Media Hash-Dependent Image Watermarking Resilient Against Both Geometric Attacks and Estimation Attacks Based on False Positive-Oriented Detection

Chun-Shien Lu, *Member, IEEE*, Shih-Wei Sun, Chao-Yong Hsu, and Pao-Chi Chang

***Abstract*—The major disadvantage of existing watermarking methods is their limited resistance to extensive geometric attacks. In addition, we have found that the weakness of multiple watermark embedding methods that were initially designed to resist geometric attacks is their inability to withstand the watermark-estimation attacks (WEAs), leading to reduce resistance to geometric attacks. In view of these facts, this paper proposes a robust image watermarking scheme that can withstand geometric distortions and WEAs simultaneously. Our scheme is mainly composed of three components: 1) robust mesh generation and mesh-based watermarking to resist geometric distortions; 2) construction of media hash-based content-dependent watermark to resist WEAs; and 3) a mechanism of false positive-oriented watermark detection, which can be used to determine the existence of a watermark so as to achieve a tradeoff between correct detection and false detection. Furthermore, extensive experimental results obtained using the standard benchmark (i.e., Stirmark) and WEAs, and comparisons with relevant watermarking methods confirm the excellent performance of our method in improving robustness. To our knowledge, such a thorough evaluation has not been reported in the literature before.**

***Index Terms*—Attack, embedding, false positive detection, media hash, mesh, robustness, watermark.**

I. Introduction

DIGITAL watermarking has been recognized as a helpful technology for copyright protection, traitor tracing, and authentication. No matter which kinds of applications are considered, robustness is a critical issue affecting the practicability of the watermarking system. Robustness refers to the capability of resistance to attacks that are used to destroy or remove hidden watermarks. In general, attacks were divided into four categories [36]: 1) removal attacks; 2) geometric attacks; 3) cryptographic attacks; and 4) protocol attacks. Among them, geometric attacks introduce synchronization errors in order to disable watermark detection without having to remove hidden information or degrade the quality of the watermarked contents. On the other hand, there exist watermark-estimation attacks (WEAs), including the collusion attack that can remove watermarks while making the attacked data further transparent to its original, and the copy attack that can cause protocol ambiguity within a watermarking system. Motivated by the need for sufficient robustness, this study focused on the challenging problem of resisting both (extensive) geometric attacks and WEAs, which has not been solved in the literature. Clearly, this ambitious goal distinguishes our work and existing methods.

The existing watermarking methods that are resistant to geometric attacks can be divided into three categories. The first category includes those which embed a watermark into the geometric invariant domain. In [14], [23], and [37], watermarking was conducted in the magnitude part of the Fourier–Mellin transform (FMT) to exploit its affine invariance. However, the Fourier–Mellin domain is inherently vulnerable to cropping and other local geometric distortions (e.g., changes of the aspect ratio). In addition, resistance to removal attacks is limited because most of the FMT information is contained in the phase instead of the magnitude part of the Fourier transformed domain. In [31], the watermark itself is designed to be circularly symmetric and is embedded in the Fourier transform magnitudes corresponding to a predefined set of mid-frequency coefficients. On the other hand, moment normalization [1], [17] Radon transformations [30], or Zernike moments [10] were employed to achieve geometric invariance. Their major limitation is the inability to resist attacks related to cropping because the lost contents lead to changes of moments.

The methods belonging to the second category use a template [24], [25], [32] or insert a periodic watermark pattern [12], [35] for the purpose of resynchronization. This kind of prior information is also known as the pilot signal [20]. In [24] and [25], templates were embedded in the discrete Fourier transform (DFT) domain to generate the shape of local peaks, which can be easily retrieved in the detection process to recover geometric parameters. On the other hand, the local peaks can also be easily extracted by pirates in order to remove templates [9]. In [12], Kutter was first to propose a watermarking scheme that

Manuscript received January 13, 2005; revised November 6, 2005. This work was supported in part by the National Science Council (NSC) of Taiwan, R.O.C., under Grant 92-2422-H-001-004. This paper was previously published in part in *Proc. SPIE: Security, Steganography, and Watermarking of Multimedia Contents VII (EI120)*, San Jose, CA, Jan. 2005. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Stefanos Kollias.

C.-S. Lu and C.-Y. Hsu are with the Institute of Information Science, Academia Sinica, Taipei, Taiwan 115, R.O.C. (e-mail: lcs@iis.sinica.edu.tw).

S.-W. Sun is with the Institute of Information Science, Academia Sinica, Taipei, Taiwan 115, R.O.C. and also with the Department of Electrical Engineering, National Central University, Chung-Li, Taiwan 320, R.O.C.

P.-C. Chang is with the Department of Electrical Engineering, National Central University, Chung-Li, Taiwan 320, R.O.C.

Digital Object Identifier 10.1109/TMM.2006.876300

can provide resistance to global geometrical distortions. The key step in this method is the embedding of a self-reference watermark, which is prepared in advance as a specific structural pattern, for the purpose of calibration. The main drawback is that the adopted global watermark structure can be totally destroyed by means of local geometric distortions. A more powerful approach [35] extends Kutter's scheme through block-based periodical placement of self-reference watermarks so that the Fourier magnitude spectrum of periodical watermarks is composed of regular peaks distributed all over the image. This particular feature, i.e., a lattice of peaks, provides the capability of recovering global/local geometrical distortions. Again, because the positioned periodical block-based pilot signals inherently reveal peaks in the transformed domain, hints remain that a watermark estimation attack (e.g., the collusion attack) can be used to efficiently destroy them [15].

The third category includes methods [2], [13], [22], [29], [33] which employ "feature-based watermarking." 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 detected. Among the existing feature point extraction methods, the Harris detector [5] is widely used in various applications. However, we have found that the Harris detector is still not robust enough to be used in digital watermarking [2]. This is because the Harris detector is rotation and scaling-sensitive [21]. In addition to the Harris detector-based feature detection, Bas *et al.* [2] proposed a watermark embedding scheme based on decomposing an image into several meshes, each of which is formed from three detected feature points and is embedded with a watermark that is warped from a right-angles isosceles triangle[1] of size 64×64 to match the mesh's shape in the spatial domain. In the extraction process, each mesh in the spatial domain is warped to be normalized mesh from which a watermark is extracted to correlate with the original watermark. Although their method seems to provide a certain degree of robustness, thorough robustness evaluation through a standard benchmark (e.g., Stirmark [26] and , [27]) is a lack and resistance to estimation attacks [15] that are particularly important for multiple watermark embedding approaches is also not discussed. In [33], Mexican-Hat wavelet filtering was used for feature point extraction. Mexican-Hat wavelet filtering was implemented in the frequency domain using fast Fourier transform (FFT). Although one-dimensional (1-D) FFT is widely used to implement two-dimensional (2-D) FFT in order to improve computational efficiency, this implementation may lead to another severe problem; i.e., the input coefficient of 1-D FFT is quite different from the rotated version such that different 1-D FFT filters will lead to different filtering results. This is mainly due to the fact that the asynchronization effect is propagated and coupled with the result of Mexican-Hat wavelet filtering. In [29], the scale-space theory was applied for feature point extraction. Feature points were determined through automatic scale selection and local extreme detection. For a chosen feature point, a circular disk is formed and used for embedding in the Fourier domain. However, there are two major drawbacks in [29]: 1) the embedding unit is a circular disk, which inherently limits the achievable robustness against geometric attacks that preserve the aspect ratio (this was also noted by the authors) and 2) since embedding is conducted in the magnitude component of the Fourier domain, as noted in the above discussions of the first category of methods, resistance to removal attacks is limited (this will be seen later in the comparison of experimental results).

