SVM Based Watermarking Technique

Shwu-Huey Yen* and Chia-Jen Wang

Department of Computer Science and Information Engineering, Tamkang University,
Tamsui, Taiwan 251, R.O.C.

Abstract

This paper presents a digital watermarking technique based on Support Vector Machines (SVMs). Use the nice characteristic of the SVM, which can result an optimal hyperplane for the given training samples, the imperceptibility and robustness requirements of watermarks are fulfilled and optimized. In the proposed scheme, to improve imperceptibility, the watermark is embedded by asymmetrically tuning blue channels of the central and surrounding pixels at the same time. Furthermore, to promote robustness, the embedded watermark bits will be re-modified if necessary according to classifying result of the trained SVM. Our scheme uses only 128 bits in training SVM, thus it is time efficient. Watermarks are embedded in spatial domain and extracted directly from a watermarked image without the requirement of original image. Experiments show that the proposed scheme provides high PSNR of a watermarked images and low extraction error rate.

Key Words: Digital Watermarking, Support Vector Machines (SVMs)

1. Introduction

Digital multimedia products, such like text, images, video, audio, etc., have revolutionized in the way of extensively manipulated and transmitted in everyday of our lives. With the help of convenient ways of storing, manipulating, and accessing these data have brought lots of benefits into the digital multimedia field. 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 using 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 the 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.

In related research, several spatial domain watermarking schemes have been proposed. The color quantization [1] scheme, 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 em-

*Corresponding author. E-mail: shyen@cs.tku.edu.tw

bedded 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.* [3] 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 watermark bits. The idea of amplitude modulation is further developed by combining SVM in [4]. Yu *et al.* [4] 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.

Recently, the image watermarking research has moved toward embedding the watermark in transformed coefficients because of the robustness consideration. In [5], DWT transformation is first applied to the domain transformation procedure. Then modulate the wavelet coefficients selected from subbands LL_1 and HH_1, respectively. At the same time, the JND value is computed for each selected coefficient to provide the maximum strength transparent watermark. Ni *et al.* [6] introduce another transform domain scheme. In this scheme, the fractal dimension and parameter α are first calculated for each block. Then feature blocks or non-feature blocks are determined. Finally, 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 [7,8] using genetic algorithms (GA) to train the frequencies on the transformed domain to embed the watermark. By this way, the image quality of the watermarked image usually can be preserved very well.

The rest of this paper is organized as follows. In Section 2, the fundamental of support vector machine will be introduced. In Section 3, a new watermarking scheme based on SVM technique will be introduced. The experimental results of the proposed scheme are shown in Section 4. Finally, the 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 [9]. Given a labeled training set:

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

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 $\boldsymbol{x}$ which lies on the hyperplane must satisfy $w \cdot x + b = 0$, where $\boldsymbol{w}$ is normal to the hyperplane and b is the bias. Finally, the optimal hyperplane H: $w_0 \cdot x + b_0 = 0$ can be determined by

$$w_0 = \sum_{i=1}^{m} \alpha_i y_i x_i \quad (2)$$

where α_i and b_0 are Lagrange multipliers and bias that 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 the training of SVM is completed, **H** is thus determined, then any data x will be classified according to the sign of the decision function. The decision function is defined as:

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

where $K(x_i, x)$ is the kernel function which maps the training samples to a higher dimensional feature space as shown in Figure 1.

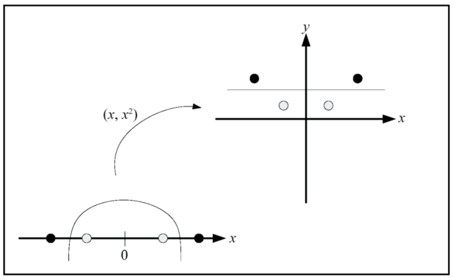


Figure 1. A mapping function from input space to feature space.

Three kinds of kernel functions are commonly adopted in SVM as indicated in Table 1. In general, the RBF network is preferred to train the classifier, because it is more powerful and more efficacious than Polynomial and Two-layer [10].

