A Shift-Resisting Blind Watermark System for Panoramic Images

Ding-Yun Chen[1], Chun-Hsiang Huang[1], Ja-Ling Wu[1,2], Ming Ouhyoung[1]
[1]Communication and Multimedia Lab
Department of Computer Science and Information Engineering
National Taiwan University, Taipei, Taiwan
[2]CyberLink Inc., Taiwan
{dynamic, bh, wjl, ming}@cmlab.csie.ntu.edu.tw

Abstract

Digital watermark is a technique used to protect the ownership of images and their associated software. A useful watermark scheme should be visually imperceptible and must be robust to resist various attacks: such as JPEG compression, pixel-shifting, cropping, blurring, etc. In this paper, we propose a new watermarking technique that achieves the prescribed goals. We add synchronization bits to maintain the correct position information of the original 8x8 DCT block during extracting phase. To avoid cropping some part of image and losing watermark information, we add watermark bits according to edging features of each block. Therefore, the proposed approach needs neither the original image nor the original watermark pattern for watermark extraction, which is a useful feature in protecting real-time generated panorama images.

1. Introduction

Watermarks of semantic meaningful patterns (e.g. binary images) that can be directly recognized by naked eyes are important in Oriental culture environments. To resist lossy image processing such as JPEG compression, watermarks are embedded in the DCT domain of the original image. To obtain good quality and to embed more watermark bits, the higher the complexity in a block, the more watermark bits should be embedded. Our watermark system can protect an image processing software, such as VideoVR[1] --- a panorama construction tool. It is impractical, in this case, to use the original image for watermark extracting, because the panoramic images are generated from the user's side, and original image is not stored.

In most of existing watermark systems, if a user shifts an image even for one pixel, the watermark cannot be extracted correctly. Our approach can resist pixel shifting or image cropping attacks, because we use feature-based approach to synchronize the watermark embedding and extraction positions.

2. Embedding Phase

The embedding phase of proposed scheme is done by retrieving watermark bits and changing the A.C. values of the DCT of the original image. Fig.1 shows the block diagram of the embedding phase. Our approach combines the techniques presented in [2] and [3] to provide an invisible blind watermark scheme, which do not need the original image for retrieving the embedded watermark. The idea is: if we want to embed message '1', change AC value to satisfy the condition AC > DC, for example, and if we want to embed message '0', change AC value to satisfy the condition AC < DC. The distance of AC and DC must be larger than a given threshold.

To avoid watermark being destroyed by shifting and cropping the test image, we predetermine watermark positions by analyzing the characteristics of each 8x8 DCT block. As shown in Fig.5 (a), we get the 1/64 (resolution-reduction) image of the original one and calculate its gradient. The main idea is that the 1/64 image of the original one after JPEG compression or other image processing has not much difference from the 1/64 image of original image. The little difference come in both intensity and gradient. Therefore, we use intensity and gradient to aid the searching of the place for watermark embedding, as shown in Fig.5 (b).

In order to find the original 8x8 DCT block and get information for embedding during the extracting phase, synchronization bits are embedded in each DCT block. Therefore, our approach can resist shifting and cropping attacks and does not need the original watermark image for watermark extraction. For example, we choose two bits of constant position for middle-frequency coefficients as the embedded synchronization bits.

There may still have some watermark bits not yet been included. Therefore, the higher the complexity in a block, the more watermark bits should be embedded. Fig.4 shows how to determine the number of embedded watermark. In essence, the intensity feature is used for Y-axis and gradient feature is used for the X-axis index for watermark embedding.

3. Extraction Phase

The extraction phase is done by retrieving bits from DCT domain and putting them in correct watermark positions. Fig.2 shows the block diagram of the extraction phase. Besides the "Original 8x8 DCT block extraction" step, the other steps are similar to the embedding counterpart. Fig.3 shows how to find the original 8x8 DCT block. There is one correct choice out of 64 shifting combinations. Therefore, we select some complex places across an image in which more watermark bits are embedded, and then try to extract watermark from one out of 64 kinds of shift, each shift has a correlation ratio as compared with the watermark. Of course, the highest correlation ratio is the most possible place to extract the correct watermark.

0-7803-6301-9/00 $10.00 © 2000 IEEE

4. Experiment Result

Our approach has already combined with our own VideoVR system --- a panorama construction tool from live video of surrounding environment. Other commercial products can also use the proposed system to protect their copyright. Fig.7 shows images with embedded watermarks. Fig.8 show the extracted watermarks under various attacks, we can easily recognize that these are distorted versions of the embedded watermark pattern.

