Robust Hash-based Image Watermarking with Resistance to Geometric Distortions and Watermark-Estimation Attack

Shih-Wei Sun
Dept. of Electrical Engineering
National Central University
Chung-Li, Taiwan 320, ROC
swsun@iis.sinica.edu.tw

Chun-Shien Lu
Institute of Information Science
Academia Sinica
Taipei, Taiwan 115, ROC
lcs@iis.sinica.edu.tw

Pao-Chi Chang
Dept. of Electrical Engineering
National Central University
Chung-Li, Taiwan 320, ROC
pcchang@ce.ncu.edu.tw

ABSTRACT

Digital watermarking provides a feasible way for copyright protection of multimedia. The major disadvantage of the existing methods is their poor resistance to both extensive geometric distortions and watermark-estimation attack (WEA). In view of this fact, our goal of this paper is to propose a robust image watermarking scheme that can withstand geometric distortions and WEA. Our scheme is mainly composed of three components: (i) robust mesh generation and embedding for resisting geometric distortions; (ii) improvement of fidelity using modified Noise Visibility Function (NVF); and (iii) construction of hash-based content-dependent watermark (CDW) for resisting WEA. Experimental results obtained from standard benchmark confirm the robustness of our method.

Keywords: Attack, Copyright protection, Embedding, Mesh, Hash, Robustness, Watermark

1. INTRODUCTION

Digital watermarking has been recognized as a helpful technology for applications of copyright protection, database retrieval, and authentication during the last decade. No matter what kinds of applications are considered, robustness is the critical issue affecting the practicability of a watermarking system. In data hiding, robustness refers to the capability of resistance to attacks that are used to destroy or remove hidden watermarks. In [19], attacks are classified into four categories: (1) removal attacks; (2) geometric attacks; (3) cryptographic attacks; and (4) protocol attacks. Up to now, resistance to extensive geometric attacks is still a challenging issue. Geometric attacks introduce synchronization errors to disable watermark detection without needing to remove the hidden information.

In the literature, the watermarking methods resistant to geometric attacks can be divided into three categories. The first category is to embed the watermark into the geometric invariant domain. In [7, 8], watermarking is conducted in the Fourier-Mellin domain and exploits its affine invariance. However, Fourier-Mellin domain is inherently vulnerable to cropping and other local geometric distortions.

The methods falling into the second category proposed to use template [9, 10] or insert periodic watermark pattern [11, 12] for the re-synchronization purpose. In [9, 10], templates were embedded in DFT domain to generate a shape of local peaks, which can be easily retrieved in the detection process for recovering geometric distortions. On the other hand, the local peaks are also easily extracted by the pirates in order to remove the templates [13]. In [11], the periodical structure of the watermark could be estimated from the autocorrelation function (ACF) to recover the imposed global transforms. However, the global watermark structure cannot deal with the local geometric distortions. In [12], the authors proposed to insert a periodic watermark pattern for the convenience of re-synchronization. The inserted periodic watermark was transformed as a lattice of peaks when ACF is applied in stego or geometrically attacked images. However, since the watermark is identical for every region, the collusion attack [3] can be used to efficiently estimate and remove the exacted watermarks. Although the synchronization problem is somewhat solved, the watermark information still cannot survive in collusion environments.3

The third category is called “feature-based watermarking scheme.” The feature points detected in the original image are used to form local regions for embedding. At the detection end, the feature points are expected to be robustly distributed at the corresponding positions. Among the ubiquitous feature point extraction methods, Harris detector has been popularly used in the fields of pattern recognition and computer vision. However, we found Harris detector [14] is still not robust enough to be used in digital watermarking. This is because Harris detector is rotation-and scaling-sensitive. In [15], Mexican-Hat wavelet filtering was used for feature point extraction. The Mexican-Hat wavelet filtering was implemented in frequency domain using FFT. Although 1-D FFT is widely used in implementing 2-D FFT to improve the computation efficiency, this implementation may lead another severe problem. That is, the input coefficient of 1-D FFT is quite different from the rotated version such that the different 1-D FFT filter will lead to different output. This is mainly due to that asynchronization effect is propagated to the final result of Mexican-Hat wavelet filtering. In [16], scale-space theory was applied for feature point extraction in that feature points were determined by automatic scale selection together with local extrema detection. Although the idea of scale-space feature point detection maybe used to solve scaling attacks, this approach is exactly a kind of exhaustive search. In addition, robust feature extraction plays a key role in this category.

In this paper, a novel robust mesh-based content-dependent image watermarking method is proposed. Our method belongs to the third category of geometric distortion resilient watermarking technologies. Because the first category is restricted to be affine invariant and the periodic patterns are easily removed in the

second category, the third category seems to be the best choice for watermarking applications. However the stability of feature points plays a key role in the third category. In view of this fact, we propose to use the Gaussian kernel as the pre-processing filter to stabilize the feature points. The Gaussian kernel is a circular and symmetric filter, so all the neighboring information of a pixel can be equally involved in filtering. A Gaussian kernel of large size, which is the marginal concept of scale-space theory, is used in our system. It is mainly adopted to generate an approximate version of an image from which second-moment matrix together with Harris detector is applied to extract feature points robustly. In order to resist watermark-estimation attacks, image hash [5] is further extracted and combined with the hidden watermarks to generate the Content-Dependent Watermark (CDW) [3]. CDW is able to resist watermark estimation attack in that even though the pirates can estimate the watermarks from meshes, they still cannot be successfully colluded to generate more correct watermark and remove it.