3. The Proposed Scheme

The overview of the proposed watermarking scheme is shown in Figure 2. The details of the scheme will be described in the following.

3.1 Embedding Algorithm

On behalf of training SVM, the watermark is extended and embedded into the original image. We assume that the watermark $\boldsymbol{W}$ is binary and consists of two sequences T and S as: $W = TS = t_0t_1t_2\ldots t_{N-1}s_0s_1s_2\ldots s_{M-1}$. The first binary sequence $T = t_0t_1t_2\ldots t_{N-1}$ denotes the training information of length N which is generated by pseudo-random number generator (PRNG) with seed$_1$. In our experimental, N is equal to 128. The sequence $S = s_0s_1s_2\ldots s_{M-1}$ 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 is shown in Figure 3.

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

3.1.1 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 host image are generated by PRNG with seed$_2$, and the embedding algorithm can be described as follows.

For $i = 0$ to $N-1$

1.Compute the luminance L_{P_i} at position P_i by the follow equation:

$$L_{P_i} = 0.299R_{P_i} + 0.587G_{P_i} + 0.114B_{P_i} \qquad (4)$$

$R_{P_i}, G_{P_i}, B_{P_i}$ represent red, green and blue channel values of the pixel at P_i position.

2.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 (5).

$$\begin{cases} B_{P_i} = B_{P_i} + \alpha(2t_i - 1)L_{P_i} & \alpha = 0.15, \\ B'_{P_i} = B'_{P_i} - \alpha(2t_i - 1)L_{P_i} & \alpha = 0.05. \end{cases} \qquad (5)$$

where α is a positive constant that determines the watermark strength, as indicated in Figure 4.

Observe that when position P_i is selected, its surroundings should not be selected again. If any of surrounding position is selected, then the value of $B_{P_i}^{'}$ will

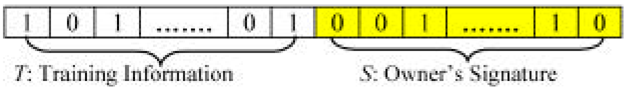


Figure 3. A binary sequence of a watermark.

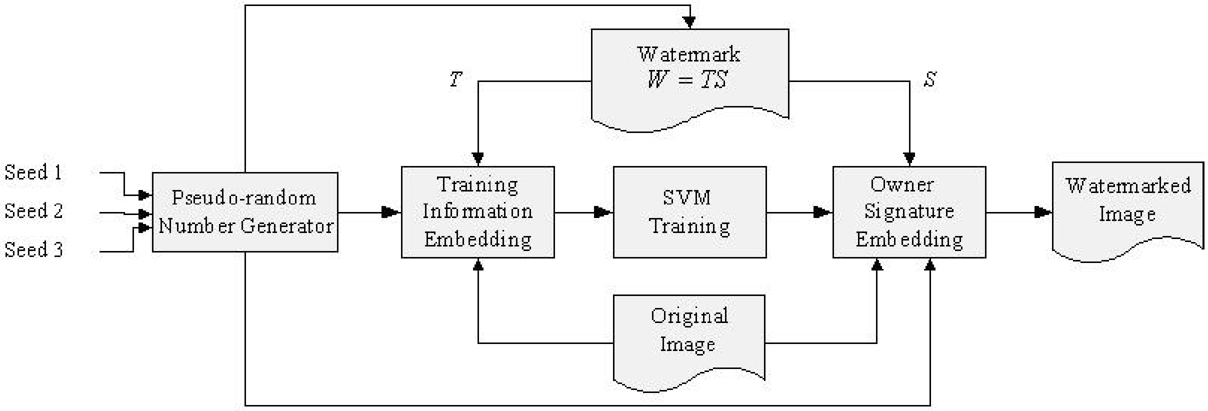


Figure 2. The diagram of embedding system.