5. Conclusion

In this paper, a shift-resistant blind watermark system is proposed, in which the embedded/extracted watermark is a semantic meaningful pattern. Besides, we can analyze the correlation ratio, in the extraction phase, to know whether an image has been watermarked or not rather than by visually inspecting the extracted watermark pattern. This project is partially funded by the National Science Council (NSC, Taiwan) under the grant number NSC88-2622-E-002-002.

Reference

[1] D.Y. Chen, M. Ouhyoung, "VideoVR: A real-time system for automatically constructing panoramic images from video clips", Proc. of CAPTECH'98, Geneva, Switzerland, Nov. 1998, pp.140-143. (Lecture Notes in AI 1537, Springer)

[2] C.T. Hsu and J. L. Wu, "Hidden digital watermarks in images," IEEE Trans. On Image Processing, vol 8., No. 1, January 1999, pp.58-68.

[3] G.C. Langelaar, R.L.Lagendijk, and J. Biemond, "Real-time labeling of MPEG-2 compressed video"

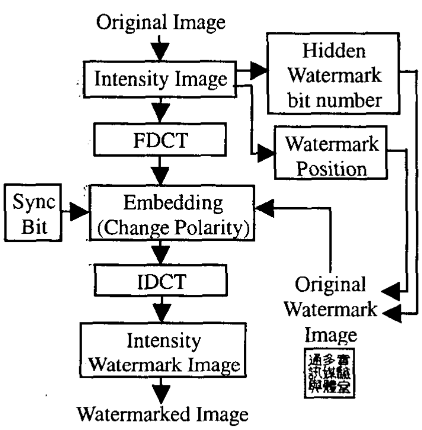

Fig1. Watermark embedding phase

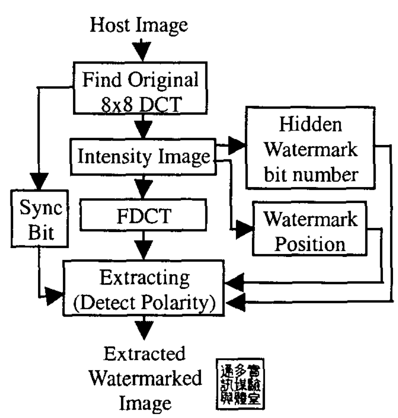

Fig2. Watermark extraction phase

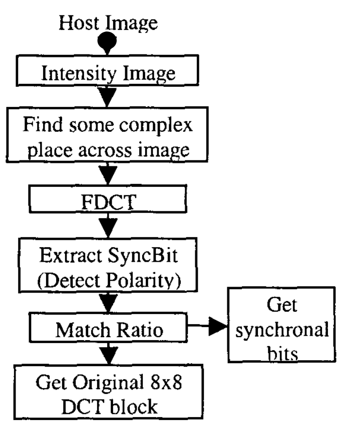

Fig3. Original 8x8 DCT block extraction

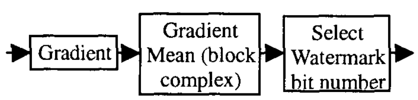

Fig4. Hidden watermark bit number assignment

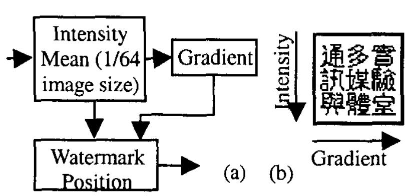

Fig5. Determination of watermark positions

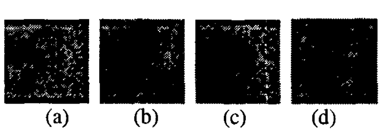

Fig.6 the black points are watermark bits which are not embedded, where (a)(b) is for the image Lena, (c)(d) is for a panoramic image, (b)(d) are the cases where more bits are embedded, and so less bits are block

Fig.7 (a)Panoramic image after watermark embedding (PSNR=33.1) (b)Cropped image from (a), size is 500x200 (c)Lena after watermark embedding (PSNR=39.9)

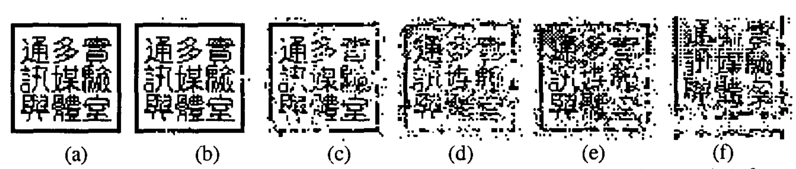

Fig.8 Original and extracted watermarks after different operations: JPEG, blur, pixel-shift, etc. (a)Original watermark, (b)Fig.7a→JPEG, (c)Fig.7a→crop to Fig.7b→JPEG, (d) Fig.7c→shift 2 pixel in vertical and horizontal→JPEG, (e) Fig.7c→blur→JPEG (f) Fig.7c→brightness value increased by 30→JPEG,. Note that '→' indicates various image processing or attacks.