

A RST RESILIENT OBJECT-BASED VIDEO WATERMARKING SCHEME

Dajun He and Qibin Sun

Institute for Infocomm Research (I2R)
21 Heng Mui Keng Terrace, Singapore 119613
Email: {djhe, qibin}@i2r.a-star.edu.sg

ABSTRACT

In this paper, a blind object-based video watermarking scheme, which is robust to MPEG4 compression and normal video editing such as Rotation, Scaling or Translation (RST), is proposed. The watermark is embedded into the Log-Polar Mapping (LPM) of DFT magnitude of the object; the synchronization problem introduced by RST is solved using PCA (Principle Component Analysis); by mapping the difference between watermarked video object and original video object from LPM domain to DFT domain, the watermarking can eventually be performed in the DFT domain so that video quality degradation caused by LPM and Inverse LPM (ILPM) is minimized. Experimental results verified the validity of the proposed robust watermarking scheme, which also has large watermark capacity.

1. INTRODUCTION

The object-based video applications are becoming growingly attractive in the areas such as video editing, wireless communication, etc because of its object-based nature. To protect the copyrights or the authenticity of the interested objects, an object-based video watermarking scheme is required. A good watermarking scheme should satisfy the following requirements: To be robust to RST because RST processing is a common processing in video editing; to be blind extraction (i.e., without original video) for video-related applications; and to have large capacity for authentication or fingerprinting purposes.

Previous object-based watermarking schemes, which reported to be robust to RST, have been proposed [1, 2, 3, 4]. In these schemes, PCA is utilized to obtain the information of rotation and scaling. For example, in [2, 3, 4], a pre-processing is employed to ensure that the same shape of video object can be acquired for watermark embedding and extraction: The main direction of the object, denoted as α_{CPA} , is computed first; then, either the object or the watermark logo is rotated by α_{CPA} ; finally, the object is scaled to a normalized object. Watermark embedding and extraction are performed on this normalized object. The practicability of these schemes mainly depends on whether the same shape of

object could be acquired during watermark embedding and extraction. However, in some cases such assumption is hard to be kept. For example, the scaling factor may be unknown; the shape error can be introduced by re-segmentation, or even the process of RST because during RST processing, shape of the object, as well as video content, may also be rotated or scaled. If the RST parameters are not small, some shape errors will be introduced by interpolation. [1] has to use the original object as the reference to extract watermark, so it is not a blind scheme; [2] assumed that the scaling factor is known for watermark extraction; [3, 4] even did not consider the issues of unknown scaling factor and the shape errors. Even though the same shape of object could be acquired for watermark extraction, the interpolation error caused by rotating or scaling the object in the pre-processing [2, 3, 4] will usually degrade the video quality around 6dB. Such serious quality degradation due to watermarking may not be acceptable in real applications.

Actually some image watermarking schemes, which reported to be resilient to RST even if the RST parameters are unknown, have been proposed [5, 6, 7]. All these schemes are developed from an FMT (Fourier Mellin Transform) based watermarking scheme [8], in which a strong RST-invariant representation is derived to embed watermark. Nevertheless, the watermarking scheme in [8] is difficult to implement and not robust to image compression which is a common processing for media storage. Lin [5] proposed to perform watermarking in LPM domain. Although it is blind and easy to be implemented, it is a one-bit scheme; and the robustness to compression is not considered either. To overcome the shortcomings in Lin's scheme, Zheng [6] proposed to embed watermark in DFT domain directly through a mapping between LPM domain and DFT domain. In [6], the synchronization problem is solved using the phase correlation between the original image and the watermarked image or 2D exhaustive search if the original image is not available. Obviously, the computation induced by 2D exhaustive search is very large and infeasible for video applications. In Kim's scheme [7], an invariant centroid of image is located to

offset the translation processing first; and then, the image is FMT transformed to embed the watermark. However, the method for locating centroid of image in their scheme cannot be used for locating centroid of video object because the size of object varies in video sequence.