After surveying the existing watermarking methods that provide a certain degree of robustness against geometric distortions, we have observed that 1) the methods in the first category are restricted to be affine invariant; 2) the pilot signals that are employed in the methods in the second category for the recovery of geometric parameters are easily removed; and 3) robust extraction of feature points plays a key role in the methods in the third category. In particular, we find that Voloshynovskiy *et al.*s scheme [35] was thoroughly verified by means of the standard benchmark, Stirmark [26], [27] and possesses strong robustness. Thus, we can treat Voloshynovskiy *et al.*'s scheme as a state-of-the-art, robust watermarking technology. However, as described previously, this method is vulnerable to collusion, so initially embedded watermarks can be removed and the ability to resist extensive geometric attacks can be lost. Furthermore, we are aware of a recent paper [20] in which Manuel *et al.* exhaustively analyzed pilot-based synchronization algorithms and confirmed that pilot signals are easy to destroy. As a consequence, we do not adopt the paradigm of pilot-based watermarking even though it exhibits promising robustness against geometric attacks. Since the purpose of this paper is to propose an image watermarking scheme that can resist extensive geometric attacks and the watermark estimation attacks [15] simultaneously, we adopt feature-based watermarking based on the possibility [19] that the robustness of feature point extraction can be enhanced. Moreover, in our companion paper [15], we proposed a block-based content-dependent watermarking scheme that combines our content-dependent watermark with the approach in [35] to tolerate the watermark estimation attacks. We also provided statistical analysis for the anti-disclosure content-dependent watermark to show its ability in resisting WEAs. However, the preset periodical regularity of a watermark pattern is destroyed, thus, resistance to geometric distortions is lost because the content-dependent watermarks resulting from all the image blocks are dissimilar. In order to further address this issue, we investigate mesh-based instead of block-based watermarking in this paper.

In this paper, we propose to use the Gaussian kernel as the preprocessing filter to stabilize the feature points. The Gaussian kernel is a circular and symmetric filter in that all the neighboring information of a pixel can be equally used to filtering, leading to geometric-invariant filtering. In order to resist watermark-estimation attacks, image hashing [19] is further extracted and combined with hidden watermarks to generate the media hash-based content-dependent watermark (CDW) [15]. CDW is able to resist the watermark estimation attacks because even though pirates can estimate watermarks from meshes, they still cannot be successfully colluded to generate an even more correct watermark that is to be removed. We also study how mesh-based watermarking can be achieved without causing perceptual

[1]In this paper, we call the domain, where the shape of either a watermark or a mesh is transferred to become a right-angles isosceles triangle, as the normalized domain.

quality degradation. In addition to robustness, due to the unique characteristic of multiple mesh-based watermark embedding, we propose a false positive-oriented watermark detection mechanism to indicate the presence/absence of a watermark. We investigate how to determine the existence of a watermark in a mesh and in an image, respectively. In order to demonstrate the performance of our method in improving robustness, the standard benchmark, Stirmark, and watermark estimation attacks (including the collusion and copy attacks) were used to perform a thorough evaluation.

The proposed method follows the framework of [2] in that a watermark is embedded and extracted from an image unit—mesh. However, there are a number of significant contributions that our paper describes. First, we investigate some important issues in Section II to improve the robustness. In particular, the common weakness of existing multiple watermark embedding approaches that are fragile against watermark estimation attacks [15] has been solved. Second, we find from [2] that the watermark signal is warped from the normalized domain to the spatial domain for embedding, while the extraction process is operated in the normalized domain. However, this asymmetric embedding and extraction paradigm cannot be used to achieve the goal of anti-estimations. This is because a pair of a watermark and a media hash is needed to be integrated to form a content-dependent watermark, as will be described in Section II-B, and the lengths of all media hashes must be kept the same. In this situation, the watermark embedding and extraction processes of our method are both performed in the normalized domain, as will be described in Section III. In addition, the modified coefficients in the normalized domain are warped to the spatial domain to accomplish embedding. Third, in Section IV, we develop a false positive-oriented watermark detection mechanism so that a reliable detection can be accomplished, and the tradeoff between correct detection and false detection can be more successfully guaranteed. Fourth, extensive experimental results together with robustness comparisons with other feature-based methods are given in Section V to verify the excellent performance of our scheme. Finally, conclusions are drawn in Section VI.

II. Robust Feature Extraction and Media Hash-Based Content-Dependent Watermark

Two issues concerning the proposed watermarking method, robust feature extraction and media hash-based content-dependent watermark, will be discussed in this section. They play key roles in achieving the desired goal.

A. Robust Feature Extraction

Since our watermarking method is mesh-based, feature point extraction needs to be robust enough to approximately tolerate common filtering, compression, and geometric attacks for robust mesh generation. In our method, Gaussian kernel filtering, local maximum determination, and scale determination are integrated for designing a robust feature point extraction algorithm.

1) Gaussian Kernel Filtering: Gaussian kernel filtering is a special case of scale-space filtering. In scale-space filtering, an image is filtered by several filters of different sizes to generate multiple frequency responses. In some applications, the filter size can be adaptive to different affine transformation environments. But in digital watermarking, we only select a fixed filter size to generate one level scale-space for watermark embedding. This benefits our watermark detection scheme in that only a small set of filters is required to achieve blind detection (as will be described in Section III-B). Let $I(x,y)$ be a cover image, and let the Gaussian kernel be defined as

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

where σ is the standard deviation. The convolution of the Gaussian kernel and the cover image is defined as

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

Because the Gaussian kernel is isotropic and circular in shape, the resultant filtering response is rotation invariant [21], [38], which is beneficial for obtaining geometric-invariant feature points. In order to show the desired rotation insensitivity, we will illustrate an example in Fig. 2 to compare the feature point extractors presented here and [2].

