Forensic Tracking Watermarking against In-theater Piracy

Min-Jeong Lee, Kyung-Su Kim, and Heung-Kyu Lee

Department of Electrical Engineering and Computer Science,
Korea Advanced Institute of Science and Technology (KAIST)
Guseong-dong, Yuseong-gu, Daejeon, Republic of Korea
{mjlee,kskim,hklee}@mmc.kaist.ac.kr

Abstract. Many illegal copies of digital movies by camcorder capture are found on the Internet or on the black market before their official release. Due to the angle of the camcorder relative to the screen, the copied movies are captured with perspective distortion. In this paper, we present a watermarking scheme for tracking the pirate using local auto-correlation function (LACF) to estimate geometric distortion. The goals of watermarking are to find the suspected position of the camcorder in the theater and to extract the embedded forensic marking data which specifies theater information and time stamp. Therefore, our watermarking system provides conclusive evidence to take the pirate to the court. Experimental results demonstrate robustness of the LACF and accuracy of the proposed modeling.

Keywords: forensic tracking watermarking, in-theater piracy, local auto-correlation function (LACF).

1 Introduction

Many illegal copies of digital movies are found on the Internet or on the black market before their official release. These copies were made by recording the projected movie with a camcorder at various angles, according to the location of the pirate. Therefore, they are translated, rotated, scaled, and projected during camcording so that it is easy to visually detect such recordings but hard for watermark to survive. When an illegal copy is found on the Internet, it should be need to find the pirate. However, the identification of the captured movie is hard to configure by only comparing with the original movie. According to the requirements for protecting digital cinema [1], the information about the theater and the time is necessary and also the position of the pirate in the theater is needed.

The watermarking technique for digital cinema provides the way to identify when and where the movie was playing by embedding the related information into the movie in real-time. Several papers addressed watermarking for digital cinema. Leest *et al.* [2] proposed a video watermarking scheme which exploited the temporal axis to embed watermark by changing the luminance value of each

S. Katzenbeisser and A.-R. Sadeghi (Eds.): IH 2009, LNCS 5806, pp. 117–131, 2009.

frame and hence achieved robustness against geometrical distortions. Since the luminance change between frames may occur a flickering effect, the luminance modulation has to be performed slowly and smoothly. Delannay *et al.* [3] investigated the restoration of geometrically distorted images occurred by the camera acquisition angles. The compensation of the distortion required both unmodified content and modified content. Lubin *et al.* [4] embedded watermark into low spatial-temporal frequency domain for invisible, robust, and secure watermarking. To determine spatial-temporal regions of video sequences in the embedding procedure, a vision model-based masking computation was employed. These papers neither considered the projective distortions by camcorder capture nor told us the position of the pirate. Lee *et al.* [5] presented a blind watermarking scheme for digital cinema using local auto-correlation function (LACF) to resist to projective transform and embedded watermark in real-time. But they did not use the estimated geometric distortion to find out where the pirate was in the theater.

Chupeau *et al.* [6] proposed a forensic tracking system without embedding watermark. Their scheme determined the camcorder viewing angle to the screen and derived the approximate position of the pirate in the theater using feature points. The estimation of the eight-parameter homographic model required both temporally synchronized source videos and captured videos.

In this paper, we propose a watermarking scheme for tracking the pirate using local auto-correlation function to estimate geometric distortion. The estimation process is used for both recovering watermark to extract the embedded forensic marking data and finding the approximate position of the camcorder in the theater. the forensic marking data payload contains location (serial number of the theater) information and time stamp. Therefore, our watermarking system provides conclusive evidence to take the pirate to the court. The paper is organized as follows. Sec. 2 introduces our watermarking scheme including watermark embedding and detection. Forensic tracking using LACF is suggested in Sec. 3. Experimental results are presented in Sec. 4 and Sec. 5 concludes.

2 Watermarking Scheme

2.1 Watermark Embedding

This section describes how the watermark is embedded in the host video. Figure 1 shows the embedding procedure, which is designed to satisfy the requirements for digital cinema [1]. The watermark pattern is generated and then inserted into the video frames based on spread spectrum way with considering HVS.

In the presented scheme, the watermark pattern is used in two ways: one is to carry forensic marking data and extract it robustly, the other is to find illegally camcording position. In order to accomplish both roles, the watermark pattern should have periodicity for LACF to calculate geometric distortions. The periodicity is obtained by tiling the basic pattern [7]. First of all, the basic pattern, that follows a Gaussian distribution with zero mean and unit variance, is generated using a secret key and consists of 2-D random sequence of size $(M/m \times N/n)$. M and N denote the width and the height of the host video and m and n denote the