In this paper, we propose a blind, robust and object-based video watermarking scheme with large capacity in LPM domain. Instead of 2D exhaustive search introduced in [6], PCA on the object is used to synchronize between watermark embedding and extraction, in the LPM domain, to reduce system computation; and a new mapping algorithm between the LPM domain and the DFT domain is also proposed to minimize the quality degradation of object/video. The paper is organized as follows. Section 2 presents an analytical description of the proposed scheme. Experimental results and conclusions are given in Section 3 and Section 4 respectively.

2. PROPOSED SCHEME

In the proposed scheme, watermarking is performed on the VOP (Video Object Plane) before MPEG4 coding. The block-diagrams of watermark embedding and extraction are shown in Fig.2 and Fig.3 respectively. As discussed in the previous section, synchronization (i.e., synchronizing the locations for watermark embedding and extraction) in LPM domain and new mapping algorithm between the LPM domain and the DFT domain are two main contributions of this paper. The details are given below.

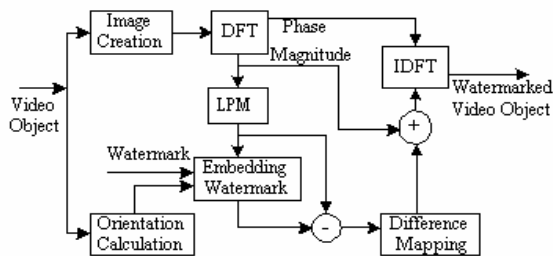


Fig.2 Block diagram of watermark embedding

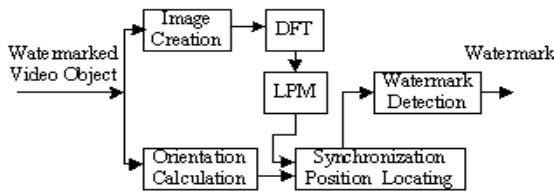


Fig.3 Block diagram of watermark extraction

2.1. Synchronization in LPM domain and difference mapping

2.1.1 Synchronization in LPM domain

Suppose $F(u, v)$ is the DFT of an image $i(x, y)$; and $F_1(u, v)$ is the DFT of image $i_1(x, y)$, obtained by rotating, scaling and translating the image $i(x, y)$, we have

$$i_1(x, y) = i(\sigma(x \cos \alpha + y \sin \alpha - x_0, \sigma(-x \sin \alpha + y \cos \alpha) - y_0)) \quad (1)$$

where $\alpha, \sigma, (x_0, y_0)$ are RST parameters. Therefore, the relation between $F(u, v)$ and $F_1(u, v)$ can be derived as

$$|F_1(u, v)| = \frac{\left| F\left(\frac{u \cos \alpha + v \sin \alpha}{\sigma}, \frac{-u \sin \alpha + v \cos \alpha}{\sigma}\right) \right|}{\sigma^2} \quad (2)$$

In log-polar coordinate, (2) can be rewritten as

$$|F_1(\rho, \theta)| = \frac{|F(\rho - \log \sigma, \theta - \alpha)|}{\sigma^2} \quad (3)$$

This equation indicates that scaled or rotated image in pixel domain corresponds to shifting the image along log-radius axis (ρ) or angle axis (θ) in LPM domain.

Instead of rotating the object twice for watermark embedding as done in [2, 3], we do it in LPM domain by shifting along the angle axis (θ). Such improvement avoids the interpolation errors introduced by rotation in [2, 3, 4]. Hence the problem of video quality degradation caused by interpolation of rotation in [2, 3, 4] is solved.

Let α_{CPA_0} and α_{CPA_1} represent the main direction of the original and watermarked objects respectively. The distances, S_0 and S_1 , by which the original and watermarked objects are required to be cyclically shifted, can be calculated as follows.

$$S_i = \text{round} \left(\frac{\alpha_{CPA_i} * R_g}{360} \right) \quad i = 0, 1 \quad (4)$$

where R_g is the resolution of image in angle axis.

