A TRACEABLE CONTENT-ADAPTIVE FINGERPRINTING FOR MULTIMEDIA

*Yu-Tzu Lin and Ja-Ling Wu, **Senior Member, IEEE***

Communication and Multimedia Laboratory, Department of Computer Science and Information Engineering
National Taiwan University, Taipei, Taiwan

ABSTRACT

This paper presents a feasible system of multimedia fingerprinting. A c-TA code which can be constructed flexibly is used. To ensure the applicability of both the maximum collusion size and the maximum number of consumers, the fingerprint length should be considerably increased. We design a content-adaptive watermarking scheme using neural networks to adaptively embed the long-length fingerprint without influencing the imperceptibility significantly. Experimental results show the high detection ratio of traitor tracing.

1. INTRODUCTION

A fingerprinting system embeds the unique key to the host data for each user. These keys can be used to identify the source of illegal copies. However, a coalition of users may collude to make an illegal copy, which is different from all colluders' copies. Collusion-resistant fingerprinting devotes to solve this problem. Boneh and Shaw initiate collusion-secure fingerprinting[5], which considers how to design the fingerprint codes to resist the coalition of c traitors, called frameproof codes. [2] proposed a traceability (TA) scheme. Combinatorial properties of frameproof codes and traceability codes then be derived in [6], in which a c-TA scheme is defined. [12] constructed an anti-collusion code by using Balanced Incomplete Block Design (BIBD). In [8], a collusion-secure fingerprinting scheme based on finite geometries is presented. The Reed-Solomon code is used to construct a traceable code in [1]. But it is still hard to find a systematic method to construct a c-TA code adaptable to various applications. The construction method of the c-TA code proposed in [10] is systematic and flexible compared to existing methods. We combine this c-TA code with a content-adaptive watermarking scheme to make this code applicable to multimedia. Human visual system (HVS) has been introduced in some watermarking schemes [3] to improve the quality of watermarked images by considering the trade-off between imperceptibility and robustness. We propose a content-adaptive watermarking using Self-Organizing Maps (SOM) to tune the watermark strength adaptively to ensure the embedded watermark is perceptually invisible and robust.

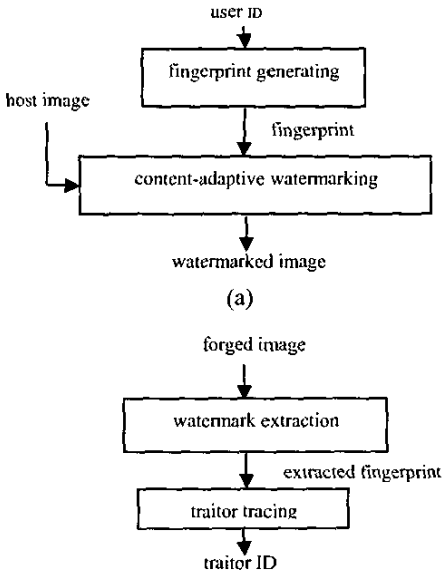

Fig. 1. System overview. (a) fingerprinting scheme. (b) traitor tracing scheme.

2. SYSTEM OVERVIEW

Fig. 1 shows our system architecture. Fig. 1 (a) illustrates the fingerprinting process, in which fingerprints are generated according to user IDs and embedded into the host image using adaptive image watermarking techniques, the produced watermarked images are then assigned to users individually. In the fingerprint generating step, a c-TA code is constructed from the error correcting code that satisfies a bound on their minimum distance. The content-adaptive watermarking scheme determines the maximum watermarking strength locally by the SOM neural network using Fourier transform coefficients. After watermark

0-7803-8554-3/04/$20.00 ©2004 IEEE.

embedding, each user acquires a particular watermarked copy with his particular fingerprint in it.

If some legal users collude to produce the pirate image, one of the colluders should be identified. We show the traitor tracing scheme in Fig. 1(b). First, the pirate code is extracted from the forged image. Then the traitor can be identified according to the extracted code from watermark extraction process. We simply compute the Hamming distance from the pirate code and all legal codes in the fingerprint database to locate one of the colluders.

3. TRAITOR TRACING

Our traitor tracing scheme is based on the sequential c-traceability code proposed in [10]. It is shown that a sequential c-TA scheme can be constructed from a q-ary error-correcting codes and theorem 6 in [11] says that mark allocation table of a sequential c-TA scheme is a c-TA code, thus we obtain a c-TA code from a q-ary ECC.