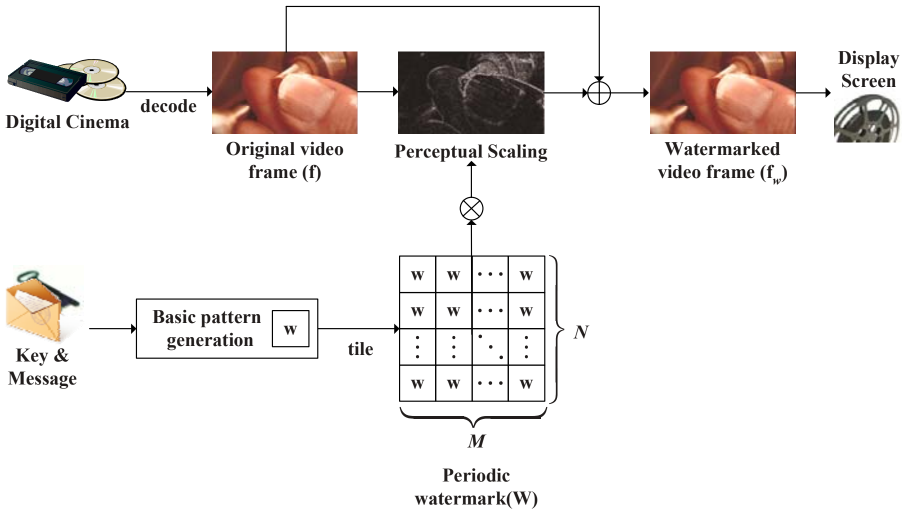


Fig. 1. Watermark embedding procedure

number of repetitions in horizontal and vertical direction, respectively. The 2-D basic pattern is then modulated to contain the bit payload (*e.g.*, serial number of the theater, time stamp, etc.). The modulated basic pattern w is repeated $m \times n$ times to get the periodicity. After a periodic watermark pattern of size $M \times N$ has been obtained, the pattern is embedded in an additive spread-spectrum way with perceptual scaling.

2.2 Watermark Detection

The entire detection process goes similar as [5]. It performs as follows: 1) find geometric distortions using the LACF on the estimated watermark pattern and 2) recover the watermark from the distortions and extract the embedded message. Figure 2 describes the process of watermark detection.

1) Estimating Geometric Distortion. Due to the fact that a blind detector is used, the embedded watermark is estimated by employing Wiener filtering as a denoising filter. Subtracting the denoised frame from the captured frame, we obtain an approximate version of the embedded watermark pattern. Both estimating geometric distortion and extracting watermark are proceeded using this extracted pattern.

As a result of camcorder capture, cinematic footage undergoes perspective-projective distortion when the position and/or viewing angle of the camcorder is changed. Let $\mathbf{x} = (x_1, x_2, x_3)^{\mathrm{T}}$ be the homogeneous vector that represents of a point in the original frame and $\mathbf{x}' = (x'_1, x'_2, x'_3)^{\mathrm{T}}$ be the homogeneous vector that represents a point in the geometrically distorted frame. The projective

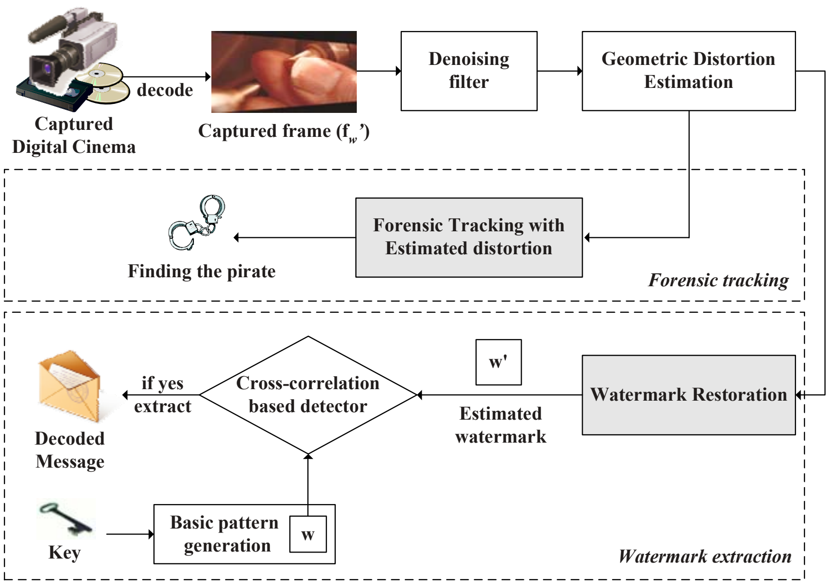


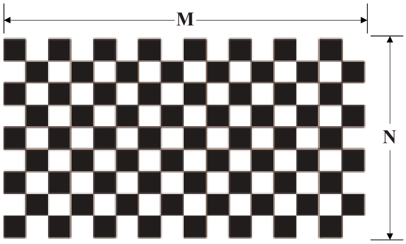
Fig. 2. Watermark detection procedure

transformation is a linear transformation on homogeneous 3-vectors represented by a non-singular 3×3 matrix [8]:

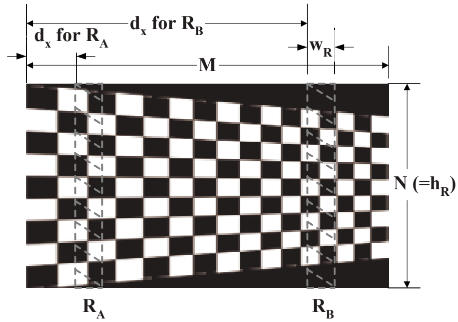
$$\mathbf{x}' = \mathrm{H}\mathbf{x}, \quad \text{where } \mathrm{H} = \begin{pmatrix} h_{00} & h_{01} & h_{02} \\ h_{10} & h_{11} & h_{12} \\ h_{20} & h_{21} & h_{22} \end{pmatrix} \tag{1}$$

Note that H is a homogeneous matrix, since as in the homogeneous representation of a point, only the ratio of the matrix elements is significant. There are eight independent ratios among the nine elements of H, and it follows that a projective transformation has eight degrees of freedom (DOF). Thus, four pairs of point-to-point correspondence in the original and distorted frames are required to determine eight DOF. In this paper, four corner points of the original video frame are selected as the original points and four corner points of the distorted video frame are chosen as the corresponding distorted points. Since both the embedder and the detector know the coordinates of four original points, it only needs to know those of four distorted points.