In addition to robustness, the transparency and false positive issues are also investigated. As to transparency, we improve original NVF [4] so that the embedded watermark energy is linearly proportional to image content's statistical variances. We also investigate the false positive issue in determining the proper threshold used to indicate the presence/ absence of a watermark. Experiment results obtained from standard benchmark verify that our scheme outperforms conventional feature-based watermarking methods [14,15,16].

The remainder of this paper is organized as follows. In section 2, we describe three important issues, including robust feature extraction, content-dependent watermark, and modified NVF, that are fundamental for embedding. In section 3, the proposed mesh-based content-dependent watermarking is described. Experimental results are demonstrated in section 4 to verify the performance of our scheme. Robustness comparisons with other methods are also conducted. Finally, conclusions are given in section 5.

2. ROBUST FEATURE EXTRACTION, CONTENT-DEPENDENT WATERMARK, and MODIFIED NVF

Several key issues of robust watermarking will be described in this section. They include robust feature extraction and content-dependent watermark for achieving robustness, and improved NVF for satisfying transparency.

2.1 Feature Extraction

A feasible feature point extraction technique should approximately tolerate common filtering, compression, and geometric attacks. In our method, Gaussian kernel filtering and Harris detector with second moment matrix are integrated for feature point extraction.

2.1.1 Gaussian Kernel Filtering

The Gaussian kernel filtering is a special case of scale-space filtering. In scale-space filtering, an image is filtered by more than one filter of different sizes to generate multiple frequency components. In some applications, filter size can be modified to adapt different affine transformation environments. But in digital watermarking, for the purpose of blind detection, we only select a specific filter size to generate one level of scale-space, which is convenient for watermark embedding and detection. In the following, Gaussian kernel filtering is described.

Let $I(x)$ be a cover image and let Gaussian kernel be defined as

$$g(\sigma) = \frac{1}{2\pi\sigma^2} \exp^{-\frac{x^2+y^2}{2\sigma^2}}.$$

The convolution of the Gaussian kernel and the cover image is defined as

$$L(x,\sigma) = g(\sigma) * I(x).$$

Because the Gaussian kernel is a circular shape, the resultant filtering response is rotation insensitive. This property inspires us to adopt it in our geometric-distortion resilient scheme. Here, the Gaussian kernel used here is the uniform scale-space kernel.

2.1.2 Harris Detector with Second Moment Matrix

Based on the filtering response obtained in 2.1.1, the local features invariant to affine transforms must be detected. Because linear derivatives are suitable for modeling the human visual front-end [1], the weighted difference computed by convolving the original signal with a derivative of the Gaussian difference operator are adopted in this paper. Based on the principle of Gaussian kernel, we have

$$L_x(x;\sigma) = \frac{\partial}{\partial x}\big(L(x,\sigma)\big) = \frac{\partial}{\partial x}\big(g(\sigma) * I(x)\big)$$
$$= \left(\frac{\partial}{\partial x} g(\sigma)\right) * I(x).$$

The Gaussian derivative is generally expressed as:

$$g_{x_1 \cdots y_m}(x,\sigma) = \frac{\partial}{\partial x_1 \cdots y_m} \frac{1}{2\pi\sigma^2} \exp^{-\frac{x^2+y^2}{2\sigma^2}},$$

where m is the derivative order, and x, y are the Cartesian coordinate in the image. Therefore, we can derive,

$$L_{x_1 \cdots y_m}(x,\sigma) = g_{x_1 \cdots y_m}(x,\sigma) * I(x). \quad (1)$$

This operation is efficient for implementing the convolution of Gaussian kernel with an image. Next the derivatives obtained from Eq. (1) form the so-called auto-correlation matrix which is defined as:

$$\mu(x,\sigma_I,\sigma_D) = \begin{bmatrix} \mu_{11} & \mu_{12} \\ \mu_{21} & \mu_{22} \end{bmatrix} = \sigma_D^2 g(\sigma_I) * \begin{bmatrix} L_x^2(x,\sigma_D) & L_x L_y(x,\sigma_D) \\ L_y L_x(x,\sigma_D) & L_y^2(x,\sigma_D) \end{bmatrix}. \quad (2)$$

The second moment matrix describes the gradient distribution of the local neighborhood of a point. The gradients are determined by σ_I (integration scale) and σ_D (derivation scale). In Eq. (2), $L_{xy}(x,\sigma_D)$ describes the second derivative along the y direction and the x direction sequentially. In addition, the derivatives are smoothed using a Gaussian window of size σ_I.

Basically, it is possible to compute the matrix for all possible combinations of kernel parameters. To making the system tractable, both derivation and integration are restricted to be $\sigma_I = s\sigma_D$. The parameter s can be experimentally determined.