3.1. c-TA code construction

The c-TA code is constructed from finite elements in $GF(q^k)$. The codeword is the vector

$$(Tr(\alpha x_1^s + \beta), Tr(\alpha x_2^s + \beta), ..., Tr(\alpha x_{q^t}^s + \beta)) \quad (1)$$

where $\alpha \in GF(q^k)^*, \beta \in \{\beta_1, \beta_2, ..., \beta_q\}$ be q elements of $GF(q^k)$ whose trace values are pairwise distinct. The positive integer s satisfies

$k = 2t, s < q^{k/2+1}$, and $\exists r$ such that $r|t, q^r = -1 \pmod{s}$

if $c < \frac{-1+\sqrt{1+4q^{3k/2}(q^{k/2}+(s-1)(q-1))}}{2q^{k/2-1}(q^{k/2}+(s-1)(q-1))}$

then the minimum distance $D \geq (1-1/c^2)L + 1/c$ [10] which satisfied the criterion of Theorem 3.1.

Theorem 3.1 ([11]) Let c be an integer, Γ denote a mark allocation table obtained from an $(L,N,D)_q$-ECC satisfying

$$D \geq (1-1/c^2)L + 1/c$$

and A is the tracing function. Then (Γ,A) is a sequential c-TA scheme.

The c-TA code vectors are then used as our fingerprints, different user IDs are mapped to different parameters (α, β, s), and so different codewords[10].

3.2. Analysis of the c-TA code

Consider the codeword vector in (1), we can derive that $L=q^k$ and $N=q(q^k-1)$, where L is the codeword length with alphabet size q and N is number of codewords. When q is large enough, the collusion size c is independent of k. So for a fixed q, we can increase N without influencing the level of security c. This property makes our code construction flexible in the number of users. This improves the drawback of Reed-Solomon codes in which increasing k will cause the reduction of c.

4. CONTENT-ADAPTIVE WATERMARKING

To achieve high PSNR when watermark embedding, a content-adaptive watermarking scheme is proposed. We establish a SOM neural network learning variant levels of tolerance of quality distortion for different multimedia-contents to simulate the human visual system. After the SOM neural net is trained, it learns the ability to adjust the watermark strength adaptively according to the content of the multimedia, so as to minimize the distortion of the content quality. In this paper, we focus on the image watermarking, this technology can be applied to other types of multimedia in a similar way.

4.1. SOM architecture

SOM neural network can imitate the mechanism of cerebral cortex. A 2-D map is used in our architecture to represent the cortex map involving the sensitivity of image distortion. During the training period of our network, each unit with a positive activity within the neighborhood of the winning unit updates its value toward the features of certain image appearances. After iterations of updating values of neurons by Kohonen's algorithm, nodes in the 2-D array organize themselves in response to the input vectors. The resulting map projects the input vectors onto 2-D map preserving the natural topology of training data, which contains the feature vectors of various types of images, including smooth, high-frequency, regular-texture images, etc..

Initially, a 2-D SOM with 4×4 neurons is constructed. Each node is randomly assigned to feature values of one of the training samples. The learning algorithm performs the weight-update process using Mexican-hat function. If one node is too coarse to represent certain category of training data, it is divided into 4 child-nodes. Further dividing should be proceeded if nodes after the first dividing are still too rough, and so on. The architecture of our SOM network is shown in Fig. 2. We express the feature vectors of images, which are obtained from performing discrete Fourier transform (DFT) on the original data. Vectors derived from images with similar characteristics are trained to gather in adjacent nodes. DFT coefficients are used as features because significant features are compressed in several coefficients. Moreover,

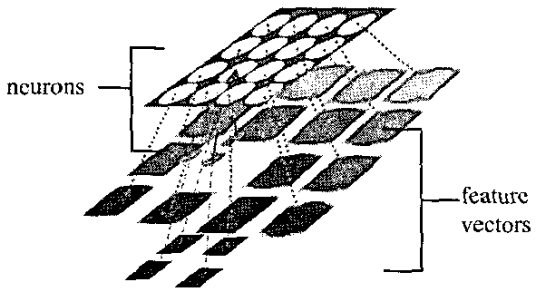

Fig. 2. The architecture of our SOM network. Solid lines indicate the links between parent-nodes and child-nodes while dotted lines map neurons to feature vectors expressed in images.