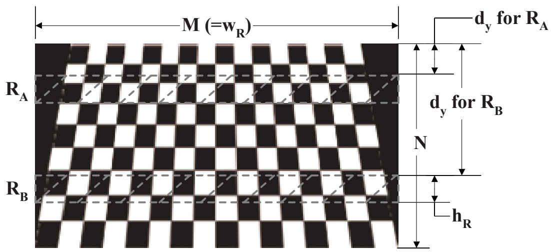
Local auto-correlation function (LACF) is employed for estimating projective transform. It computed the auto-correlation function of two local areas of the image that are parallel to each other instead of computing auto-correlation of the whole image. As shown in Fig. 3, two horizontally parallel local areas are needed for vertical projection, while two vertically parallel local areas are needed for horizontal projection. The two parallel local areas are denoted by R_A and R_B,



(a) Original watermark pattern

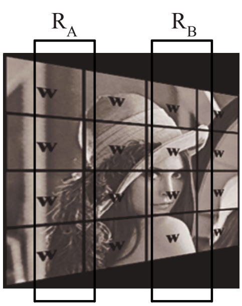


(b) Horizontal projection example

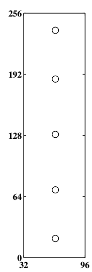


(c) Vertical projection example

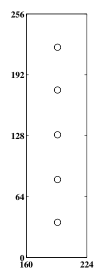
Fig. 3. Examples of projection attacks



(a) Watermarked image



(b) LACPs of $lacf_{R_A}$



(c) LACPs of $lacf_{R_B}$

Fig. 4. Watermarked image and LACPs: horizontal projection 20° with $fp = 10^{-6}$

respectively. The application of the LACF to (x, y) of region R on the estimated watermark pattern W' is modeled as

$$lacf_R(x,y) = \sum_{i=-w_R+1}^{w_R-1} \sum_{j=-h_R+1}^{h_R-1} W'(x+d_x(R)+\frac{i}{2}, y+d_y(R)+\frac{j}{2})^2 \tag{2}$$

where w_R and h_R are the width and the height of the region R, $d_x(R)$ and $d_y(R)$ are the distance of x-axis and y-axis from the upper-left corner point for selecting region R for LACF. The distance and the size of regions are adaptively selected by the size of the used basic pattern and lower bounds of projective distortion in a practical point of view. The LACF is calculated by FFT-based fast equation as follows:

$$lacf_R = \frac{\mathrm{IFFT}(\mathrm{FFT}(R) \cdot \mathrm{FFT}(R)^*)}{|R|^2}, \tag{3}$$

where "$*$"-*operator* denotes complex conjugation and R is the selecting local parallel region R_A or R_B. The result yielded by the LACF shows a periodic peak pattern. Then the geometric distortions are estimated and reversed by using a local auto-correlation peak (LACP). The LACP is detected from the results of the LACF by applying an adaptive threshold as follows:

$$lacp > \mu_{lacf} + \alpha_{lacf}\sigma_{lacf}, \tag{4}$$

where μ_{lacf} and σ_{lacf} denote the average and standard deviation of the LACF respectively. α_{lacf} is a value that is related to the false positive error rate. Presetting the maximum false positive error rate, we calculate α_{lacf} and obtain the threshold. Figure 4 shows the results of applying the LACF results to a watermarked image that has been subjected to projective distortion. In Fig. 4(b) and Fig. 4(c), two LACF results show the different intervals between each auto-correlation peak. Using the LACF results as a basis, we can calculate the coordinates of the distorted frame.

Now, it needs to construct a mathematical model using the intervals between LACP as parameters. The application of the LACF yields LACPs of R_A on the line $x = d_x(R_A) + w_R/2$ with interval δ_A and LACPs of R_B on the line $x = d_x(R_B) + w_R/2$ with interval δ_B as shown in Fig. 5. Let four corner points of the original watermark pattern be P_1, P_2, P_3, and P_4, respectively. C_{14} is the center point of $\overline{P_1P_4}$, C_{23} is the center point of $\overline{P_2P_3}$, and P_v is a vanishing point which intersects the extensions of $\overline{P_1P_2'}$ and $\overline{P_4P_3'}$. We denote C_A as a center point of LACPs of R_A and C_B as a center point of LACPs of R_B. The length of $\overline{C_AC_B}$ is obtained by

$$\overline{C_AC_B} = d_x(R_B) + \frac{w_R}{2} - (d_x(R_A) + \frac{w_R}{2}) = d_x(R_B) - d_x(R_A) \tag{5}$$

Our goal is to obtain the coordinates of projective-distorted points P_1, P_2', P_3', P_4. In this geometry, only the coordinates of P_2' and P_3' need to be computed because the positions of P_1 and P_4 do not change. Therefore, it is necessary to know the length of $\overline{P_2'C_{23}}(= \overline{C_{23}P_3'})$. First, we define P_A which is located at intervals of $n\delta_A/2$ from C_A and P_B which is located at intervals of $n\delta_B/2$ from C_B, where n is the vertical repetition times of basic watermark pattern (n can