Finally, Harris detector [2], widely used in salient point detection, is applied to detect the salient points. As to second moment matrix, $\mu(x,\sigma_I,\sigma_D)$ is closely related to the local auto-correlation function. Let α and β be the eigenvalues of $\mu(x,\sigma_I,\sigma_D)$. They will be proportional to the principal curvatures of the local auto-correlation function and form a rotationally invariant description of $\mu(x,\sigma_I,\sigma_D)$. In [2], if both curvatures are high, such that the local auto-correlation function is sharply peaked, then μ will be increased when shifts occur to indicate the existence of a salient point. In order to avoid calculating the explicit eigenvalues of μ, $Tr(\mu)$ and $\det(\mu)$ can be determined alternatively as:

$$Tr(\mu) = \alpha + \beta = \mu_{11} + \mu_{22}$$

$$\det(\mu) = \alpha\beta = \mu_{11} \cdot \mu_{22} - \mu_{12} \cdot \mu_{21},$$

$$H(x,\sigma_I,\sigma_D) = \det(\mu) - k \cdot Tr^2(\mu).$$

Feature point extraction is achieved by selecting the local maximum of $H(x,\sigma_I,\sigma_D)$, which is defined as

$$H(x,\sigma_I,\sigma_D) > H(x_w,\sigma_I,\sigma_D) \;\; \forall x_w \in NB(x),$$

where $NB(x)$ denotes the neighborhood of a pixel x.

2.2 Content-Dependent Watermark

Some researches [12, 14, 15, 16] proposed to insert multiple redundant watermarks into an image with the hope that it suffices to maintain robustness as long as at least one watermark exists. The common framework is that some kinds of image units such as blocks [12], meshes [14], or disks [15, 16] were extracted as carriers for embedding. With this unique characteristic, we propose to treat each image unit in an image like a frame in a video; in this way, collusion attacks can be equally applied to those image watermarking methods that employ a multiple redundant watermark embedding strategy. Therefore, once the hidden watermarks are successfully removed by means of a collusion attack, the function of robustness disappears so that the false negative problem occurs. Of particular interest is the possible quality improvement of attacked media data by means of collusion attack. In addition, copy attack is also efficient in defeating a watermarking system by creating ambiguity problem. Since the common operation of realizing both the collusion and copy attacks is watermark estimation, they are called watermark-estimation attack (WEA) [3].

In order to withstand watermark-estimation attack, we propose to embed content-dependent watermark (CDW) [3], which is composed of a watermark and a hash. Since this paper investigates a mesh-based watermarking scheme, the mesh-based hash [5] is considered here.For each mesh, its robust hash is extracted in the 8x8 block-DCT domain [5]. First, each normalized mesh is flipped and padded with its flipped version to form a 32×32 block. For a pair of 8x8 blocks, a hash bit, defined as the magnitude relationship between two AC coefficients, is represented as

$$MH_i(s) = \begin{cases} 1, & if \quad |f_k(p_1)| - |f_l(p_2)| \geq 0 \\ 0, & otherwise, \end{cases}$$

where $MH_i(\cdot)$ is a hash bit in a hash sequence MH_i, and $f_k(p_1)$ and $f_l(p_2)$ are two AC coefficients at positions p_1 and p_2 in 8×8 blocks k and l, respectively.

Given a pair of a hash MH_i and a watermark W, CDW_i can be generated as

$$CDW_i = S(W, MH_i),$$

where $S(\cdot)$ is a mixing function, which is basically application-dependent and will be used to control the combination of W and MH_i. The sequence CDW_i is the watermark that we want to embed in each mesh.

2.3 Modified NVF Embedding

In order to maintain transparency after watermarking, Noise Visibility Function (NVF) [4], which is an image-dependent visual model, is adopted in this paper. However, we find a defect in NVF that makes it not really transparent for smoothing regions of images. In this section, we provide a modification for NVF.

According to [4], NVF function was derived as

$$NVF(i,j) = \frac{1}{1+\theta\sigma_x^2(i,j)},$$

where θ is a tuning parameter that is calculated from every particular image and is defined as

$$\theta = \frac{D}{\sigma_{\max}^2},$$

where $\sigma_{\max}^2$ is the maximum local variance for a given image. In addition, $D \in [50,100]$ is experimentally determined. Based on NVF, the content adaptive watermark embedding in [4] was designed as

$$y = x + (1 - NVF) \cdot n \cdot S \tag{3}$$

and

$$y = x + (1 - NVF) \cdot n \cdot S + NVF \cdot n \cdot S_1, \tag{4}$$

respectively, where S and S_1 denote watermark strength. Eq. (14) is used to embed watermarks only in non-flat areas while Eq. (15) is used to embed watermarks both in the flat and non-flat areas.

However, we find that Eqs. (3) and (4) represent two extreme cases, as shown in Fig. 1. In order to satisfy transparency gracefully, we modify NVF and design as

$$y = x + (1 - NVF) \cdot n \cdot S + NVF \cdot n \cdot (1 - NVF) \cdot S_1. \tag{5}$$

The third term of Eq. (5) can be used to modify larger coefficients in highly textured areas and modify smaller coefficients in flat areas simultaneously so that the trade-off between transparency and robustness can be achieved gracefully. The comparison between the modified NVF and the conventional NVF is depicted in Fig. 1. It is observed that (i) for Eq. (3), no matter how complex or smooth the image content is, the third term is always zero such that watermark cannot be detected from flat areas; (ii) Eq. (4) will lead to severe quality degradation in smooth areas; and (iii) the modified NFV improves (i) and (ii) significantly.