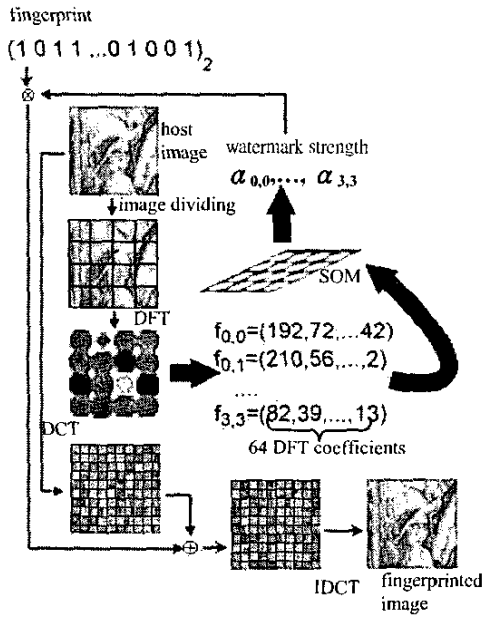

Fig. 3. The fingerprint embedding process.

the good properties of scaling-invariance and rotation-invariance in the DFT domain will make our features approximate the human's impression.

Training images of size 128×128 are categorized manually. For each training image, increase the watermark strength until the watermark is perceptible, then classify this image to that category in which all images can tolerate equal strength of the watermark. Use the first 64 DFT coefficients of these classified training images as input vectors when network learning, the SOM eventually memories the maximum watermark strength imperceptible to humans for every category of images after iterations of learning.

4.2. Adaptive watermarking

After fingerprint generation, the fingerprint codeword should be embedded in the host image. First, the host image is divided into several sub-images of size 128x128, DFT coefficients of these sub-images are then fed to the SOM individually to get their watermark strength. Finally, the fingerprint is added to DCT coefficients in 8x8 blocks of the host image according to the obtained strength in the last stage. The watermark embedding procedure is illustrated in Fig. 3.

Our watermarking method is oblivious, in other word, the watermark detection doesn't need the original image. Elements of the codeword vector (1) are in $GF(q)$, we first denote them in binary form $(v_0, v_1, ..., v_{q^k-1})$. Then, modify the DCT coefficients as the following.

$$\tilde{x}_k = x'_k + \alpha_b \cdot Q_k \cdot x_0 \quad \text{if } v_i=1$$
$$\tilde{x}_k = x'_k - \alpha_b \cdot Q_k \cdot x_0 \quad \text{if } v_i=0 \qquad (2)$$

where $k \in C, C$ is the set of index of DCT coefficients selected for watermark embedding. x_k is the kth coefficient in current DCT block and x'_k is the kth coefficient in previous block. α_b is the watermark strength of the bth sub-image. Q_k is the kth quantization value of the default JPEG quantization table.

The watermark can be extracted by simply comparing the values between x_k and x'_k. The watermark value is

$$v'_i = \begin{cases} 1 \text{ if } x_k > x'_k \\ 0 \text{ if } x_k < x'_k \\ ? \text{ otherwise} \end{cases} \qquad (3)$$

If the extracted value is "?", we randomly assign 0 or 1 to that bit.

5. EXPERIMENTAL RESULTS

256×256 and 512×512 images are used as the test data for our fingerprinting system. We construct the fingerprint codewords as in (1) from $GF(64^2)$ and let s=5, thus we can serve 262080 users and the length of the binary codeword is 24576. That is to say, 24576 coefficients in total 256×256 or 512×512 coefficients must be modified. This ratio is much larger than 1000/256×256 in [7] or several other literatures [4,9]. The maximum collusion size that our system can resist is 2, which is applicable in most applications.

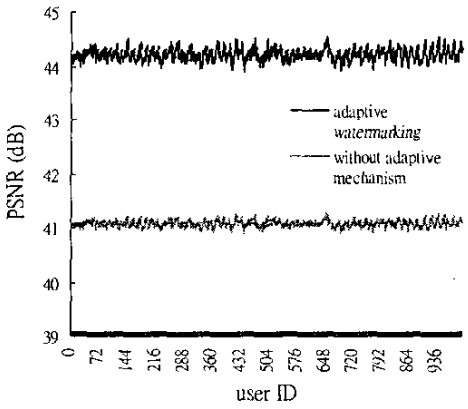

Fig. 4. Imperceptibility comparisons between the adaptive watermarking and the naïve watermarking schemes with fixed strength.