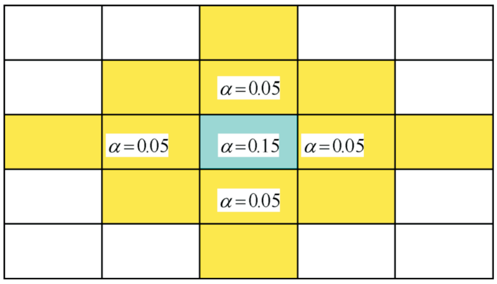


Figure 4. P_i, its 4-neighbors, and the corresponding surroundings.

be re-modified that may cause the inaccuracy in the retrieval phase. Figure 4 shows P_i (the center one), its 4-neighbors, and the corresponding surroundings (12 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 (6).

$$F=\{\ TF_i=(V_i,d_i),\ i=0\ldots N-1\} \tag{6}$$

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 [4], V_i is defined as in (7).

$$V_i=(\delta^1_{P_i},\delta^2_{P_i},\delta^3_{P_i}) \tag{7}$$

The feature value $\delta^k_{P_i}$ is the difference between the blue channels of the central pixel and the average from its neighbors. The feature values $\delta^k_{P_i}$ are described in Eq. (8–10) with the corresponding neighbors illustrated in Figures (5–7).

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

$$\delta^2_{P_i}=B_{P_i}-\frac{1}{8}(\sum_{k=-2}^{2}B_{(x_{i+k},y_i)}+\sum_{j=-2}^{2}B_{(x_i,y_{i+j})}-2B_{(x_i,y_i)}) \tag{9}$$

$$\delta^3_{P_i}=B_{P_i}-\frac{1}{8}(\sum_{k=-2}^{2}B_{(x_{i+k},y_{i+k})}+\sum_{j=-2}^{2}B_{(x_{i-j},y_{i+j})}-2B_{(x_i,y_i)}) \tag{10}$$

When training procedure for an SVM is completed, an optimal hyperplane with corresponding α_i and b_0 can be determined. Now the trained SVM will be used to embed the owner's signature.

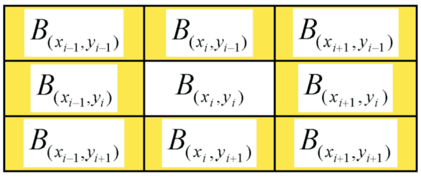


Figure 5. Mask for feature 1.

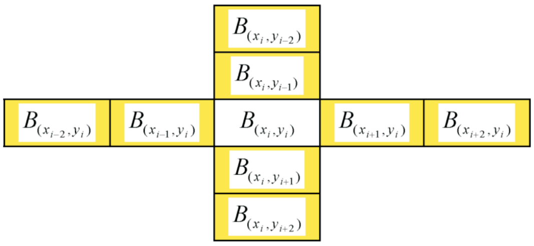


Figure 6. Mask for feature 2.

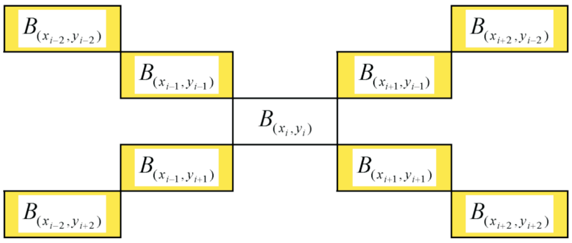


Figure 7. Mask for feature 3.

3.1.2 Owner's Signature Embedding

The overview of the owner's signature embedding is shown in Figure 8. Similarly, M positions are generated by PRNG with seed$_3$. For each position P_i, the feature vector V_i is constructed according to Eq. (7). The signature embedding algorithm can be 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

$$s'_i=\begin{cases}0, & \text{if } d(V_{N+i})=-1\\ 1, & \text{if } d(V_{N+i})=\ \ 1\end{cases} \tag{11}$$

Compare the signature bit s_i with $s^{'}_i$:

- $s_i=s^{'}_i$: change the blue channel of the pixel at position P_i according to (12).

$$B_{P_i}=B_{P_i}+\alpha(2t_i-1)L_{p_i},\qquad \alpha=0.05 \tag{12}$$

- $s^{'}_i\neq s_i$: modify the blue channels, B_{P_i} and $B^{'}_{P_i}$, at position P_i and its 4-neighbors as in (13).