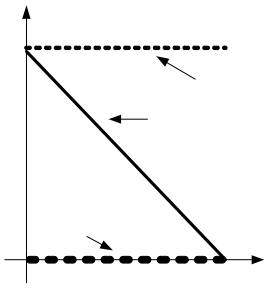

Fig. 1 Comparison between improved NVF and original NVF.

3. PROPOSED METHOD

Basically, the proposed method is similar to the mesh-based watermarking framework [14]. The major difference is that we have investigated some important issues (described in Section 2) to further improve the overall performance. In the main body of watermarking embedding and detection, our mesh warping is also different from [14] in that the false positive problem is taken into consideration. In the following, the watermark embedding and extraction processes will be described as follows.

3.1 Watermark Embedding

The watermark embedding process is outlined in Fig. 2. The content-dependent watermark [3] is embedded into each basic embedding unit, i.e., mesh, to combat watermark-estimation attack. Our embedding algorithm is described step by step in the following.

1) The cover image I is used to detect the feature points for decomposing into meshes. Let the set of feature points be $P = \{p_i \in R^2\}_{i=1\cdots N}$.

2) The Delaunay tessellation is performed using P to generate a set of meshes, $T = \{T_i\}_{i=1,2,\ldots,N}$.

3) The set of mesh-based robust media hash, $MH = \{MH_i\}_{i=1,2,\ldots,N}$ is extracted from T. In our proposed method, the size of hash bits is 64 [4].

4) Generate the image watermark W according a secrete key k.

5) Each mesh-based hash MH_i and the watermark W are combined to generate the content-dependent watermark, i.e.,

$$CDW = \{CDW_i\}, \quad 0 \le i \le N.$$
$$CDW_i = MH_i \cdot W$$

Although there is only one watermark W embedded for a cover image, the principle of CDW would lead to different embedded signals for different meshes. Therefore, the collusion attack will fail to estimate the watermarks from meshes and then collude them to obtain the exacted watermark W.

6) During embedding, the CDW_i should be repeatedly embedded into a mesh, in order to accommodate possible shifts of feature points caused by attacks. Here, each CDW_i is repeated kt times (in our test, $kt = 8$) and denotes as CDW_{R_i} before embedding. By considering the trade-off between robustness and transparency, we propose to shuffle the repeated watermark into a noisy form by multiplying the pseudo noise pn_tri. The resultant embedded signal is defined as

$$W_{T_i} = pn\_tri \cdot CDW_{R_i},$$

where W_{T_i} is a right triangle of size 32×32.

7) Affine transform is performed to transform W_{T_i} into the mesh shape of T_i to form W_{A_i}.

8) The modified NVF of T_i is calculated based on (5) as

$$MNVF_{w_i}(T_i, W_{A_i})$$
$$= (1 - NVF) \cdot n \cdot S + NVF \cdot n \cdot (1 - NVF) \cdot S_1.$$

9) W_{A_i} is embedded into the mesh T_i through the following embedding rule:

$$T_{w_i} = T_i + MNVF_{w_i}(T_i, W_{A_i}),$$

where T_{w_i} is the watermarked mesh. Finally, all the watermarked meshes T_{w_i} are generated and a stego image is produced.

3.2 Watermark Extraction

The watermark extraction process is depicted in Fig. 3. Basically, the watermark extraction process is the inverse process of watermark embedding. The watermark extraction process is described step by step in the following.

1) For a suspect image, the set of feature points, P, is generated and then the set of meshes, T, is generated for watermark extraction. In addition, the hash, MH_i, of each mesh is calculated. The original watermark W is generated based on a secret key k that is only known to owners. By integrating MH_i and W, the content-dependent watermark CDW_i can be produced. By repeating CDW_i kt times and shuffling the repeated result with the pseudo noise pn_tri, the right-triangle watermark W_{T_i} is made. An affine transformed watermark W_{A_i} is found by transferring W_{T_i} according to the shape of T_i. So far, the watermark W_{A_i} and the corresponding watermark positions in T_i are ready to extract the hidden watermark.

2) The popular MAP/ML estimator, Wiener filtering, is used to blindly extract the hidden signal. Wiener filtering is considered to be an efficient way [6] because watermark is usually a high-frequency signal.

3) The affine transformed watermark W_{A_i} is used for locating the position of watermark determined in T_i, . In addition, affine pseudo-noise $pn\_tri_{A_i}$ is used to separate the Wiener predicted signals $\hat{T}_i$ from the noisy signal $pn\_tri_{A_i}$.