After the synchronization position in the angle axis is found, the synchronization position in log-radius axis can be located by exhaustively searching a synchronization pattern along the log-radius axis. It means the search space has been reduced from 2D to 1D. Therefore, the computation for watermark extraction is significantly reduced. Let R_ρ represent the resolution of image in log-radius axis. The number of times to locate synchronization position in LPM domain is no longer $R_\rho \times R_g$ as described in [6], but R_ρ . In practical applications, the maximum and minimum scaling factors are usually known. So a search range can be limited to further reduce the computation. For example, if the maximum and minimum scaling factors are denoted as σ_{\max} and σ_{\min} respectively, the search range is $\left[\frac{\log \sigma_{\min} * R_\rho}{\log r_{\max}}, \frac{\log \sigma_{\max} * R_\rho}{\log r_{\max}} \right]$. Here, r_{\max} is the maximum radius.

2.1.2 Difference mapping

Instead of directly doing ILPM, which actually involves computational interpolation (converting the watermarked

image back from LPM domain), the difference between the watermarked and original images in LPM domain is calculated and mapped back from LPM domain to DFT domain to alleviate video quality degradation caused by LPM and ILPM. So the proposed watermarking is actually performed in the DFT domain though an effective LPM domain based watermarking is achieved, which is similar to [6]. However, our proposed algorithm for difference mapping is more accurate. For non-additive watermarking schemes, our scheme outperforms [6] in terms of the accuracy of watermark detection.

Suppose A is a sampling point in LPM domain; and A' is its corresponding point in the DFT domain. Usually, A' is not located at the sampling point, so we denote its four adjacent sampling points as A'_1, A'_2, A'_3 and A'_4 respectively (shown in Fig.4). Let Δ represents the difference of DFT magnitude in point A caused by watermark embedding. The new mapping algorithm is as follows:

- Given Δ , calculate the differences of points A'_1, A'_2, A'_3 and A'_4 (denoted as $\Delta_{A'_1}, \Delta_{A'_2}, \Delta_{A'_3}$ and $\Delta_{A'_4}$) based on their original magnitudes and their distances to point A' . For simplicity, $\Delta_{A'_1}, \Delta_{A'_2}, \Delta_{A'_3}$ and $\Delta_{A'_4}$ are all set as Δ in this paper.
- Modify $\Delta_{A'_1}, \Delta_{A'_2}, \Delta_{A'_3}$ and $\Delta_{A'_4}$ to satisfy equation (5).

This is to ensure that the magnitudes of the DFT coefficients modified by watermark embedding (Step g) are not negative. That is,

$$|F_{A'_i}| + \Delta_{A'_i} \geq 0 \quad i = 1, 2, 3, 4 \quad (5)$$

If $|F_{A'_i}| + \Delta < 0$, we modify them based on equation (6).

$$\begin{cases} \Delta_{A'_1} = |F_{A'_1}| \\ \Delta_{A'_i} = \Delta + (|\Delta| + |F_{A'_i}|) \quad i = 2, 3, 4 \end{cases} \quad (6)$$

To make sure that one sampling point in the DFT domain is mapped onto by only one point in the LPM domain to preserve the embedded watermark (otherwise the watermarks could be counteract by the mapping). We have to carefully select the points for watermark embedding in the LPM domain.