$$\begin{cases}B_{P_i}=B_{P_i}+\alpha(2t_i-1)L_{p_i}, & \alpha=0.15\\ B^{'}_{P_i}=B^{'}_{P_i}-\alpha(2t_i-1)L_{p_i} & \alpha=0.05\end{cases} \tag{13}$$

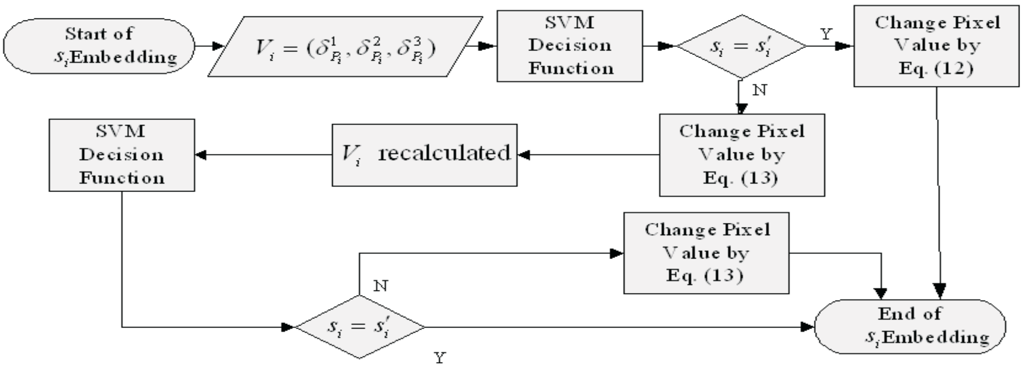

Start of s_iEmbedding
$V_i = (\delta^1_{P_i}, \delta^2_{P_i}, \delta^3_{P_i})$
SVM Decision Function
$s_i = s_i'$
Y
Change Pixel Value by Eq. (12)
N
Change Pixel Value by Eq. (13)
V_i recalculated
SVM Decision Function
Change Pixel Value by Eq. (13)
N
$s_i = s_i'$
Y
End of s_iEmbedding

Figure 8. Owner's signature embedding procedure.

After modification, V_i will be recalculated and classified again. To strike a balance between robustness and imperceptibility, Eq. (13) will be repeated one more time if the retrieved sign s_i' is still not the same with the signature bit s_i.

3.2 Extraction Algorithm

When the recipient receives the watermarked image, he must train the SVM first then use this SVM to extract the signature. The algorithm is described as follows.

3.2.1 SVM Training for Signature Extraction

The recovery of training information and corresponding embedding positions is the first step to train the SVM. After they are determined by PRNG with the seed$_1$ and seed$_2$, the training feature vectors are evaluated from watermarked image. Similar to Eq. (6), the training feature set F′ is defined as:

$$F' = \{TF_i' = (V_i', d_i),\ i = 0\ldots N-1\} \tag{14}$$

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.

3.2.2 Signature Extraction

Likewise, PRNG with seed$_3$ is used to generate M positions 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. (11) for $i = 0,\ldots, M-1$.

4. Experimental Results

These experiments are implemented in an environment using the Intel Centrino-Mobile 1.4 GHz 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", and "House" are used for host images as shown in Figure 9(a-e). The binary image, "Rose" of 64 × 64, is used as the watermark shown in Figure 9(f). The RBF kernel is employed with $\sigma = 0.2$ in SVM. We compare the proposed scheme with the schemes by Yu *et al.* [4] and kutter *et al.* [3]. As in Yu's scheme, 1024 training samples are used in obtaining the SVM which takes a lot of computation time in the extraction phase. Therefore, for the time sake, the training sample size of [4] is reduced to 512 in the experiment. 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 [4] is recorded as zero. Similarly, in Kutter's method [3], there is no train-