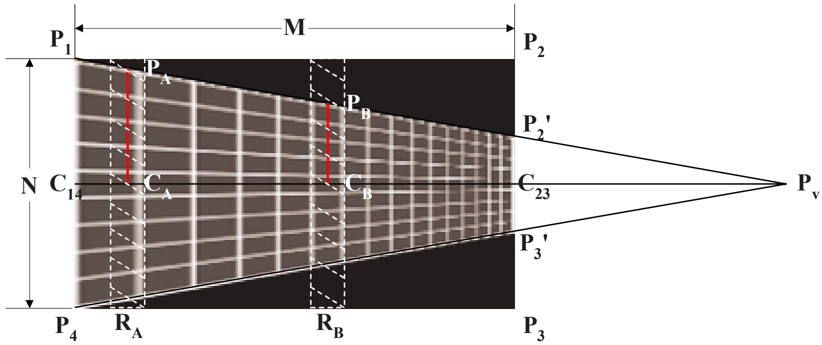


Fig. 5. A geometry of horizontally projected image

be replaced by m in case of vertical projection). Next, the similarity of triangle $\triangle P_A C_A P_v$ and $\triangle P_B C_B P_v$ is employed to obtain the length of $\overline{C_B P_v}$.

$$\overline{P_A C_A} : \overline{P_B C_B} = (\overline{C_A C_B} + \overline{C_B P_v}) : \overline{C_B P_v} \implies \overline{C_B P_v} = \frac{\overline{C_A C_B} \cdot \overline{P_B C_B}}{\overline{P_A C_A} - \overline{P_B C_B}} \tag{6}$$

The length of $\overline{C_{14} C_A}$ is obtained using $\overline{C_A P_v}$, $\overline{P_A C_A}$ and the similarity of triangle $\triangle P_A C_A P_v$ and $\triangle P_1 C_{14} P_v$.

$$\begin{aligned} \overline{P_1 C_{14}} : \overline{P_A C_A} = (\overline{C_{14} C_A} + \overline{C_A P_v}) : \overline{C_A P_v} \implies \\ \overline{C_{14} C_A} = \overline{C_A P_v} \left(\frac{\overline{P_1 C_{14}}}{\overline{P_A C_A}} - 1 \right) \end{aligned} \tag{7}$$

where $\overline{P_1 C_{14}} = N/2$. Using Eq. (5) - (7), $\overline{P_2' C_{23}} (= \overline{C_{23} P_3'})$ is defined as follows:

$$\overline{P_2' C_{23}} = \frac{N(\overline{P_{14} P_v} - M)}{2\overline{P_{14} P_v}} \tag{8}$$

Four pairs of points in Fig. 5 are transformed as follows:

$$\begin{aligned} A(-M/2+1, N/2) &\Rightarrow A(-M/2+1, N/2) \\ B(M/2, N/2) &\Rightarrow B'(M/2, \overline{P_2' C_{23}}) \\ C(M/2, -N/2+1) &\Rightarrow C'(M/2, -\overline{P_2' C_{23}}) \\ D(-M/2+1, -N/2+1) &\Rightarrow D(-M/2+1, -N/2+1) \end{aligned} \tag{9}$$

The center of the frame is designated as the origin $(0, 0)$ because it provides simplicity for camera modeling in Sec. 3. Finally, nine coefficients of projection matrix H in Eq. 1 are obtained by substituting above coordinates as follows:

$$\begin{aligned}
h_{00} &= \frac{MN - M + 2\overline{P_2'C_{23}}M - 4\overline{P_2'C_{23}}}{MN - M + 2\overline{P_2'C_{23}}M - 2N + 2}, \\
h_{01} &= h_{21} \;=\; 0, \\
h_{02} &= \frac{1}{2}\,M\frac{-M + MN - 2\overline{P_2'C_{23}}M - 2N + 2 + 4\overline{P_2'C_{23}}}{MN - M + 2\overline{P_2'C_{23}}M - 2N + 2}, \\
h_{10} &= \frac{-2\overline{P_2'C_{23}}}{MN - M + 2\overline{P_2'C_{23}}M - 2N + 2}, \\
h_{11} &= \frac{4\overline{P_2'C_{23}}(M-1)}{MN - M + 2\overline{P_2'C_{23}}M - 2N + 2}, \\
h_{12} &= \frac{-\overline{P_2'C_{23}}(M-2)}{MN - M + 2\overline{P_2'C_{23}}M - 2N + 2}, \\
h_{20} &= \frac{2(-1 + N - 2\overline{P_2'C_{23}})}{MN - M + 2\overline{P_2'C_{23}}M - 2N + 2}, \\
h_{22} &= 1
\end{aligned} \tag{10}$$

2) Watermark Extraction. The basic pattern of size $(M/m \times N/n)$ is generated using a secret key as a reference watermark. The watermark pattern is recovered from the geometric distortion using the inverse matrix H^{-1} with Eq. 10. Normalized cross-correlation between the estimated watermark w' and the reference watermark pattern w can be calculated so that it can be performed in less time with FFT by

$$C = \frac{\mathrm{IFFT}(\mathrm{FFT}(w') \cdot \mathrm{FFT}(w)^*)}{|w'| \cdot |w|} \tag{11}$$

If the normalized cross-correlation C exceeds a pre-defined threshold T, the hidden message is extracted successfully. The decision D to verity the existence of the watermark is made by

$$D = \max_{x,y}(C(x, y)) > T \tag{12}$$

where T is the detection threshold defined by

$$T = \mu_c + \alpha_c \sigma_c, \tag{13}$$

where μ_c is the average and σ_c is the standard deviation of $C(x, y)$. α_c is a pre-defined value that is related to false positive error rate.