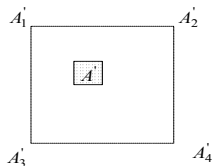


Fig.4 Four adjacent sampling points around A'

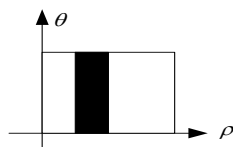


Fig.5 Watermarking area in the LPM domain

2.2. Watermark embedding

The watermark is embedded based on the following procedure:

- Calculate the orientation (α_{CPA_0}) of the object.
- Zero-pad the object to form a rectangular image. Please refer to our previous work [2] for details.
- For the padded image, compute the DFT coefficients and shift the DC coefficient to the center of the DFT domain.
- Log-polar map the DFT coefficients.
- Embed watermark: The watermark is embedded in the middle frequencies of the DFT domain; the corresponding area in the LPM domain is shown as the black part in Fig.5. Before embedding, the image is shifted by S_0 (calculated according to Equation (4)) along the angle axis; reverse shift is necessary after embedding. During embedding, to embed one bit watermark information, two points with identical radius are randomly selected as one group; and their magnitudes (F_A and F_B) are modified iteratively until equation (7) is satisfied

$$\begin{cases} |F_A| - |F_B| > TH & \text{watermark bit} = 1 \\ |F_B| - |F_A| > TH & \text{watermark bit} = 0 \end{cases} \quad (7)$$

where TH is a function of statistical distribution of video object and frequency at selected points [2].

- Calculate the difference between the watermarked and original images; and map this difference from LPM domain to DFT domain.
- Derive the watermarked DFT coefficients by adding the mapped differences to the magnitude of original DFT coefficients. Watermarked video object can then be acquired by Inverse DFT.

2.3. Watermark extraction

The procedure in watermark extraction is similar to that in watermark embedding: the received video object is zero-padded to form a rectangular image; and the DFT coefficients of this image are converted into LPM domain. Subsequently, watermark extraction is performed in LPM domain. Before detecting a watermark, the synchronization position should be located as described in Section 2.1. The watermark bit can be extracted by

$$\begin{cases} |F_A| \geq |F_B| & \text{watermark bit} = 1 \\ |F_B| < |F_A| & \text{watermark bit} = 0 \end{cases} \quad (8)$$

3. EXPERIMENTAL RESULTS

We have evaluated the proposed scheme on CIF testing videos: "Akiyo" and "Bream". The hidden information consists of 16 bits of synchronization pattern and 144 bits of watermark per frame. Fig.6 shows the results of

watermarked video quality in terms of PSNR. The PSNR of most frames exceeds 38dB.

The robustness evaluation is performed on different scales, rotations, MPEG4 compressions and their combinations. Fig.7 shows the robustness in terms of BER (Bit Error Rate) among a total of 144 watermark bits for video “Akiyo”. (We use different BER scales for better visual illustration). Similar results can be obtained for video “Bream”. From Fig.7, we can see that the BER does not exceed 0.2. Note that we did not employ any error correction coding scheme during testing. To further evaluate the robustness to compression, we compressed the watermarked video object under different JPEG Quality Factor after the object is zero-padded (Simulate the MPEG4 compression in I frame coding). The result is shown in Fig.8. It indicates that the watermarking scheme is remarkably robust to compression.

4. CONCLUSION

In this paper, we have proposed a new object-based video watermarking solution. The watermark is embedded in the Log-Polar Mapping (LPM) of DFT magnitude; and the PCA of the object is utilized to locate the synchronization position. Since the watermark is embedded in the middle frequencies of the DFT coefficients, video quality and robustness are well balanced. Experimental results demonstrated that the proposed solution is robust to lossy compression and normal object-based video editing such as RST, with a large capacity.