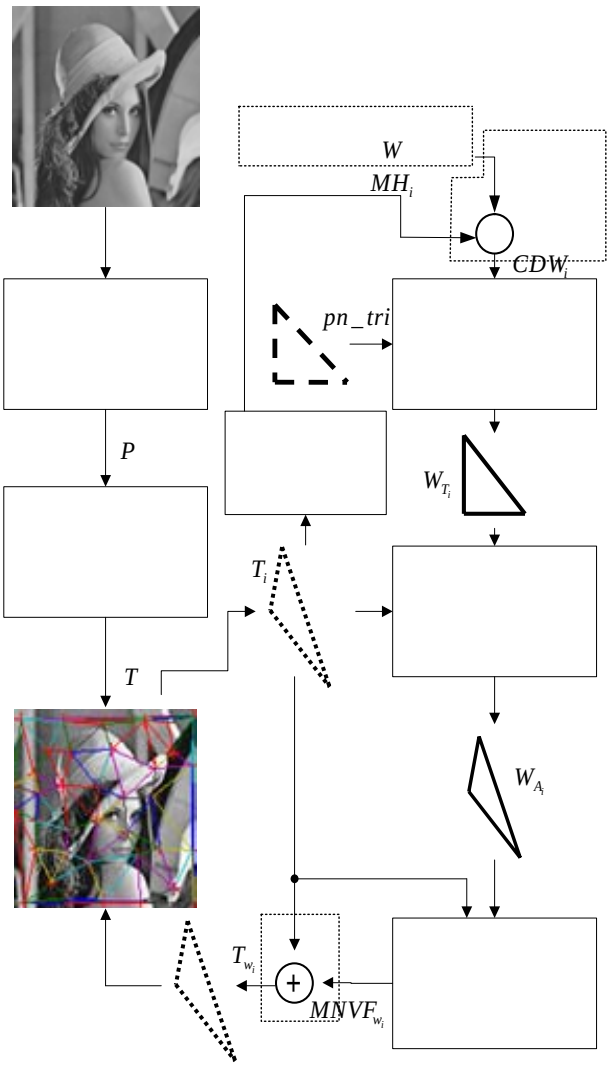

Fig. 2 The proposed watermark embedding process.

4) Each bit of the extracted watermark CDW_{D_i} is decided by a majority selection rule according to the repetition factor kt. If the number of ones is larger than $kt/2$, the watermark bit is determined as one. If the number of zeros is smaller than $kt/2$, the watermark bit is decided as zero. Otherwise, the watermark bit is given by means of random guess.

5) The extracted watermark W_{D_i} after eliminating the hash information is generated as

$$W_{D_i} = MH_i \cdot CDW_{D_i},$$

6) The Bit-Error Rate (BER_i) between W and W_{D_i} is calculated for each mesh. If BER_i is smaller than Th, it is said that a watermark exists in a mesh. In addition, if there are at least λ meshes detected to contain watermarks, the suspected image is determined to be a watermarked one.

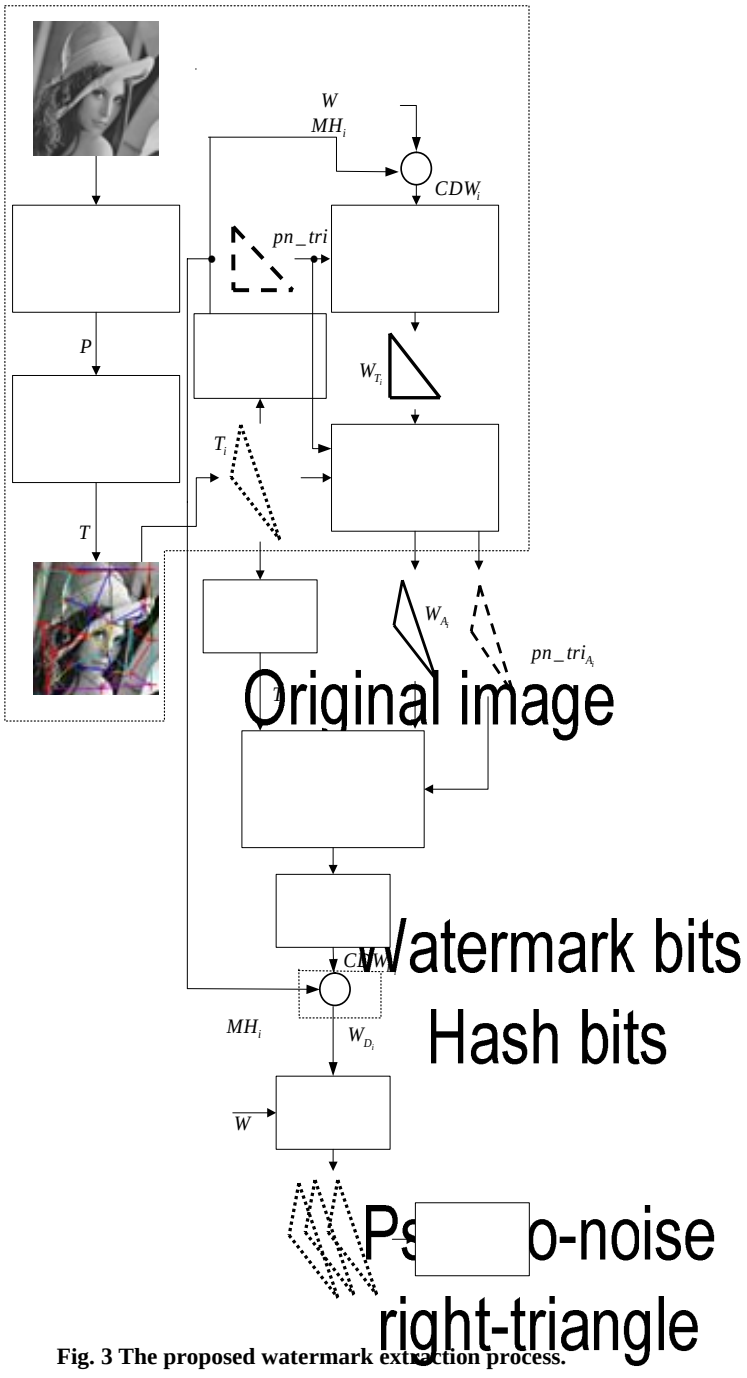

Fig. 3 The proposed watermark extraction process.