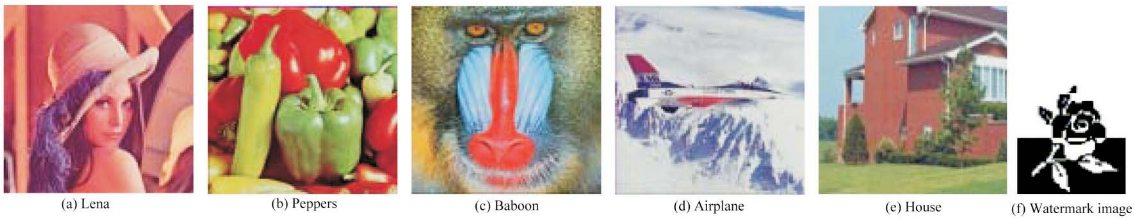

(a) Lena (b) Peppers (c) Baboon (d) Airplane (e) House (f) Watermark image

Figure 9. Host images and watermark image.

ing time consumed in either embedding or extracting phases. In addition, a watermark is embedded three times in Kutter's method to enhance the accuracy of the extracted watermark. 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 (15) and (16) respectively.

$$MSE = \frac{\sum_{i=0}^{H-1}\sum_{j=0}^{W-1}\left((R_{ij}-\bar{R}_{ij})^2+(G_{ij}-\bar{G}_{ij})^2+(B_{ij}-\bar{B}_{ij})^2\right)}{3\times H\times W} \tag{15}$$

$$PSNR = 10\log_{10}\frac{255^2}{MSE} \tag{16}$$

where H and W represent the height and width of the image. In general, if the PSNR value is greater than 35 dB then the perceptual quality is acceptable. The error rate of the extracted watermark is defined in (17) 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.

$$Error \quad Rate = \frac{\sum_{i=0}^{M-1}|s_i - s_i''|}{M} \tag{17}$$

Results are summarized in Tables 2 to 6. In images "Baboon", "Airplane", and "House", the extraction phase in [4] 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 [3] 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 [4]. Overall, we conclude that reasonable embedding and extraction time, higher image quality, lower extraction error rate are achieved in the proposed method.

In the following experiments, how the choice of σ in RBF kernel of the SVM affects the performance of the watermarking scheme is investigated. There are 2 ranges for σ being considered; $range_1$ is 2 to 50 steps by 2 and $range_2$ is 0.1 to 0.5 steps by 0.05. Experiments of different σ are implemented and values of Error Rate (in Figure 10), image quality (PSNR, shown in Figure 11) are observed. In Figure 10, for σ ranging from 2 to 50, the error rates do not vary too much except "Peppers". On the other hand, when σ ranging from 0.1 to 0.5 the fluctuations of the error rates are obvious for "F-16", "Baboon", and "House". In Figure 11, it demonstrates that the choice of σ does not have much impact on PSNR values. Combining the observations above, selecting σ to be 0.35 or 10 causes the best performance. Since the experi-

Table 2. Lena

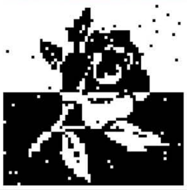
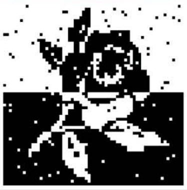
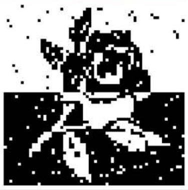
	Our method	Yu *et al.*[4]	kutter *et al.*[3]
MSE	1.817	2.852	2.778
PSNR	45.536	43.578	43.693
Embedding time (sec)	6	0	0
Extraction time(sec)	2	248	0
Error rate	0.013	0.024	0.034
Watermarked image			
Extracted watermark			

Table 3. Peppers

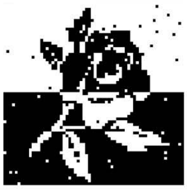
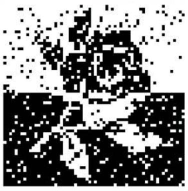
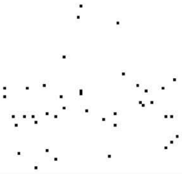
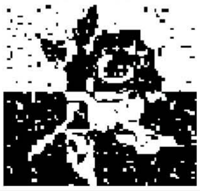
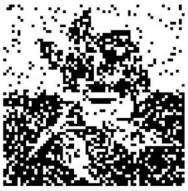
	Our method	Yu *et al.*[4]	kutter *et al.*[3]
MSE	1.79	1.462	2.693
PSNR	45.59	46.48	43.827
Embedding time (sec)	5.4	0	0
Extraction time(sec)	4.1	882	0
Error rate	0.01	0.096	0.531
Watermarked image			
Extracted watermark			