REFERENCES

[1] Guo Jie and Shi Peng-fei; “Object-based watermarking scheme robust to object manipulations”, *Electronics Letter*, Dec 2002, Vol. 38 (25), pp. 1656–1657.
 [2] Dajun He, Qibin Sun and Qi Tian; “An Object Based Watermarking Solution for MPEG4 Video authentication,” *Acoustics, Speech, and Signal Processing*, 2003. *Proceedings. (ICASSP '03)*. 2003 IEEE International Conference on, Vol.3, April 6-10, 2003 Page(s): III_537 -III_540
 [3] Mei-Yi Wu and Yu-Kuen Ho; “A Robust Object-Based Watermarking Scheme Based on Shape Self-Similarity Segmentation,” *Multimedia Software Engineering*, 2003. *Proceedings*. Fifth International Symposium on, 2003 Pages: 110 – 114
 [4] P. Bas, and B. Macq, “A New Video-Object Watermarking Scheme Robust to Object Manipulation”, *Proc. of ICIP01*, Tesseloniki, Greece, 2001
 [5] Ching-Yung Lin, Min Wu, Jeffrey A. Bloom, Matt L. Miller, Ingemar Cox, and Yui Man Lui, "Rotation, Scale, and Translation Resilient Public Watermarking for Images", *IEEE Trans. on Image Processing*, May 2001
 [6] Dong Zheng; Jiying Zhao; El Saddik, A; “RST-invariant digital image watermarking based on log-polar mapping and phase correlation,” *Circuits and Systems for Video Technology*,

IEEE Transactions on, Volume: 13, Issue: 8, Aug. 2003 Pages: 753 – 765

[7] Bum-Soo Kim etc. “Robust digital image watermarking method against geometrical attacks”, *Real Time Imaging*, vol.11, no.2, pp.139-149, 2003

[8] J. O’Ruanaidh and T. Pun, “Rotation, Scale, and Translation Invariant Digital Image Watermarking,” *Signal Processing*, Vol.66, No. 3, pp. 303-317, 1998.

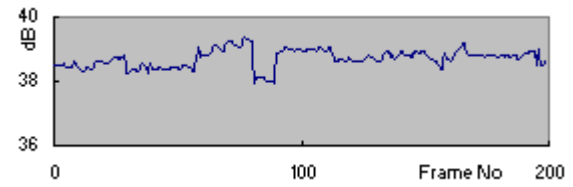
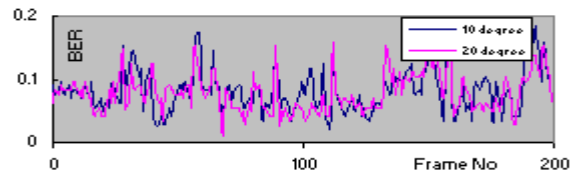
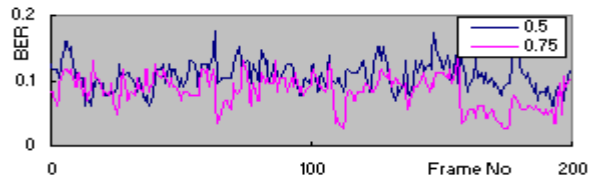


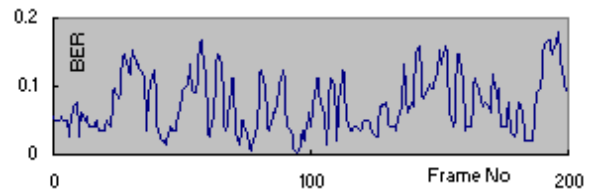
Fig.6 PSNR of Watermarking



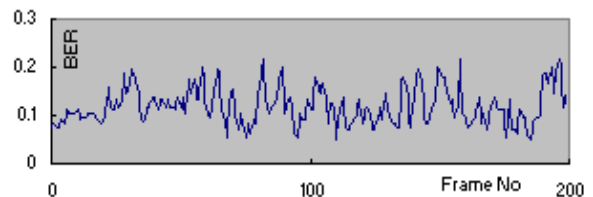
(a) Rotation (10° and 20°)



(b) Scaling (0.5 and 0.75)



(c) MPEG 4 Compression



(d) Rotation (10°), scaling (0.75) and compression
 Fig.7 The robustness to various video processing

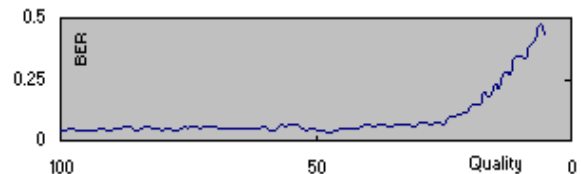


Fig.8 Robustness to compression