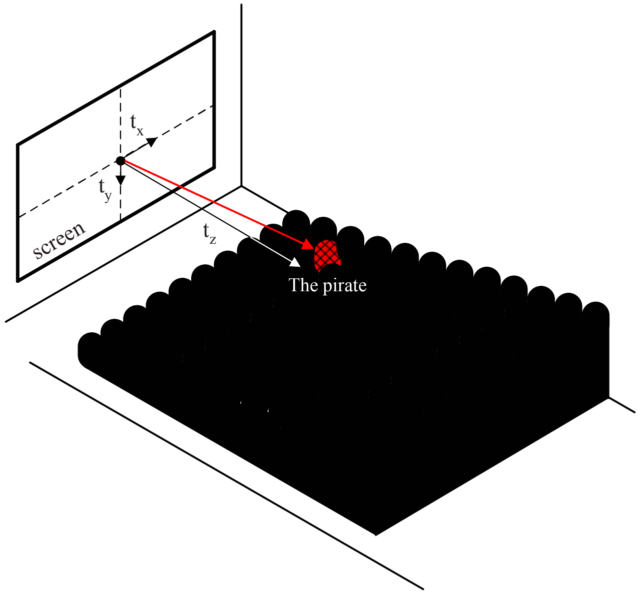


Fig. 6. Screen and camcorder geometry : 3D view

3 Forensic Tracking: Where a Camcorder Captures

3.1 Projective Geometry

For simplification purpose, we assume that the screen is planar and consider the camera models in [8]. The overall movie projection and camcorder capture consists of a screen and a camcorder. A geometrical representation is given in Fig. 6 and Fig. 7. Let now introduce the notation $\mathbf{X}$ for the world point of the screen represented by the homogeneous 4-vector $(X, Y, Z, 1)^{\mathrm{T}}$ and $\mathbf{x}$ for the captured image point by the camcorder represented by a homogenous 3-vector as defined in Eq. 1. Then the central projection mapping from world to image coordinates is given by:

$$\mathbf{x} = \mathrm{P}\mathbf{X}, \text{ where } \mathrm{P} = \lambda\mathrm{K}[\mathrm{R} \mid \mathbf{t}] \text{ with } \mathbf{t} = -\mathrm{R}[t_x, t_y, t_z]^{\mathrm{T}} \tag{14}$$

where P denotes a 3×4 homogeneous camera projection matrix, λ is a scale factor, and K is the camera calibration matrix. R is the rotation matrix representing the orientation of the camera coordinate frame and $[t_x, t_y, t_z]^{\mathrm{T}}$ is the coordinates of the camcorder center. The general form of the 3×3 camera calibration matrix K of a CCD camera is

$$\mathrm{K} = \begin{bmatrix} \alpha f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{15}$$