Table 4. Baboon

	Our method	Yu *et al.* [4]	kutter *et al.* [3]
MSE	3.223	2.82	2.890
PSNR	43.066	43.627	43.521
Embedding time (sec)	72	0	0
Extraction time(sec)	42	More than 10 hours	0
Error rate	0.09	N/A	0.178
Watermarked image			
Extracted watermark		Not Available	

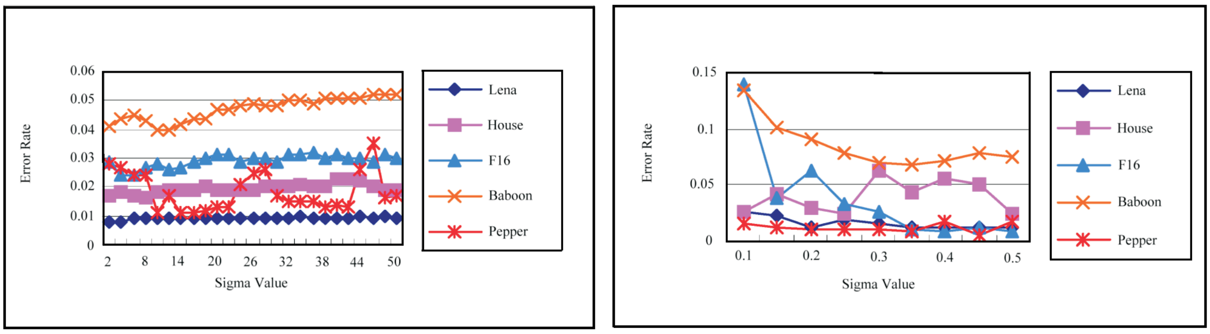


Figure 10. Error rate for experimental images.

Table 5. Airplane

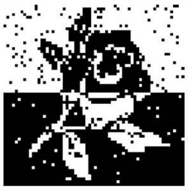
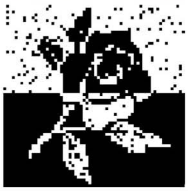
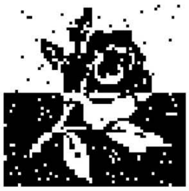
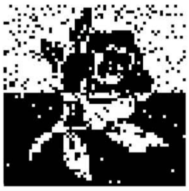
	Our method	Yu *et al.* [4]	kutter *et al.* [3]
MSE	14.6	32.336	5.270
PSNR	36.47	33.033	40.912
Embedding time (sec)	71	0	0
Extraction time(sec)	42	More than 5 hours	0
Error rate	0.062	N/A	0.040
Watermarked image			
Extracted watermark		Not Available	

Table 6. House

	Our method	Yu *et al.* [4]	kutter *et al.* [3]
MSE	7.775	18.246	14.099
PSNR	39.223	35.519	36.638
Embedding time (sec)	2	0	0
Extraction time(sec)	1	More than 5 hours	0
Error rate	0.029	N/A	0.068
Watermarked image			
Extracted watermark		Not Available	