Let the Gaussian kernel be represented with at least k times of standard deviation, which is described as a 2-D filter of size $(2k*\sigma+1)\times(2k*\sigma+1)$. For a Gaussian distribution, the probabilities [7] within one, two, and three standard deviations of its mean are about 68%, 95%, and 99.7%, respectively. Since three standard deviations of mean can sufficiently represent the energy of a Gaussian distribution, $k=3$ is adopted here.

2) Local Maximum Determination: The local maximum determination process is operated in the Gaussian kernel filtered signal for feature point extraction. First, a maximum filter of size 3×3 is applied to $L(x,y,\sigma)$ and is expressed as

$$MF(x,y) = \max_{(x_t,y_t)\in(N_8(L(x,y,\sigma))\cup L(x,y,\sigma))} \{L(x_t,y_t,\sigma)\} \tag{1}$$

where $N_8(L(x,y,\sigma))$ denotes the 8-neighborhood of $L(x,y,\sigma)$. Next, the set of feature points is determined as

$$P = \{(x,y)|MF(x,y) = L(x,y,\sigma)\} \tag{2}$$

which means that a feature point at (x,y) satisfies that the filtering responses, $MF(x,y)$ and $L(x,y,\sigma)$, are equal. In addition, the set of feature points, P, is used to form a set of meshes by means of the Delaunay tessellation. In this paper, each mesh is a basic unit used for watermark embedding and extraction.

3) How Can We Choose σ?: When the Gaussian kernel is used as the feature point detector, it is important to determine how many σ's have to be used. If a larger σ is used, lower frequency (corresponding to larger scale) information tends to be revealed. On the other hand, high frequency (smaller scale) information can be detected when a smaller σ is used. Therefore, which σ should be used is an important issue. The selection of σ's is also related to the ability to deal with geometric attacks

TABLE I
NUMBER OF DETECTED FEATURE POINTS AT DIFFERENT SCALES (σ's) AND THE DETERMINED σ_s's FOR THE IMAGE LENA OF DIFFERENT SIZES

image size	$\sigma=2$	$\sigma=3$	$\sigma=4$	$\sigma=5$	$\sigma=6$	σ_s
128×128	20	6	2	-	-	4
256×256	55	18	6	2	-	5
512×512	224	55	19	6	2	6

because if the σ's do not properly match the characteristics of geometrically attacked images, then the feature points will not be correctly detected.

These problems can be dealt with by observing the number of feature points across different σ's (ranging from 2 to 5) for different image sizes (up to 512×512), as shown in Table I. Since at least three points are required to form a mesh, we need to choose σ's that can produce at least three feature points. Let σ_s be the largest value that cannot generate at least three feature points. In addition, the number of feature points cannot be so large as to yield small meshes such that a watermark cannot be completely embedded. According to Table I, the value of σ_d that can be effective for watermark embedding is set to $\sigma_s - 3\ (\geq 1)$, which is defined as a detection scale. For example, for a 512×512 image, $\sigma_d = 6 - 3 = 3$ is adopted.

B. Content-Dependent Watermark

Some researchers [2], [29], [33], [35] have proposed inserting multiple redundant watermarks into an image in the hope that this will suffice to maintain resistance to geometric distortions as long as at least one watermark exists. The common framework is that certain types of image units, such as blocks [35], meshes [2], or disks [29], [33], are 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, we argue that once the hidden watermarks are successfully estimated by means of a collusion attack, the ability to resist geometric distortions become weaker such that the false negative problem occurs. Of particular interest is the possible quality improvement of attacked media data that can be achieved by means of collusion attack. In addition, copy attack can also efficiently defeat a watermarking system by creating ambiguity problems. Since the common operation involving in both collusion and copy attacks is watermark estimation, they are called WEAs [15].

To withstand watermark-estimation attack, the key is to make the embedded watermarks different so that the hidden watermark cannot be approximately estimated by means of collusion. To this end, we propose to embed a media hash-based CDW, which is composed of a watermark and a media hash. Our analyses [15] show that CDW is able to resist both copy and collusion attacks. Here, the block-based content-dependent watermark [15] is introduced. Each block of size $L_B \times L_B$ is divided into subblocks of size $L_{\text{sub}} \times L_{\text{sub}}$, and a block-pair relation is created by means of a secret key (the key is the same as that used to generate the watermark). Before hash generation, all subblocks are DCT transformed. For a pair of $L_{\text{sub}} \times L_{\text{sub}}$ blocks, a hash bit, defined as the magnitude relationship between two AC coefficients, is represented as

$$\text{MH}(b) = \begin{cases} 1, & \text{if } |f_k(p_1)| - |f_l(p_2)| \geq 0 \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

where $\text{MH}(\cdot)$ is a hash bit in a hash sequence MH, and $f_k(p_1)$ and $f_l(p_2)$ are two AC coefficients at positions p_1 and p_2 in $L_{\text{sub}} \times L_{\text{sub}}$ blocks k and l, respectively. Given a pair consisting of hash MH and watermark W, a media hash-based content-dependent watermark can be generated as

$$\text{CDW} = S(W, \text{MH}) \tag{4}$$

where $S(\cdot)$ is a shuffling function, which is basically application-dependent and will be used to control the combination of W and MH. In our implementation, MH is first shuffled and the shuffled MH is used to disorder W. Here, a transposition cipher [4], [11] is used to change the position of a hash sequence MH with the aim of retaining robustness against bit errors (without error propagation). A more secure block cipher (e.g., the well-known DES algorithm) is not adopted due to its fragility to a single bit error. The signal CDW is the watermark that we want to embed into a local region.

It should be noted that the robustness of media hash MH against attacks is crucial for CDW to successfully resist attacks. Robust media hashing can also be applied for content authentication, copy detection, and identification [19]. On the other hand, we provided in [15] the statistical analysis of CDW to confirm its ability in resisting WEAs. Since these issues are beyond the theme of this paper, readers should refer [15], [19] for more details.

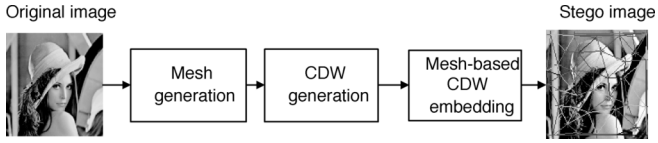

Fig. 1. Block diagram of the embedding process.

III. PROPOSED WATERMARKING METHOD

In this section, the proposed media hash-based content-dependent watermarking scheme, which encompasses the watermark embedding process, watermark extraction process, and complexity analysis, is described.

A. Watermark Embedding

The watermark embedding process is outlined in Fig. 1. In this paper, the hidden watermark W is generated with a secret key and is a bipolar sequence of length L_W, i.e., $W = \{W_j\}_{j=1,2,\cdots,L_W}$ with each $W_j \in \{-1, +1\}$.