3.3 False Positive Analysis

It is meaningful to claim the robustness of watermarking system only when the false positive is taken into consideration in measuring robustness. Under a sufficiently small false positive

and with *Th=0.375* (note that *Th* can also be used as a variable for analyses), the number λ of meshes that are required to contain a watermark in order that a suspect can be determined to be a watermarked one can be derived as follows. Recall that the watermark size is *64* bits. It is said that two random signals (one from the original watermark and the other from the extracted signal) are similar if their bit error rate is smaller than or equal to *th*.

More specifically, the probability, p_m, of finding a pair of signals that satisfy a BER equal to *th* can be expressed as

$$p_m = \frac{(C_0^{32})^2 + (C_1^{32})^2 + \cdots + (C_{12}^{32})^2}{(C_0^{32})^2 + (C_1^{32})^2 + \cdots + (C_8^{32})^2 + \cdots + (C_{32}^{32})^2} \approx 3.97 \times 10^{-2}, \quad (6)$$

where C_b^{32} denotes the number of possible cases where ***2b*** bits are found to be different between two compared signal. Based on the above equation and a given value of λ, the false positive probability, p_{fp}, is defined as

$$p_{fp} = \sum_{n=\lambda}^{|T|} C_n^{|T|} (1-p_m)^{|T|-n} p_m^n \geq C_\lambda^{|T|} (1-p_m)^{|T|-\lambda} p_m^\lambda \approx C_\lambda^{|T|} p_m^\lambda, \quad (7)$$

where $C_n^{|T|}(1-p_m)^{|T|-n} p_m^n$ with $n > \lambda$ is sufficiently smaller than $C_\lambda^{|T|}(1-p_m)^{|T|-\lambda} p_m^\lambda$, and $(1-p_m)^{|T|-\lambda}$ is approximately 1 because $|T|$, denoting the number meshes in an image, is not large enough for $(1-p_m)^{T-\lambda}$ to be small. It is obvious from Eq. (7) that p_{fp} is lower bounded by $C_\lambda^{|T|} p_m^\lambda$. Let $\lambda = 6$, $p_{fp} \approx 4.0 \times 10^{-9}$, which is sufficiently small, could be obtained. In this paper, *Th=0.375* and $\lambda = 6$ are adopted for watermark detection.

4. EXPERIMENTAL RESULTS

The robustness of the proposed scheme is verified using standard benchmark, Stirmark 3.1 [17, 18]. Three standard images, Baboon, Lena, and Pepper, are used as cover images. After mesh-based watermark embedding, the PSNR values between the cover image and its stego image for Baboon, Lena, and Pepper are *35.31*dB, *38.61*dB, and *38.29*dB, respectively. No perceptual difference could be sensed. Although the PSNR of stego Baboon is smaller than *36*dB, it is still hard to find any quality degradation because the Baboon image is rather noisy.

The robustness test results are summarized in Table 1. In this table, each attack's name is followed by a digit, which indicates the number of times that the attack was performed with different parameters. In addition, each field shows the numbers of attacked images that are successfully identified as the watermarked ones. The detection thresholds were set as *Th=0.375* and $\lambda = 6$, as described in Sec. 3.3. We can observe that most of attacked images could be successfully detected except for few exceptions. These mostly include severe cropping attacks that break the connectivity of meshes and severe scaling attacks that make the feature points disappear.

Table 1

Robustness of our scheme vs. Stirmark 3.1: attacks are denoted as SPA: Signal Processing Attack including median filtering, Gaussian filtering, sharpening, and Frequency Mode Laplacian Removal (FMLR); JPEG: compression with quality factors, 90%~10%,; GLGT: General Linear Geometric Transform; CR: Color Reduce; CAR: Change of the Aspect Ratio: LR: Line Removal; RC: Rotation+Cropping; Scaling: with factors ranging from 0.5 to 2.0; RRS: Rotation+ReScaling; RB: Random Bending.

	Baboon	**Lena**	**Pepper**
SPA (6)	6	6	6
JPEG (12)	12	12	12
GLGT (3)	3	3	3
CR(1)	1	1	1
Flipping (1)	1	1	0
CAR (8)	6	8	8
LR (5)	5	5	5
Cropping (9)	7	8	8
RC (16)	16	15	14
Scaling (6)	4	5	4
RRS (16)	13	15	15
Shearing (6)	6	6	6
RB(1)	1	1	1

In order to demonstrate the superiority of our method, we made comparisons with other feature-based watermarking methods [14,15,16]. Robustness is meaningful only if false positive is taken into consideration. In [15], if the numerator value is detected to be larger than zero, then the suspect image is declared to be watermarked one. In [16], if at least one disk is detected to contain a watermark, the suspect image is declared to be a watermarked one. Although false positive analyses were conducted in [14,15,16], their results did not include this factor. In our method, a suspect image is detected to be truly watermarked based on the false positive analysis if at least six meshes are detected to contain a watermark with BER smaller than or equal to *th*.

Due to the limit of space, the comparisons are reported briefly as follows. Basically, our method can survive all non-geometric attacks of Stirmark 3.1, but the others [14,15,16] cannot. In particular, they cannot resist compression with higher ratios. For example, they can only tolerate JPEG compression with quality factor up to 30%. However, our method can resist JPEG with the lowest quality provided by Stirmark 3.1.