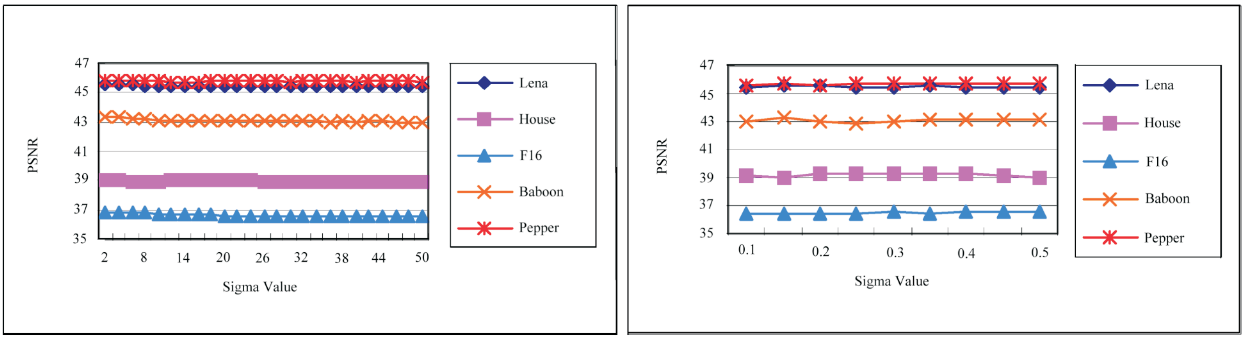


Figure 11. PSNR for experimental images.

ment for different images and different methods is time consuming, in addition, all experiments mentioned above are performed by σ chosen as 0.2 from the beginning, as it also gives low error rate and high image quality, thus we keep the results in this paper. In the future, we will enlarge our experiment and test a larger range for values of σ to have an even better result.

5. Conclusions

In this paper, a new image watermarking scheme based on Support Vector Machines is proposed. The SVM, with only additional 128 training bits, is employed to train an optimal hyperplane to classify the signature bits. The proposed scheme not only is efficient in computation but also applicable to general images.

To embed the watermark bits, the proposed scheme modifies blue channels of the central and surrounding pixels at same time. By asymmetrically tuning, both the robustness strength of the watermark and imperceptibility of the watermarked image can be preserved. To extract the watermark bits, the recipient only need to have the seed numbers and the watermarked image. Use PRNG and seed numbers, one can get the training bits as well as pixels where training bits and watermark bits are embedded. From training bits, SVM can be trained and use this SVM to extract the watermark bits. The value of parameter σ in RBF kernel is also studied. A set of tests is implemented to investigate the relationship between σ and extraction error rate, and image quality. We observe that different values of σ will result in different performances, and σ = 0.2 is an appropriate choice in the balance of extraction error rate and image quality.

Acknowledgements

This work was supported by National Science Council (Taiwan, ROC) under Grant No. NSC 93-2815-C-032-016-E.

Reference

[1] Tsai, P., Hu, Y. C. and Chang, C. C., "A Color Image Watermarking Scheme Based on Color Quantization," *Signal Processing* 84, pp. 95−106 (2004).

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

[4] Yu, P.-T., Tsai, H.-H. and Sun, D.-W., "Digital Watermarking of Color Images Using Support Vector Machines," *2003 National Computer Symposium (NCS'03)* (2003).

[5] Chen, L. H. and Lin, J. J., "Mean Quantization Based Image Watermarking," *Image and Vision Computing* 21, pp. 717−727 (2003).

[6] Ni, R., Ruan, Q. and Cheng, H. D., "Secure Semi-blind Watermarking Based on Iteration Mapping and Image Features," *Pattern Recognition* 38, pp. 357−368 (2005).

[7] Shieh, C. S., Huang, H. C., Wang, F. H. and Pan, J. S., "Genetic Watermarking Based on Transform-domain Techniques," *Pattern Recognition* 37, pp. 555−565 (2004).

[8] Shih, Frank Y. and Wu, Y.-T., "Enhancement of Image Watermark Retrieval Based on Genetic Algorithms," *Journal of Visual Communication & Image Representation* Vol. 16, pp. 115−133 (2005).

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

[10] Hsu, C.-W., Chang, C.-C. and Lin, C.-J., "A Practical Guide to Support Vector Classification," http:// www. Csie.ntu.edu.tw/~cjlin/papers/

[11] Burges, C. J. C., "A Tutorial on Support Vector Machines for Pattern Recognition," *Data Mining and Knowledge Discovery*, Vol. 2, pp. 121−167 (1998).

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

[13] 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).

Manuscript Received: Aug. 11, 2005
Accepted: Oct. 25, 2005