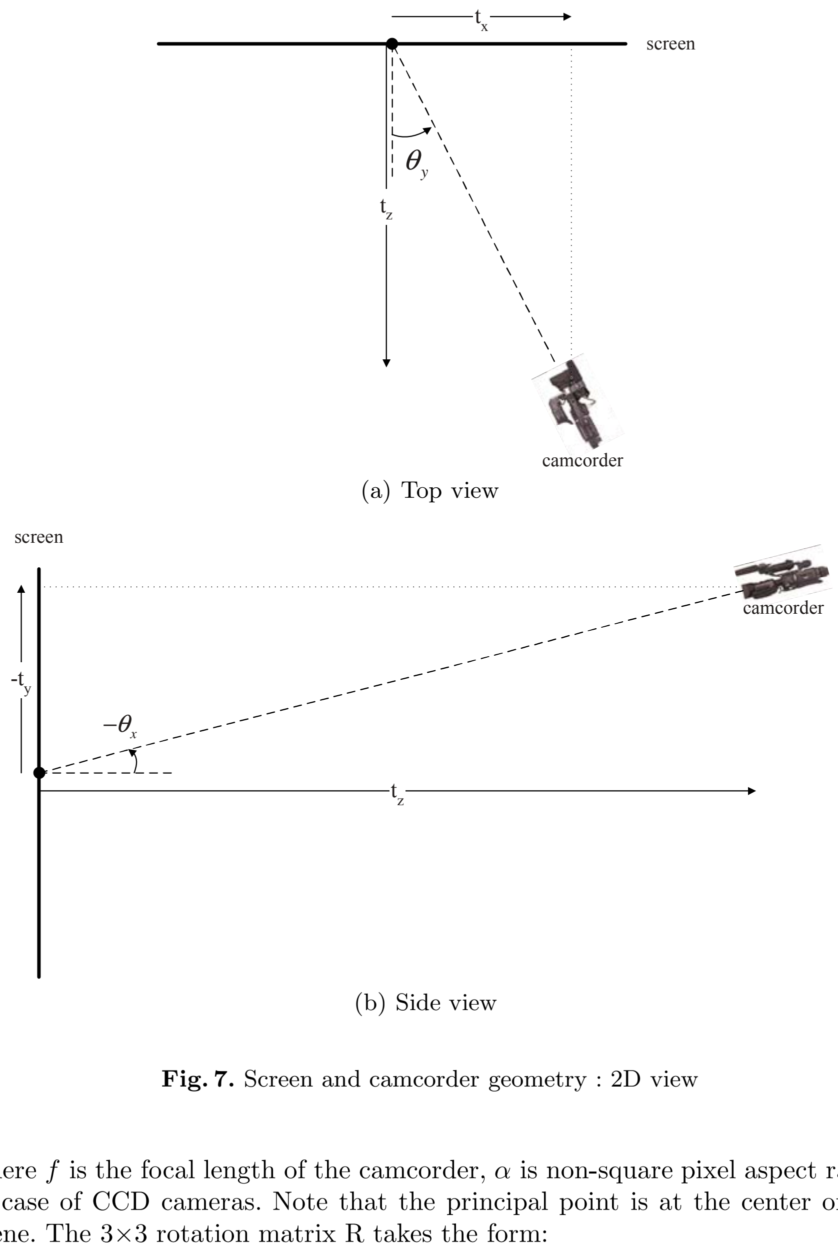


Fig. 7. Screen and camcorder geometry : 2D view

where f is the focal length of the camcorder, α is non-square pixel aspect ratios in case of CCD cameras. Note that the principal point is at the center of the scene. The 3×3 rotation matrix R takes the form:

$$\mathrm{R} = \mathrm{R}_z \mathrm{R}_y \mathrm{R}_x \tag{16}$$

In $\mathbb{R}^3$, coordinate system rotations of the x-, y-, and z-axis in a counter clockwise direction when looking towards the origin give the matrices

$$\mathrm{R}_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_x & \sin\theta_x \\ 0 & -\sin\theta_x & \cos\theta_x \end{bmatrix}, \mathrm{R}_y = \begin{bmatrix} \cos\theta_y & 0 & -\sin\theta_y \\ 0 & 1 & 0 \\ \sin\theta_y & 0 & \cos\theta_y \end{bmatrix}, \mathrm{R}_z = \begin{bmatrix} \cos\theta_z & \sin\theta_z & 0 \\ -\sin\theta_z & \cos\theta_z & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (17)$$

Then Eq. 16 is rewritten as

$$\mathrm{R} = \begin{bmatrix} \cos\theta_z\cos\theta_y & \sin\theta_z\cos\theta_x + \cos\theta_z\sin\theta_y\sin\theta_x & \sin\theta_z\sin\theta_x - \cos\theta_z\sin\theta_y\cos\theta_x \\ -\sin\theta_z\cos\theta_y & \cos\theta_z\cos\theta_x - \sin\theta_z\sin\theta_y\sin\theta_x & \cos\theta_z\sin\theta_x + \sin\theta_z\sin\theta_y\cos\theta_x \\ \sin\theta_y & -\cos\theta_y\sin\theta_x & \cos\theta_y\cos\theta_x \end{bmatrix} \quad (18)$$

Since it assumes that the screen is planar, we can set $Z = 0$ and redefine $\mathbf{X}$ as $(X, Y, 0, 1)^{\mathrm{T}}$. Then Eq. 14 is concisely expressed as

$$\mathbf{x} = \lambda\mathrm{K}[\mathbf{r_1}\ \mathbf{r_2}\ \mathbf{t}]\begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} \quad (19)$$

where $\mathbf{r_1}$ and $\mathbf{r_2}$ are the first and second column of the matrix R. Since Eq. 19 has the same form as Eq. 1, we decompose the matrix H in Eq. 1 as follows:

$$\mathrm{H} = \lambda\mathrm{K}[\mathbf{r_1}\ \mathbf{r_2}\ \mathbf{t}] \quad (20)$$

As shown in Fig. 6 and Fig. 7, $\mathbf{t}$ should be estimated to determine the position of the camcorder in the theater.

3.2 Modeling the Distortion

Let the inhomogeneous coordinates of $\mathbf{x}$ be (x, y). The projective transformation of Eq. 1 can be written in inhomogeneous form as

$$\begin{cases} x = \dfrac{h_{00}X + h_{01}Y + h_{02}}{h_{20}X + h_{21}Y + h_{22}} \\ y = \dfrac{h_{10}X + h_{11}Y + h_{12}}{h_{20}X + h_{21}Y + h_{22}} \end{cases} \quad (21)$$

with:

$$\begin{cases} h_{00} = \lambda\alpha f\cos\theta_z\cos\theta_y \\ h_{01} = \lambda\alpha f(\sin\theta_z\cos\theta_x + \cos\theta_z\sin\theta_y\sin\theta_x) \\ h_{02} = -\lambda\alpha f(\cos\theta_z\cos\theta_y t_x + (\sin\theta_z\cos\theta_x + \cos\theta_z\sin\theta_y\sin\theta_x)t_y + \\ \qquad + (\sin\theta_z\sin\theta_x - \cos\theta_z\sin\theta_y\cos\theta_x)t_z) \\ h_{10} = -\lambda f\sin\theta_z\cos\theta_y \\ h_{11} = \lambda f(\cos\theta_z\cos\theta_x - \sin\theta_z\sin\theta_y\sin\theta_x) \\ h_{12} = -\lambda f(-\sin\theta_z\cos\theta_y t_x + (\cos\theta_z\cos\theta_x - \sin\theta_z\sin\theta_y\sin\theta_x)t_y + \\ \qquad + (\cos\theta_z\sin\theta_x + \sin\theta_z\sin\theta_y\cos\theta_x)t_z) \\ h_{20} = \lambda\sin\theta_y \\ h_{21} = -\lambda\cos\theta_y\sin\theta_x \\ h_{22} = -\lambda(\sin\theta_y t_x - \cos\theta_y\sin\theta_x t_y + \cos\theta_y\cos\theta_x t_z) \end{cases} \quad (22)$$