1) Mesh Generation: The first step in mesh generation is to filter a cover image using Gaussian filtering, as described in Section II, so that a set of feature points P can be obtained. Next, the Delaunay tessellation is performed using P to generate a set of meshes, $M = \{M_i\}_{i=1,2,\cdots,L_M}$, where L_M denotes the

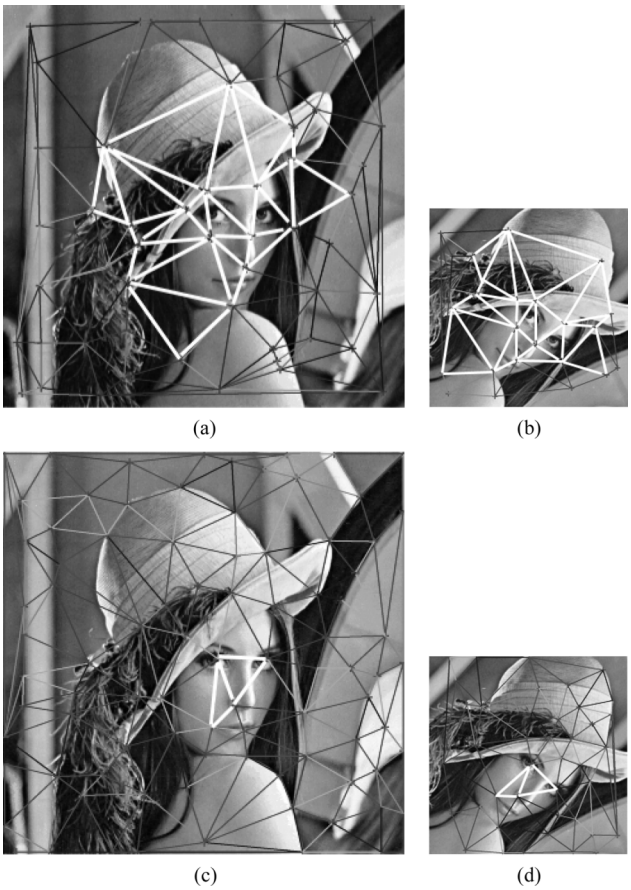

Fig. 2. Illustration of feature point extraction and mesh generation: (a) and (b) are obtained using our method and (c) and (d) are obtained using [2]. (b) and (d) are the rotated and cropped versions of their original images (a) and (c), respectively.

number of meshes extracted from a cover image. Each M_i is a basic unit used for watermark embedding and extraction.

Fig. 2 shows the comparisons of feature point extraction and mesh generation between our technique and [2]. By using our technique, Fig. 2(a) and (b) shows the detected feature points and the resultant meshes with respect to the original and rotated images. Similarly, Fig. 2(c) and (d) shows the results obtained from [2]. In Fig. 2, the triangular meshes with white bold lines represent those meshes that can be found from both the original and rotated images. Obviously, the use of Gaussian kernel for feature point extraction achieves the goal of rotation insensitivity, which is especially desired in digital watermarking.

2) Content-Dependent Watermark Generation: The content-dependent watermark generation process, including: 1) mesh normalization; 2) media hash extraction; and 3) hash-based content-dependent watermark, will be described in the following.

a) Mesh normalization: Before embedding is performed, each triangle mesh has to be normalized to obtain a canonical form. Here, a mesh normalization process is performed to affine transform each extracted mesh M_i to obtain a right-angled isosceles triangle, which is called a normalized mesh, NM_i. The goals are not only to extract a fixed-length hash, but also to reduce the effect of image content shifting caused by the imperfect extraction of feature points. If the watermark signals are embedded in the spatial domain, the shifting problem, even with slice loss or pixel loss, may cause the watermark extraction process to fail. Therefore, the size of a normalized mesh needs to be properly determined. Our empirical research has shown that if a larger region is warped into a small region, which means that the warping process is a multiple-to-one pixel mapping, then one pixel in NM_i represents several pixels in M_i. Under this circumstance, fewer pixels in NM_i will be affected by slice missing or shifting, which implies that a small normalized mesh of small size is beneficial for achieving robustness. In this study, the size of a normalized mesh is empirically found to be 48×48 for achieving a tradeoff between transparency and robustness (this choice will become clear in the next two paragraphs). Let $\mathrm{NM} = \{\mathrm{NM}_i\}_{i=1,2,\cdots,L_M}$ denote the set of normalized meshes.

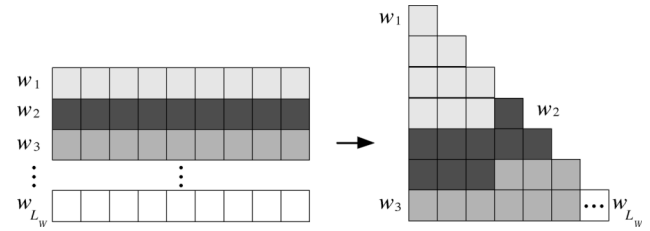

Fig. 3. Repeated watermark bits (left)—each bit is repeated nine times—are arranged and embedded in a normalized mesh (right).

b) Mesh-based hash extraction: A mesh-based media hash, MH_i, is extracted from each normalized mesh NM_i, as described in Section II-B. Since this paper investigates a mesh-based watermarking scheme, each normalized mesh prior to hash extraction needs to be transformed into a block. Here, each normalized mesh is flipped and then the flipped mesh is padded with the original version to form a block. If we set $L_B = 48$ and $L_{\mathrm{sub}} = 6$, then the length of a hash sequence is 64.

c) Media hash-based content-dependent watermark: In this paper, the watermark length (L_W) is set to be 128 bits. Although the length of the media hash (MH_i) is 64 bits, by repeating it two times, a media hash of 128 bits can be generated. Then, each media hash MH_i and watermark W are combined (4) to generate the content-dependent watermarks, i.e., $\mathrm{CDW} = \{\mathrm{CDW}_i\}_{i=1,2\cdots L_M}$. Although only one watermark W is embedded for a cover image, the principle behind CDW leads to different signals embedded in different meshes.

3) Arrangement of Watermark Bits for Embedding: Since the length of a content-dependent watermark is 128 and the size of a normalized mesh is $(48 \times 48)/2 = 1152$, we propose to repeatedly embed the watermark to enhance robustness, as shown in Fig. 3. It is not hard to see that the time of repetition is $\lfloor (1152/128) \rfloor = 9$. Let $R9\mathrm{CDW} = \{R9\mathrm{CDW}_i\}_{i=1,2\cdots L_M}$, where each element of CDW_i is repeated nine times to form $R9\mathrm{CDW}_i$. This repeated embedding is very important for achieving better robustness, in particular when the mesh is (slightly) perturbed because its constituent feature points are not exactly the same as the ones detected in the embedding process. In other words, the feature extraction error and other numerical errors such as interpolation errors and rounding errors will affect the watermark detection performance. In order to deal efficiently with these problems, the repeated embedding of a watermark bit is performed [6], [15], [16], [28]. Recall that

in [29] the authors proposed to deal with this problem through locally searching (75 times) for the possibly correct feature point in the neighborhood of the detected point.