Table. 1. The robustness of our watermarking scheme.

Image Processing	NC	Image Processing	NC
50% JPEG	0.71	1/4 Cropping	0.55
Contrast enhancing	0.72	5% Noise adding	0.62

5.1. Imperceptibility

1000 codewords belonging to the first 1000 users are embedded into the host image to test the visibility of our system. If the image size is 512×512, the average PSNR of embedded images is 44.21dB. But if the image size is 256×256, the average PSNR reduces to 32.4937dB because we should embed much more bits in an 8×8 DCT block. Fig. 4 demonstrates the superiority of the content-adaptive watermarking in contrast to embedding with fixed watermark strength.

5.2. Robustness

We measure the robustness of watermarking by the normalized correlation $(NC) = \frac{\sum_i v_i v_i'}{\sum_i v_i^2}$. Table. 1 shows the NC values of extracted fingerprints after applying several types of image processing to fingerprinted images.

5.3. Correctness of traitor tracing

We randomly choose 50 pairs of users to be the 50 collusions, and then use average attacks to produce the pirate images. Only one false alarm occurs in all 50 pairs of colluders.

6. CONCLUSIONS

We have introduced a traceable content-adaptive fingerprinting system for multimedia. A c-TA code which can be constructed systematically and flexibly for multimedia is used. Combing this code with the content-adaptive watermarking can achieve high PSNR of fingerprinted images and the correctness of traitor tracing is close to 100%. This fingerprinting scheme with good performance is implemented under practical parameters, including applicable collusion size and number of consumers.

7.AKNOWLEDGEMENT

This work was partially supported by the National Science Council and the Ministry of Education of ROC under the contract No. NSC92-2622-E-002-002, NSC92-2213-E-002-023 and 89E-FA06-2-4-8. We would also like to acknowledge Chun-Hsiang Huang for helpful discussions concerning this research.

8. REFERENCES

[1] A. Silverberg, J. Staddon and J.L. Walker, "Applications of List Decoding to Tracing Traitors", IEEE Trans. I.T., Vol. 49, pp.1312-1318, 2003.

[2] B. Chor, A. Fiat, and M. Naor, "Tracing Traitors", CRYPTO'94, Vol. 839, pp.480-491, 1994.

[3] C. I. Podilchuk and W. Zeng, "Image-Adaptive Watermarking Using Visual Models", IEEE J. SAC, Vol. 16, pp.525-539, 1998.

[4] C.T. Hsu and J.L. Wu, "Hidden Digital Watermarks in Images", IEEE Trans. Image Processing, Vol. 8, pp. 58-68, 1999.

[5] D. Boneh and J. Shaw, "Collusion-Secure Fingerprinting for Digital Data", Proc. CRYPTO'95, pp.452-465, 1995.

[6] D.R. Stinson and R. Wei, "Combinatorial Properties and Constructions of Traceability Schemes and Frameproof Codes", SIAM J. Discr. Math., Vol. 11, pp.41-53, 1998.

[7] I.J. Cox, J. Kilian, F. T. Leighton and T. Shamoon, "Secure Spread Spectrum Watermarking for Multimedia", IEEE Trans. I.P., Vol. 6, pp.1673-1687, 1997.

[8] J. Dittmann, "Combining Digital Watermarks and Collusion Secure Fingerprints for Customer Copy Monitoring", IEE E. & C., London, pp. 6/1 - 6/6, 2000.

[9] Q. Cheng and T. S. Huang, "An Additive Approach to Transform-Domain Information Hiding and Optimum Detection Structure", IEEE Trans. Multimedia, Vol. 3, pp.273-284, 2001.

[10] R. Safavi-Naini and Y. Wang, "A Code for Sequential traitor tracing", ASPCS, pp. 211-224, 2002.

[11] R. Safavi-Naini and Y. Wang, "Sequential Traitor Tracing", IEEE Trans. I.T., Vol. 49, pp.1319-1326, 2003.

[12] W. Trappe, M. Wu, J. Wang and K.J.R. Liu, "Anti-Collusion Fingerprinting for Multimedia", IEEE Trans. S.P., Vol. 51, pp.1069-1087, 2003.