As to comparisons of resistance to geometric distortions, the results are shown in Table 2. In Table 2, the label of Mesh means "number of detected mesh/ number of total mesh." Yes/No means the presence/absence of a watermark. Besides, if the detection results obtained by our method in Table 2 are empty, this implies the parameters of attacks are not provided in Stirmark 3.1. It can be observed that all the line removal attacks are successfully

detected in our method and in [16]. Our method can detect the watermark from cropped Lena and cropped Pepper up to cropping factor 50%. Our method also survives general linear-geometric transform and change of aspect ratio very well. The reason we find is that our mesh detection is robust than disk detection [15,16]. The attack of rotation plus cropping was only tested up to 5° in [15]. When the attack was with large rotation angle (say up to 45°), the method [16] could survive. However, ours can only detect few. The main reason is that even there are mesh-watermarks detected in Lena and Pepper, robustness is satisfied by taking false positive into account. In Rotation+ReScaling attacks, our system can survive up to 45° except for the case of Baboon rotated with 45°. For scaling attacks, our method works well for scaling factors larger than *1*. When the scaling factor is significantly smaller than *1*, it is still a challenging problem for the feature-based watermarking methods. For shearing up to x-5%, y-5%, only our method can successfully extract the hidden watermarks.

Resistance of our method to watermark-estimation attacks is similar [3]. However, the content-independent watermarking methods [14,15,16] cannot survive WEA. In sum, extensive experiment results verify that our method outperforms all the other feature-based watermarking methods.

Table 2

Our scheme vs. [14,15,16] for robustness comparisons with Stirmark 3.1. The attacks are briefly described as follows. LR: Line Removal, column and row; Crop: Cropping with percentage; GLGT: General Linear Geometric Transform: parameter: (1.013, 0.008, 0.011, 1.008); CAR: Change of the Aspect Ratio: parameter (1.00, 1.20); RC: Rotation+Cropping with degree; Scaling: with factors ranging from 0.5 to 2.0; RRS: Rotation+ReScaling with degree; Shearing: not specific in Stirmark 3.1; Shearing 5: x-5% y-5%; RB: Random Bending.

Table 2.1 Geometric attacks for Baboon

Attacks	Proposed method		[16]	[15]	[14]
	Mesh	Yes/No			
LR: 5 ,1	50/213	Yes		6/11	
LR: 17, 5	28/205	Yes	1, 2, 2	3/11	
Crop 10%	27/172	Yes		2/11	
Crop 25%	14/114	Yes	1, 2, 2		
Crop 50%	4/36	No			
GLGT	30/226	Yes	0, 3, 3	5/11	
CAR	10/253	Yes			
RC 5.00	20/188	Yes		0/11	
RC 10.00	20/164	Yes			OK
RC 20.00			1, 3, 3		
RC 45.00	6/104	Yes	1, 1, 1		
RRS 1.00	24/218	Yes		4/11	
RRS 30.00	6/215	Yes			
RRS 45.00	4/234	No			
SC 80%					defeat
SC 90%	4/170	No	2, 3, 4		
SC 150%	19/532	Yes			
SC 200%	32/1109	Yes			
Shearing					OK
Shearing 5	12/207	Yes	0, 0, 0	0/11	
RB	23/203	Yes	0, 2, 3		

Table 2.2 Geometric attacks for Lena

Attacks	Proposed method		[16]	[15]	[14]
	Mesh	Yes/No			
LR: 5 ,1	69/208	Yes		3/8	
LR: 17, 5	35/199	Yes	5, 6, 6	0/8	
Crop 10%	33/166	Yes		2/8	
Crop 25%	22/118	Yes	4, 4, 4		
Crop 50%	8/54	Yes			
GLGT	47/211	Yes	7, 7, 7	4/8	
CAR	18/237	Yes			
RC 5.00	21/177	Yes		0/8	
RC 10.00	8/158	Yes			OK
RC 20.00			5, 5, 5		
RC 45.00	4/96	No	2, 2, 3		
RRS 1.00	24/205	Yes		0/8	
RRS 30.00	8/197	Yes			
RRS 45.00	9/201	Yes			
SC 80%					OK
SC 90%	6/170	Yes	4, 5, 5		
SC 150%	17/493	Yes			
SC 200%	27/860	Yes			
Shearing					OK
Shearing 5	15/182	Yes	1, 1, 1	1/8	
RB	26/212	Yes	4, 5, 5		

Table 2.3 Geometric attacks for Pepper

Attacks	Proposed method		[16]	[15]	[14]
	Mesh	Yes/No			
LR: 5 ,1	75/210	Yes		3/4	
LR: 17, 5	43/201	Yes	5, 5, 5	1/4	
Crop 10%	43/171	Yes		2/4	
Crop 25%	27/129	Yes	2, 2, 2		
Crop 50%	6/50	Yes			