In summary, it can be observed that the watermark's length, the hash's length, and the normalized mesh's size are all designed in a sophisticated way to satisfy the embedding purpose so that robustness can be better achieved.

4) Mesh-Based Embedding: In order to maintain transparency after performing watermarking, we adopt the noise visibility function (NVF) [34], which is an image-dependent visual model. Content-adaptive watermark embedding is designed to insert watermarks into the cover image $I(,)$ to form a stego image $I^w(,)$ as follows:

$$I^w(x,y) = I(x,y) + (1 - \mathrm{NVF}(x,y)) \cdot w_j \cdot S + \mathrm{NVF}(x,y) \cdot w_j \cdot S_1 \quad (5)$$

where S and S_1 denote the watermark strength, and w_j is an element of a bipolar watermark signal. In [34], the authors proposed to set S_1 to 3 for most real-world and computer-generated images. As for S, it can be adjusted to keep the peak signal-to-noise ratio (PSNR) higher than a certain value. In our method, $S_1 = 3$ is adopted, and S is adjusted to keep the PSNRs all at about 38 dB. Therefore, in our watermarking scheme, the watermark embedding process can be designed as

$$\begin{cases} \mathrm{NM}_i^w(x,y) = \mathrm{NM}_i(x,y) + (1 - \mathrm{NVF}(x,y)) \cdot \\ r9cdw_{ij} \cdot S + \mathrm{NVF}(x,y) \cdot r9cdw_{ij} \cdot S_1 \end{cases} \quad (6)$$

where $r9cdw_{ij}$ denotes the jth watermark element of $R9\mathrm{CDW}_i$, which is embedded in NM_i. Once the watermarked normalized mesh NM_i^w is obtained, the inverse normalization process is used to yield a watermarked mesh. Although "direct inverse normalization" is intuitive, transparency may be degraded because blocking effects are caused by the one-to-multiple pixel mapping. To deal with this problem, the difference between NM_i and NM_i^w, i.e., the second term on the right-hand side of (6), which is caused by watermarking in the normalized domain, is inversely normalized to yield the difference M_i^{diff} in the spatial domain. Hence, the watermarked mesh in the spatial domain can be obtained as

$$M_i^w = M_i + M_i^{\mathrm{diff}}. \quad (7)$$

Based on (7), the original high-frequency components of M_i can be preserved to maintain transparency. Finally, by integrating all watermarked meshes, we can obtain the stego image.

In order to illustrate the advantage of our embedding method (7) over inverse normalization (6), an example is shown in Fig. 4 for visual comparison. Fig. 4(a) shows a stego Lena image that is generated through inverse normalization of watermarked meshes. Many interpolation errors and blocky effects can be observed. On the other hand, if the embedded signal in the normalized domain is transformed back to the spatial domain and then added to the original image, then as Fig. 4(b) shows, the visual quality is not perceptually degraded.

(a) (b)

Fig. 4. Transparency comparison for watemarked Lena images based on (a) direct inverse normalization of watemarked meshes [see(6)], $\mathrm{PSNR} = 28.99$ dB and (b) inverse normalization of the embedded signal [see (7)] plus the original image, $\mathrm{PSNR} = 39.87$ dB.

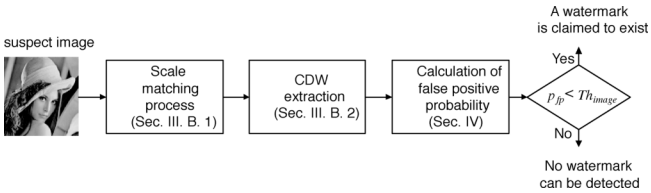

Fig. 5. Block diagram of the process of determining the existence of a watermarked.

B. Watermark Extraction

The process of determining the existence of a watermark is depicted in Fig. 5. Basically, the watermark extraction process is the inverse process of watermark embedding.

1) Scale Matching Process: In the watermark extraction end, the first step is to determine σ's that will be used for filtering (Section II-A). Initially, σ_d as determined in the embedding end can be used; however, due to possible modifications of the stego image, a single value, σ_d, cannot be guaranteed to match the characteristics of the encountered attacked images. In order to tolerate varied attacks, in addition to σ_d, other σ's may be needed. Some scenarios that will change the size of an image are described in the following to prove the need for several σ's. If the size of a stego image is changed due to cropping (e.g., rotation + cropping), then σ_d will fail to capture the characteristics of the cropped images because it cannot distinguish between scaling and cropping that lead to changes of the images' sizes. On the other hand, for a huge image, the watermark embedding and extraction processes should be operated in a tiling manner. The tile size selected in our proposed scheme is 512×512, which always sets σ_d to a fixed value 3, as described in Section II-A3. If the huge image is scaled down or up, then σ_d will be useless for capturing this change. Therefore, a scale matching process is proposed here to help us determine proper σ's for filtering.

First of all, we have to know the possible range of change of an image's size. Let us take the standard benchmark, Stirmark [26], [27] as an example. For all non-geometric attacks, scaling, and other attacks that cause slight changes of an image's size, σ_d as determined in the embedded process can be used. For those attacks that have cropping effects (in Stirmark, rotation of 45° and cropping of more than 50% cause severer cropping

effects), the size of an image could be quartered. Under these circumstances, $\sigma_d + 1$ instead of σ_d needs to be used. Here, let $\sigma_d + 1$ be written as σ_{d+1}.

On the other hand, in the case of a huge image, it is not known whether the contents contained within a tile have been attacked or not. When we consider the modifications caused by scaling with factors ranging from 50%$\sim$200% (as provided in Stirmark), it is not hard to see that $\sigma_{d-1} = \sigma_d - 1, \sigma_d$, and σ_{d+1} are necessary to adapt to various tile sizes.

In summary, three filtering parameters, σ_{d-1}, σ_d, and σ_{d+1}, are required for filtering to extract the desired feature points under the constraint that Stirmark is considered for possible attacks. Of course, more filtering parameters can be used at the cost of more time spent to deal with attacks that cause severer effects. Here, let M_{d-1}, M_d, and M_{d+1}, respectively, denote the sets of meshes extracted using σ_{d-1}, σ_d, and σ_{d+1}.

2) Media Hash-Based Content-Dependent Watermark Extraction: The proposed content-dependent watermark extraction process is depicted in Fig. 6. The normalization process is used to, respectively, transform the three sets of meshes, M_d, M_{d+1}, and M_{d-1}, into corresponding sets of normalized meshes, $\mathrm{NM_d}$, NM_{d+1}, and NM_{d-1}, from which three sets of media hashes, MH_d, MH_{d+1}, and MH_{d-1}, can be extracted.

