboon*, *Airplane*, and *House*, the extraction phase in [10] is not completed due to the long extraction time. Since there are [5(number of watermark bits) + 128] versus [3(number of watermark bits)] many pixels are affected in our method and the Kutter's method respectively, some PSNR values of watermarked images in [14] are outperformed ours method. We also observe that the execution time in the proposed method, although it has additional embedding time, is far less than in the method [10]. The proposed method has almost zero extraction error rates which apparently superior to the other two methods.

To test the robustness of the proposed method against image attacks, table 8 shows error rates of attacks including blurring, sharpening, JPEG compression and adding noises for our method. These attacks are performed by Photoshop 6.0 with JPEG quality factor is set to 12 and 5% uniform noises. As it shows the proposed method performs satisfactorily especially in the attacks of sharpening and JPG compression.

5. CONCLUSIONS

In this paper, a new image watermarking scheme based on Support Vector Machines and Tolerable Position Map is proposed. The TPM of an image is to predict positions that are immune to image attacks. With additional averagely 10 seconds to train an SVM for producing the TPM in latter watermarks embedding, the robustness to image attacks, watermarked images quality and extraction error rate are much improved. By training another SVM with supplementary training bits, an optimal hyperplane is obtained and used for embedding and extracting watermarks.



(a) Lena (b) Peppers (c) Baboon

(d) Airplane (e) Sailboat (f) House

(g) Watermark

Fig. 6 Host images and the watermark.

Table 2 Lena

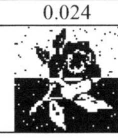
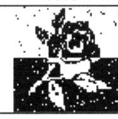
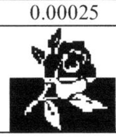
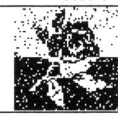
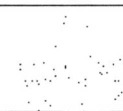
	Our method	Yu *et al.*[10]	Kutter *et al.*[14]
MSE	2.8093	2.852	2.778
PSNR	43.6448	43.578	43.693
Embedding time (sec)	1	0	0
Extraction time(sec)	1	248	0
Error rate	0	0.024	0.034
Extracted watermark			

Table 3 Peppers

	Our method	Yu *et al.*[10]	Kutter *et al.*[14]
MSE	2.7614	1.462	2.693
PSNR	43.7195	46.48	43.827
Embedding time (sec)	1	0	0
Extraction time(sec)	0	882	0
Error rate	0.00025	0.096	0.531
Extracted watermark			

To further improve the robustness and imperceptibility, the proposed scheme modifies blue channels asymmetrically, i.e. tunes the central and surrounding pixels at same time, to embed watermark bits, and possibly re-modified if necessary according to classifying result of the trained SVM. Overall, we conclude that reasonable embedding and extraction time, higher image quality, lower extraction error rate are achieved in the proposed method.

Table 4 Baboon

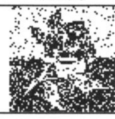
	Our method	Yu *et al.*[10]	Kutter *et al.*[14]
MSE	2.8156	2.82	2.890
PSNR	43.6352	43.627	43.521
Embedding time (sec)	24	0	0
Extraction time(sec)	30	More than 10 hours	0
Error rate	0.011963	N/A	0.178
Extracted watermark		Not Available	

Table 5 Airplane

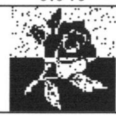
	Our method	Yu *et al.*[10]	Kutter *et al.*[14]
MSE	3.5637	32.336	5.270
PSNR	42.6117	33.033	40.912
Embedding time (sec)	1	0	0
Extraction time(sec)	1	More than 5 hours	0
Error rate	0.000244	N/A	0.040
Extracted watermark		Not Available	

Table 6 Sailboat

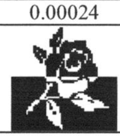
	Our method	Yu *et al.*[10]	Kutter *et al.*[14]
MSE	3.6634	4.381	3.1158
PSNR	42.492	41.715	43.1951
Embedding time (sec)	1	0	0
Extraction time(sec)	1	More than 5 hours	0
Error rate	0.00024	N/A	0.0834961
Extracted watermark		Not Available	

Table 7 House

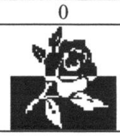
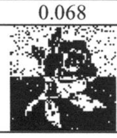
	Our method	Yu *et al.*[10]	Kutter *et al.*[14]
MSE	13.9797	18.246	14.099
PSNR	36.6758	35.519	36.638
Embedding time (sec)	1	0	0
Extraction time(sec)	1	More than 5 hours	0
Error rate	0	N/A	0.068
Extracted watermark		Not Available	