Such a distortion model is actually a so-called homographic model, exactly describing the deformation of a flat scene photographed by a CCD camera, the optical axis of which is not perpendicular to the scene. Then solving the equations in Eq. 22 using the obtained projection matrix H in Eq. 10, we can identify the position of the camcorder $[t_x, t_y, t_z]^{\mathrm{T}}$.

4 Experimental Results

On HD-resolution clips of digital cinema, the fidelity, robustness, and accuracy of detecting the position are measured against camcorder capture attack. A 40-bit payload was embedded into each five-minutes clip to adhere to digital cinema initiatives [1]. A 2-D basic pattern whose size is 120×120 is generated and modulated to contain 2 bits of payload. Then the modulated pattern is tiled sixteen times ($m = 16$) to horizontal axis and nine times ($n = 9$) to vertical axis. Thus, the watermark pattern is formed 1920×1080 dimensions and embedded in the entire frame. To insert 40-bit payload into the five-minutes clip, a set of 20 differently modulated patterns is required. The watermark patterns in the set are repeated seven times during five minutes. For the LACF, the parameters in Eq. 2 are set for both horizontal and vertical projection. For horizontal projection, w_R is set to 120 ($= M/m$) and h_R is set to 1080 ($= N$). The $d_x(A)$ for region R_A is set to 180 and the $dx(B)$ for region R_B is set to 1500. Both $d_y(A)$ and $d_y(B)$ are set to zero. For vertical projection, w_R is set to 1920 ($= M$) and h_R is set to 120 ($= N/n$). The $d_y(A)$ is set to 180 and the $d_y(B)$ is set to 660. Both $d_x(A)$ and $d_x(B)$ are set to zero. After embedding, the average PSNR value was 46.0 dB for the test videos. Fidelity testing was performed as described in [4]. Clips were displayed using an EPSON EMP-TW1000 projector onto a wide screen. Projected clips were about 2.20m and 1.24m in horizontal and vertical directions, respectively.

Four expert observers participated in a two-alternative, forced-choice experiment in which each trial consisted of two presentations of the same clip, once with and once without the watermark present. Observers viewed the screen from two picture heights and were asked to indicate which clips contained the watermark. Each source clip was played in four times in each trial. Each trial lasted five minutes. No observer could determine the identity of the watermarked clip with certainty in any case.

4.1 Robustness Test

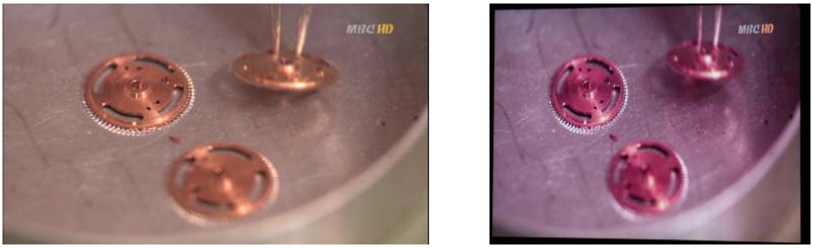
Against camcorder capture attack in practice, the robustness was tested, which included projective and affine transform as well as signal processing distortions at the same time. Watermarked clips were projected on a screen with the same environment for fidelity test and captured with a SONY HDR-FX1 camcorder tripod-mounted. In all cases, the 40 bits payload were extracted with the 100% reliability where average correlation values were larger than the threshold satisfying false positive probability 10^{-6}.

Table 1. Accuracy test result for synthetically projective distorted videos

		Horizontal projection	Vertical projection
Snapshot			
Synthesized	Coordinates [pixel]	A(-959,540) B'(960,360) C'(960,-359) D(-959,-539)	A'(-659,540) B'(660,540) C(960,-539) D(-959,-539)
	H matrix	$\begin{bmatrix} 1.0002 & 0 & 192.1335 \\ 0.0001 & 0.7999 & 0.1001 \\ 0.0002 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0.8148 & 0.0002 & 0.0926 \\ 0 & 1.0003 & 99.9862 \\ 0 & 0.0003 & 1 \end{bmatrix}$
Estimated	Coordinates [pixel]	A(-959,540) B'(960,357.6) C'(960,-356.6) D(-959,-539)	A'(-661.7,540) B'(662.7,540) C(960,-539) D(-959,-539)
	H matrix	$\begin{bmatrix} 1.0002 & 0 & 195.2167 \\ 0.0001 & 0.7966 & 0.1017 \\ 0.0002 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0.8168 & 0.0002 & 0.0916 \\ 0 & 1.0003 & 98.9250 \\ 0 & 0.0003 & 1 \end{bmatrix}$