In this paper, Wiener filtering is used to blindly extract the hidden signal. Wiener filtering is considered to be an efficient method [8], [15], [35] because the watermark is usually a high-frequency signal. Let $R9\mathrm{CDW}^e_{di}$, $R9\mathrm{CDW}^e_{d+1i}$, and $R9\mathrm{CDW}^e_{d-1_i}$ be, respectively, extracted from NM_{di}, NM_{d+1i}, and NM_{d-1i}. Since the watermark bits are redundantly embedded, a bit is finally determined based on a majority selection rule. In this paper, each bit is repeatedly embedded into a mesh nine times. For an embedded bit, if most of its corresponding extracted bits are $1(-1)$, then the extracted bit is finally determined to be $1(-1)$. Let CDW^e_{di}, CDW^e_{d+1i}, and CDW_{d-1i} be the extracted watermarks after the majority determination process is completely.

Next, three sets of extracted media hashes, MH_d, MH_{d+1}, and MH_{d-1}, corresponding to σ_d, σ_{d+1}, and σ_{d-1}, respectively, are separated from their corresponding watermarks, CDW^e_{di}, CDW^e_{d+1i}, and CDW_{d-1i}, as follows:

$$W^e_d = \{W^e_{d\,i}\}_{i=1,2\cdots L_M},$$
$$W^e_{d\,i} = (\mathrm{CDW}^e_{di}/\mathrm{MH}_{di}) \tag{8}$$
$$W^e_{d+1} = \left\{W^e_{d+1i}\right\}_{i=1,2\cdots L_M},$$
$$W^e_{d+1i} = \left(\mathrm{CDW}^e_{d+1i}/\mathrm{MH}_{d+1i}\right) \tag{9}$$
$$W^e_{d-1} = \left\{W^e_{d-1i}\right\}_{i=1,2\cdots L_M},$$
$$W^e_{d-1i} = \left(\mathrm{CDW}^e_{d-1i}/\mathrm{MH}_{d-1i}\right). \tag{10}$$

Thus, we obtain the extracted watermark signals W^e_d, W^e_{d+1}, and W^e_{d-1}.

C. Complexity of Our Method

The complexity of our watermarking algorithm, as depicted in Fig. 1 and Fig. 6, is actually dominated by mesh normalization because it involves the most operations among all the steps of our method. More specifically, the number of arithmetic operations for pixel transformation during mesh normalization is

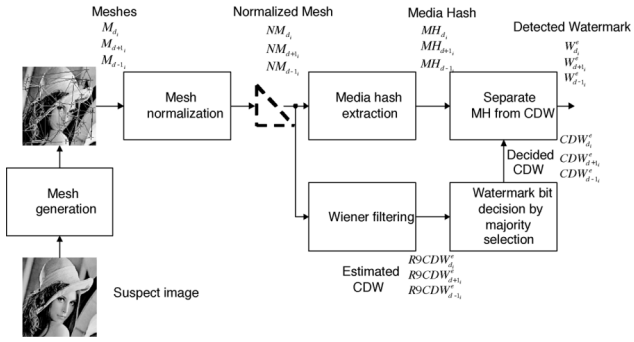

Fig. 6. Block diagram of the CDW extraction process.

constant and is proportional to the number of pixels in a mesh. Since the total number of pixels in all meshes that are required to execute normalization is approximately equal to the size of an image, as a result, it can be concluded that the time complexity of mesh normalization is proportional to image's size. Basically, the complexity of mesh normalization during embedding in our method and Bas *et al.*'s method [2] is the same, but our execution time is twice of theirs (see the explanations in the last paragraph of Section I, and Section III-A). As for the watermark extraction process, it is basically the inverse process of watermark embedding and, thus, both possess the same time complexity. However, in order to deal with scaling, as described in Section III-B1, the execution time of our watermark extraction process becomes three times longer than [2].

IV. False Positive-Oriented Determination of the Existence of a Watermark

In order to indicate the presence/absence of a watermark in an image, the first step is to determine whether a watermark exists in a mesh. For each NM_{di} (or NM_{d+1i}, NM_{d-1i}), the bit-error rate (BER) between W and $W^e_{d\,i}$ (or W and W^e_{d+1i}, W and W^e_{d-1i}) is calculated. If the BER is smaller than a threshold $\mathrm{Th_{mesh}}$, it is said that a watermark exists in a mesh. The threshold $\mathrm{Th_{mesh}}$ needs to be determined by considering the false positive factor because to claim the robustness of a watermarking system is meaningful only when the false positive probability is taken into consideration in measuring robustness. In this study, the bit-detection process is treated as an independent random Bernoulli trail with probability p_b, which is the probability that the bit b (-1 or 1) will occur, and is considered to always be 0.5 here. Theoretically, the probability of truly detecting a watermark in a mesh when $\mathrm{BER} \le \mathrm{Th_{mesh}}$ holds can be represented as

$$p_{Ms} = \sum_{j=(L_W - L_W \times \mathrm{Th_{mesh}})}^{L_W} \binom{L_W}{j} p_b^j (1-p_b)^{L_W - j}. \tag{11}$$

Equation (11) also specifies the probability that a watermark can be found in a mesh that has not, in fact, been watermarked. As a result, determining the threshold $\mathrm{Th_{mesh}}$ is important.

In order to reasonably determine $\mathrm{Th_{mesh}}$, p_{Ms} in (11) should be consistent with practical results. To this end, the BERs obtained from extensive "sequence-pair" comparisons were collected. A sequence-pair is composed of the watermark known

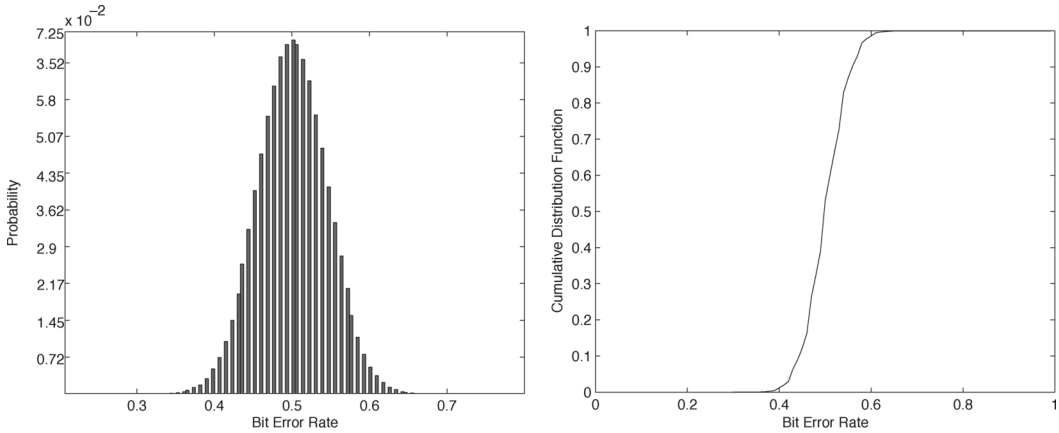

Fig. 7. Sequence-pair comparisons (one is the watermark and the other one is one of the signals extracted from the Corel image database: (a) distribution of the BERs and (b) cumulative distribution of (a).