GLGT	63/223	Yes	5, 5, 5	0/4	
CAR	8/244	Yes			
RC 5.00	28/177	Yes		0/4	
RC 10.00	22/157	Yes			OK
RC 20.00			3, 4, 4		
RC 45.00	5/112	No	1, 1, 1		
RRS 1.00	49/209	Yes		2/4	
RRS 30.00	6/194	Yes			
RRS 45.00	12/201	Yes			
SC 80%					OK
SC 90%	10/175	Yes	6, 6, 6		
SC 150%	12/455	Yes			
SC 200%	27/801	Yes			
Shearing					OK
Shearing 5	26/199	Yes	0, 1, 1	0/4	
RB	41/212	Yes	3, 3, 3		

5. CONCLUSIONS

A mesh-based content-dependent image watermarking method that can resist extensive geometric attacks and watermark estimation attacks is proposed. The major contribution of our method is threefold. First, traditional NVF function that is commonly adopted to satisfy transparency is modified to further improve transparency for various images. Second, a robust mesh extraction is proposed to enhance the feasibility of feature-based watermarking methods. Third, content-dependent watermark that is composed of a watermarking and a hash is proposed to resist watermarking-estimation attack. Standard benchmark has verified the robustness of the proposed scheme.

However, the major weakness of our scheme is its high complexity. Most of time is spent in the mesh warping operation. As a result, our system at its current status is not suitable for real-time applications. This problem can be properly dealt with, if our system is integrated with grid computing.

Acknowledgment: This paper was supported, in part, by the National Science Council under NSC grant 92-2422-H-001-004.

6. REFERENCES

[1] K. Mikolajczyk, "Detection of local features invariant to affine transformations", Ph.D. thesis, INPG Grenoble, July 2002.

[2] C. Harris and M. Stephen, "A combined corner and edge detector," in *Proc. 4th Alvey Vision Conf.*, pp.147-151, 1988.

[3] C. S. Lu and C.Y. Hsu, "Content-Dependent Anti-Disclosure Image Watermark," *Proc. 2nd Int. Workshop on Digital Watermarking*, LNCS 2939, pp. 61-76, Seoul, Korea, 2003.

[4] S.Voloshynovskiy, A.Herrigel, N.Baumgartner and T.Pun, "A stochastic approach to content adaptive digital image watermarking," *Proc. Int. Workshop on Information Hiding*, LNCS 1768, pp. 211-236, 1999.

[5] C.S Lu, C.Y. Hsu, S.W. Sun, and P.C. Chang, "Robust Mesh-based Hashing for Copy Detection and Tracing of Images," *Proc. IEEE Int. Conf. on Multimedia and Expo*: special session on Media Identification, Taipei, Taiwan, 2004.

[6] J. R. Hernandez and F. Perez-Gonzalez, "Statistical analysis of watermarking schemes for copyright protection of images," *Proc. IEEE*, Vol. 87, pp. 1142-1143, July 1999.

[7] J. O'Ruanaidh and T. Pun, "Rotation, scale and translation invariant spread spectrum digital image watermarking," *Signal Processing*, Vol.66, No. 3, pp. 303–317, May 1998.

[8] C. Y. Lin, M. Wu, J. A. Bloom, I. J. Cox, M. L. Miller, and Y. M. Lui, "Rotation, scale and translation resilient watermarking for images," *IEEE Trans. Image Processing*, Vol. 10, No. 5, pp. 767–782, May 2001.

[9] S. Pereira, T. Pun, "Robust template matching for affine resistant image watermarks," *IEEE Trans. Image Processing*, Vol. 9, No. 6, pp. 1123-1129, June 2000.

[10] S. Pereira, T. Pun, "An iterative template matching algorithm using the Chrip-Z transform for digital image watermarking," *Pattern Recognition* (33), pp. 173-175, 2000.

[11] M. Kutter, "Watermarking resisting to translation, rotation and scaling," *Proc. SPIE International Symposium on Voice, Video, and Data Communication*, Boston, November 1998.

[12] S. Voloshynovskiy, F. Deguillaume, and T. Pun, "Multibit digital watermarking robust against local nonlinear geometrical distortions," in *Proc. IEEE Int. Conf. Image Processing*, Thessaloniki, pp. 999–1002, Oct. 2001.

[13] A. Herrigel, S. Voloshynovskiy, Y. Rytsar, "The watermark template attack," *Proc. SPIE Security and Watermarking of Multimedia Contents III* (Vol. 4314), San Jose, January 2001.

[14] P. Bas, J. M. Chassery, and B. Macq, "Geometrically invariant watermarking using feature points," *IEEE Trans. Image Processing*, Vol. 11, No. 9, pp.1014-1028, September 2002.

[15] C. W. Tang and H. M. Hang, "A Feature-Based Robust Digital Image Watermarking Scheme," *IEEE Trans. Signal Processing*, Vol. 51, No. 4, pp.950-958, April 2003.

[16] J. S. Seo and C. D. Yoo, "Localized image watermarking based on feature points of scale-space representation," *Pattern Recognition* (37), pp. 1365-1375, 2004.

[17] F. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Attacks on Copyright Marking Systems", *Proc. Int. Workshop on Information Hiding*, LNCS 1575, pp. 219-239, 1998.

[18] F. Petitcolas, "Watermarking Schemes Evaluation," *IEEE Signal Processing Magazine*, Vol. 17, No. 5, pp. 58-64, 2000.

[19] S. Voloshynovskiy, S. Pereira, V. Iquise, and T. Pun, ``Attack Modelling: Towards a Second Generation Watermarking Benchmark," *Signal Processing*, Vol. 81, pp. 1177-1214, 2001.