Table 8 Error rate results on different attacks

Attacks \ Images	Lena	Peppers	Baboon	Airplane	Sailboat	House
Blur	0.0166	0.0151	0.1103	0.0083	0.0080	0.0268
Sharpen	0	0.0004	0.0148	0	0.0002	0
JPG	0	0.0002	0.0119	0.0002	0.0002	0
Noise	0.0410	0.0427	0.0834	0.0310	0.0363	0.1137

The proposed method could be further studied in several ways. First, create an optimal TPM for an image. In this paper, TPM is produced by blurring and JPEG compression, and the experiments show that it can extract watermarks perfectly especially in attacks of sharpening and JPEG compression. Moreover, 4 features in Eq. (10 – 13) are employed with equal weights as input for the SVM. In the future work, we will compare the effects on each feature as well as possible new features to have better result for watermark embedding and extracting. We will get more rigorous testing and analysis.

Reference

[1] Tsai, P., Hu, Y. C., Chang, C. C., "A color image watermarking scheme based on color quantization," *Signal Processing* 84, pp. 95-106 (2004).

[2] Wu, D. C., Tsai, W. H., "Embedding of any type of data in images based on a human visual model and multiple-based number conversion," *Pattern Recognition Letters* 20, pp. 1511-1517 (1999).

[3] Chen, L. H., Lin, J. J., "Mean quantization based image watermarking," *Image and Vision Computing* 21, pp. 717–727 (2003).

[4] Ni, R., Ruan, Q., Cheng, H. D., "Secure semi-blind watermarking based on iteration mapping and image features," *Pattern Recognition* 38, pp. 357-368 (2005).

[5] Shieh, C. S., Huang, H. C., Wang, F. H., Pan, J. S., "Genetic watermarking based on transform-domain techniques," *Pattern Recognition* 37, pp. 555-565 (2004).

[6] Burges, C. J. C., "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, Vol. 2, No. 2, pp. 121-167 (1998).

[7] Keerthi, S. S., Shevade, S. K., Bhattacharyya, C., Murthy, K. R. K., "Improvements to Platt's SMO Algorithm for SVM Classifier Design," *Neural Computation*, Vol. 13, pp. 637-649 (2001).

[8] Platt, J. C., "Fast training of support vector machines using sequential minimal optimization, Advances in Kernel Methods - Support Vector Learning," B. Schölkopf, C. Burges, and A. Smola, eds., *MIT Press*, pp. 185-208 (1999).

[9] Platt, J. C., Adult and Web Site. http://www.research.microsoft.com/~jplatt

[10] Yu, Pao-Ta, Tsai, Hung-Hsu, Sun, Duen-Wu, "Digital Watermarking of Color Images Using Support Vector Machines," *2003 National Computer Symposium (NCS'03)* (2003).

[11] Hsu, Chih-Wei, Chang, Chih-Chung and Lin, Chih-Jen, "A Practical Guide to Support Vector Classification," http://www.csie.ntu.edu.tw/~cjlin/papers/

[12] Vapnik, V. "The Nature of Statistical Learning Theory, Springer-Verlag," New York, (1995).

[13] Shih, Frank Y., Wu, Yi-Ta, "Enhancement of image watermark retrieval based on genetic algorithms," *Journal of visual communication & image representation* 16, pp. 115–133 (2005).

[14] Kutter M., Jordan F., Bossen F., "Digital Signature of Color Images using Amplitude Modulation," *Electronics Imaging* **7**(2), pp.326–332 (1997).

[15] Y. G. Fu, R. M. Shen, L. P. Shen, X. S. Lei, "Reliable Information Hiding Based on Support Vector Machine," *INFORMATICA*, 2005, Vol. 16, No. 3, pp.333–346.

[16] F. Y. Shih, Y. T .Wu, "Robust watermarking and compression for medical images based on genetic algorithms," *Information Sciences* Vol.175 pp. 200–216 (2005).

[17] M. Barni, F. Bartolini, A. Piva, "Improved Wavelet-Based Watermarking Through Pixel-Wise Masking," *IEEE TRANSACTIONS ON IMAGE PROCESSING*, VOL. 10, NO. 5, MAY 2001.