by the owner and a signal that is extracted from one of the meshes in a random image. First of all, every unwatermarked image chosen from the Corel image database was applied as the input to our watermark detection process, as described in Section III-B. For each image, a set of BERs could be obtained after sequence-pair comparisons were performed. After testing all 20 000 images in the Corel image database, we obtained the BER distribution and its cumulative distribution shown in Fig. 7. Based on this information, if $\mathrm{Th}_{\mathrm{mesh}}$ is chosen to be 0.375, then p_{Ms} in (11) is calculated to be 0.003, which is very close to the cumulative distribution function (cdf), $cdf(BER \leq 0.375) = 0.0027$, of the BER distribution measured using the Corel image database. Consequently, it can be concluded that $\mathrm{Th}_{\mathrm{mesh}} = 0.375$ is a reasonable choice.

On the other hand, there are three vertexes in each M_i. However, some geometric attacks may change the relationship between the three vertexes, which is crucial for mesh normalization. In order to deal with this problem, we do not merely detect a watermark from one possible normalized mesh; instead, $6(= 3!)$ possible normalized meshes are all fed into the watermark extraction process. Thus, the probability of detecting a watermark in a mesh, p_M, can be derived as

$$p_M = (p_{Ms})^1 \times (1 - p_{Ms})^5 \approx p_{Ms} \tag{12}$$

which is still numerically close to p_{Ms}, as derived in (11). On the other hand, the probability of failing to detect a watermark is derived as $p_{\mathrm{unwatermarked}} = 1 - (p_{Ms})^1 \times (1 - p_{Ms})^5$.

So far, we have discussed how one can determine the existence of a watermark in a mesh. Now, we will proceed to explain how one can determine the existence of a watermark in an image by incorporating the mesh-based detection results. Recall that L_M is the number of meshes in an image (no matter whether it is attacked or not). Let D_M be the number of meshes found to have been watermarked, as described in the above paragraphs. The probability of determining that a suspect image was watermarked before is derived as

$$p_{\mathrm{fp}} = \sum_{i=D_M}^{L_M} \binom{L_M}{i} {p_M}^i (1 - p_M)^{L_M - i} \tag{13}$$

based on the constraint that D_M out of a total L_M of meshes are regarded to having been watermarked. In fact, (13) also reveals the probability that a random image will be "wrongly" determined as having been watermarked. Furthermore, this also implies that different attacks lead to different p_{fp}'s; i.e., a more challenging attack will generate a higher false positive probability.

In order to claim the presence of a watermark with strong confidence (without causing a non-negligible false positive), p_{fp} should be low. On the other hand, p_{fp} should be large to achieve robustness. Here, a reasonable threshold, $\mathrm{Th}_{\mathrm{image}}$, is required to satisfy the tradeoff between robustness and false positive. Again, the Corel image database was adopted here to derive $\mathrm{Th}_{\mathrm{image}}$. Every unwatermarked image chosen from the Corel image database was applied as the input to our watermark detection process. For each image, one p_{fp} was obtained based on (13). By integrating all the p_{fp}'s, the cumulative distribution function showed that $cdf(p_{\mathrm{fp}} \leq (3.50e-004)) = 0$ and $cdf(p_{\mathrm{fp}} \leq (4.00e{-}004)) = (6.28e{-}005)$. Thus as a guideline, it is helpful to set the threshold $\mathrm{Th}_{\mathrm{image}}$ to $3.50e{-}004$ according to the information obtained from the Corel image database. It should be noted that although meshes are adopted in this paper, similar results can be obtained using other types of image units, such as blocks or disks.

It should be noted that since three σ's are employed for watermark detection, three p_{fp}'s are generated. The smallest value will be chosen as the final p_{fp} (corresponding to the largest D_M).

A. Comparison With Other Methods

In this section, some recent papers that have proposed feature-based watermarking methods will be discussed. False positive probability analysis was also conducted in [29] and [33], which proposed to embed watermarks into disks that are extracted from an image. However, the existence of a watermark was not finally determined by taking the derived false positive probability into consideration. On the contrary, these authors only indicate the number of disks (out of the number of total disks) that can be found to contain the hidden watermark.

TABLE II
NONGEOMETRIC ATTACKS FOR BABOON

attack	proposed method		[29]		[33]	
	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}
Median filter 2x2	8/106	1.52e-009	-	-	6/11	7.07e-013
Median filter 3x3	6/111	1.26e-006	-	-	2/11	6.24e-004
Median filter 4x4	6/109	1.13e-006	1/100	1.91e-005	-	-
Gaussian filter 3x3	8/108	1.77e-009	0/100	1.00e-000	8/11	2.94e-018
JPEG 90	5/111	2.39e-005	-	-	-	-
JPEG 80	7/115	7.22e-008	-	-	9/11	3.35e-021
JPEG 70	4/115	4.30e-004	1/100	1.91e-005	11/11	7.10e-028
JPEG 60	6/103	8.13e-007	1/100	1.91e-005	7/11	1.72e-015
JPEG 50	5/110	2.29e-005	1/100	1.91e-005	5/11	2.07e-010
JPEG 40	4/113	4.02e-004	1/100	1.91e-005	7/11	1.72e-015
JPEG 30	5/114	2.72e-005	0/100	1.00e-000	4/11	4.34e-008
JPEG 20	4/106	3.15e-004	-	-	-	-
JPEG 10	1/124	3.11e-001	-	-	-	-
FMLR	6/106	9.62e-007	4/100	5.30e-021	-	-
Color reduce	8/109	1.90e-009	2/100	1.82e-010	4/11	4.34e-008
Sharpening 3x3	4/120	5.04e-004	0/100	1.00e-000	2/11	6.24e-004