4.2 Accuracy Test

For testing accuracy of the proposed modeling, we perform two kinds of experiments on videos. One is to measure the accuracy against synthetic distortions, the other is to demonstrate the accuracy against real camcorder capture attack. Since camcorder capture attack in practice occurs not only geometric distortions but also various signal processing attacks, we primarily prove the performance of our scheme on the situation of geometric distortions only. Table 1 shows the experimental results for synthetically distorted videos. In case of synthetic distortion, the geometry of each video frame is manipulated by given geometric distortion parameters without practical shooting so the real world coordinates cannot be determined. Instead, the estimated pixel coordinates and P matrices are compared with original ones when the videos are synthesized. Four original pixel coordinates are corner points of the video frame denoting A, B, C, and D depicted in Fig. 5. In case of horizontal projection, for example, the corresponding points B' and C' are calculated but A and D are fixed. As shown in Table 1, our scheme estimate the approximate P matrix using given geometric distorted videos.



(a) Snapshot of original video (1920×1080)

(b) Snapshot of camcorder captured video (1440×1080)

Fig. 8. Comparison of the original video and camcorder captured video

Table 2. Accuracy test result of position estimates on camcorder captured videos

[unit: meter]

	Original position			Estimated position			Error		
	t_x	t_y	t_z	t'_x	t'_y	t'_z	e_x	e_y	e_z
Case 1	-1.617	0	2.3	-1.7013	0.0003	2.4226	-0.0843	-0.0003	-0.1226
Case 2	0	0.108	2.5	0.0003	0.1417	3.2793	-0.0003	-0.0337	-0.7793

Figure 8 shows the comparison between the original video frame and the captured frames by the camcorder. The screen-displayed videos are recorded at a resolution of 1440×1080 pixels. By computing the distortion parameters of the captured videos, a numerical analysis is performed as described in Table 2. We compared the original position $[t_x, t_y, t_z]^{\mathrm{T}}$ of the camcorder and the estimated one $[t'_x, t'_y, t'_z]^{\mathrm{T}}$ and then measured the error $[e_x, e_y, e_z]^{\mathrm{T}}$. In addition to projective distortion, the captured videos suffered affine distortion and signal processing including aspect ratio change, contrast enhancement, and gamma correction. Although the error of the estimates is bigger than the cases of synthesized videos, its value does not exceed 1 meter in average. That is, when it assumes that a volume of a seat per one persion in the theater is bigger than 1.0 m^3, the errors in Table 2 are acceptable. The more the pirate wants to get visually qualified movies, the more the result of proposed modeling is exact. At worst case scenario, only one person next to the pirate would be a suspect. With an assumption that the pirate are picked out among more than one suspect, our scheme is still a strong evidence to prove his or her crime.

5 Conclusion

Many illegal copies of digital movies by camcorder capture are found on the Internet or in black market before their official release. It needs to provide conclusive evidence to take the pirate to the court for eradicating illegal camcorder capturers in the theater. To do so, we present a watermarking scheme for tracing the pirate using local auto-correlation function (LACF) to estimate geometric distortion. The message in the embedded watermark contains the information of time and location about showing a movie. The estimated geometric distortion tells about the approximate position of the pirate in the theater. Using LACF, our experiments proved the robustness against projective distortions and the accuracy of detecting the position of the pirate. Moreover, the proposed scheme was designed to embed the watermark in real-time to be applicable for digital cinema. In the future, our work can be extended to focus on the situation with combined projections.

Acknowledgments. This research is supported by Ministry of Culture, Sports and Tourism(MCST) and Korea Culture Content Agency(KOCCA) in the Culture Technology(CT) Research & Developement Program 2009, and by the IT R&D program of MKE/IITA. [2007-S017-01, Development of user-centric contents protection and distribution technology].

References

1. Digital Cinema Initiatives, LLC: Digital cinema system specification version 1.2 (2008), http://www.dcimovies.com/DCIDigitalCinemaSystemSpecv1_2.pdf
2. Leest, A., Haitsma, J., Kalker, T.: On digital cinema and watermarking. In: Proc. SPIE, vol. 5020, pp. 526–535 (2001)
3. Delannay, D., Delaigle, J., Macq, B., Barlaud, M.: Compensation of geometrical deformations for watermark extraction in the digital cinema application. In: Proc. SPIE, vol. 4314, pp. 149–157 (2003)
4. Lubin, J., Bloom, J., Cheng, H.: Robust, content-dependent, high-fidelity watermark for tracking in digital cinema. In: Proc. SPIE, vol. 5020, pp. 536–545 (2003)
5. Lee, M.J., Kim, K.S., Lee, H.Y., Oh, T.W., Suh, Y.H., Lee, H.K.: Robust watermark detection against D-A/A-D conversion for digital cinema using local auto-correlation function. In: Proc. ICIP, pp. 425–428 (2008)
6. Chupeau, B., Massoudi, A., Lefèbvre, F.: In-theater piracy: finding where the pirate was. In: Proc. SPIE, vol. 6819, pp. 68190T–1–68190T–10 (2008)
7. Maes, M., Kalker, T., Linnartz, J.P., Talstra, J., Depovere, G., Haitsma, J.: Digital watermarking for dvd video copy protection. IEEE Sig. Process. Magazine 17, 47–57 (2000)
8. Hartley, R.I., Zisserman, A.: Multiple view geometry in computer vision. Cambridge University Press, Cambridge (2004)