TABLE III
NONGEOMETRIC ATTACKS FOR LENA

attack	proposed method		[29]		[33]	
	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}
Median filter 2x2	28/110	1.98e-045	-	-	1/8	2.69e-002
Median filter 3x3	16/111	2.68e-022	-	-	1/8	2.69e-002
Median filter 4x4	16/102	6.40e-023	5/100	1.95e-026	-	-
Gaussian filter 3x3	23/103	4.00e-036	3/100	1.14e-015	5/8	2.53e-011
JPEG 90	32/106	1.97e-054	-	-	-	-
JPEG 80	35/104	2.38e-061	-	-	6/8	4.32e-014
JPEG 70	31/104	1.40e-052	3/100	1.14e-015	7/8	4.22e-017
JPEG 60	26/111	3.08e-041	3/100	1.14e-015	6/8	4.32e-014
JPEG 50	15/111	1.49e-020	1/100	1.91e-005	5/8	2.53e-011
JPEG 40	21/117	6.22e-031	1/100	1.91e-005	3/8	2.18e-006
JPEG 30	18/116	1.65e-025	0/100	1.00e-000	2/8	3.19e-004
JPEG 20	7/100	2.74e-008	-	-	-	-
JPEG 10	1/114	2.90e-001	-	-	-	-
FMLR	3/97	3.23e-003	1/100	1.91e-005	-	-
Color reduce	29/104	2.59e-048	4/100	5.30e-021	7/8	4.22e-017
Sharpening 3x3	18/115	1.40e-025	1/100	1.91e-005	4/8	9.29e-009

TABLE IV
NONGEOMETRIC ATTACKS FOR PEPPER

attack	proposed method		[29]		[33]	
	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}
Median filter 2x2	38/108	2.33e-067	-	-	1/4	1.35e-002
Median filter 3x3	40/107	4.12e-072	-	-	1/4	1.35e-002
Median filter 4x4	24/109	1.83e-037	4/100	5.30e-021	-	-
Gaussian filter 3x3	36/108	7.07e-063	5/100	1.95e-026	1/4	1.35e-002
JPEG 90	39/111	4.63e-069	-	-	-	-
JPEG 80	44/109	5.36e-081	-	-	3/4	1.57e-007
JPEG 70	44/107	1.90e-081	6/100	5.93e-032	3/4	1.57e-007
JPEG 60	33/106	1.33e-056	6/100	5.93e-032	1/4	1.35e-002
JPEG 50	30/108	7.24e-050	4/100	5.30e-021	3/4	1.57e-007
JPEG 40	27/111	2.92e-043	4/100	5.30e-021	1/4	1.35e-002
JPEG 30	20/112	1.75e-029	4/100	5.30e-021	0/4	1.00e-000
JPEG 20	9/118	1.31e-010	-	-	-	-
JPEG 10	2/115	4.72e-002	-	-	-	-
FMLR	11/101	2.20e-014	0/100	1.00e-000	-	-
Color reduce	54/109	2.44e-105	2/100	1.82e-010	1/4	1.35e-002
Sharpening 3x3	21/117	6.22e-031	5/100	1.95e-026	4/4	1.34e-010

In [29, eq. (23)], the false positive probability for one image was defined as follows:

$$P_{FA-\text{image}} = \sum_{i=\mu}^{N} \binom{N}{i} (P_{FA-\text{disk}})^i (1 - P_{FA-\text{disk}})^{N-i} \tag{14}$$

where the watermark is detected from at least μ disks and N is the number of disks in an image that are available for watermarking. In their method, $N = 100$.

In [33], the false positive probability derived from each disk was defined in (5) of their paper as follows:

$$\begin{cases} P_{False-alarm\ on\ one\ disk} \\ = \sum_{r_1=n, r_2=n}^{r_1=T_1, r_2=T_2, r_1+r_2 \geq T} \left(\frac{1}{2}\right)^n \\ \cdot \left(\left(\frac{n!}{r_1!(n-r_1)!} \right) \cdot \left(\frac{1}{2}\right)^n \cdot \left(\frac{n!}{r_2!(n-r_2)!} \right) \right. \end{cases} \tag{15}$$

where $n = 16$, $T_1 = 10$, $T_2 = 10$, and $T = 24$. When the parameters are substituted into (15), $P_{False-alarm\ on\ one\ disk} =$

TABLE V
GEOMETRIC ATTACKS FOR BABOON

attack	proposed method		[29]		[33]	
	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}
1 column, 1 row removed	5/111	2.39e-005	-	-	-	-
5 column, 1 row removed	11/115	9.47e-014	-	-	6/11	7.07e-013
1 column, 5 row removed	6/111	1.26e-006	-	-	-	-
17 column, 5 row removed	3/106	4.14e-003	1/100	1.91e-005	3/11	6.37e-006
5 column, 17 row removed	8/100	9.55e-010	-	-	-	-
Cropping 1% off	10/110	2.11e-012	-	-	-	-
Cropping 2% off	4/112	3.89e-004	-	-	-	-
Cropping 5% off	5/110	2.29e-005	-	-	2/11	6.24e-004
Cropping 10% off	6/96	5.36e-007	-	-	2/11	6.24e-004
Cropping 15% off	7/526	1.21e-003	4/100	5.30e-021	-	-
Cropping 20% off	5/87	7.32e-006	-	-	-	-
Cropping 25% off	5/411	8.52e-003	1/100	1.91e-005	-	-
Cropping 50% off	13/648	1.38e-007	-	-	-	-
Cropping 75% off	3/138	8.56e-003	-	-	-	-
Linear(1.007, 0.010, 0.010, 1.012)	6/115	1.55e-006	3/100	1.14e-015	4/11	4.34e-008
Linear(1.010, 0.013, 0.009, 1.011)	5/109	2.19e-005	1/100	1.91e-005	4/11	4.34e-008
Linear(1.013, 0.008, 0.011, 1.008)	6/111	1.26e-006	0/100	1.00e-000	5/11	2.07e-010
Aspect ratio change(0.80, 1.00)	7/84	8.09e-009	-	-	-	-
Aspect ratio change(0.90, 1.00)	8/93	5.33e-010	-	-	-	-
Aspect ratio change(1.00, 0.80)	1/90	2.37e-001	-	-	-	-
Aspect ratio change(1.00, 0.90)	4/96	2.16e-004	-	-	-	-
Aspect ratio change(1.00, 1.20)	7/121	1.02e-007	-	-	-	-
Aspect ratio change(1.00 1.10)	9/115	1.04e-010	-	-	-	-
Aspect ratio change(1.10, 1.00)	8/127	6.40e-009	-	-	-	-
Aspect ratio change(1.20, 1.00)	6/131	3.31e-006	-	-	-	-
Rotation 1.00	11/113	7.78e-014	-	-	3/11	6.37e-006
Rotation 2.00	6/107	1.02e-006	-	-	1/11	3.68e-002
Rotation 5.00	3/103	3.82e-003	-	-	0/11	1.00e-000
Rotation 10.00	9/99	2.67e-011	-	-	-	-
Rotation 15.00	4/84	1.29e-004	-	-	-	-
Rotation 30.00	4/61	3.69e-005	-	-	-	-
Rotation 45.00	8/359	1.64e-005	1/100	1.91e-005	-	-
Rotation 90.00	5/111	2.39e-005	-	-	-	-
Flipping	1/111	2.84e-001	-	-	-	-
Rotation Scale 1.00	6/113	1.40e-006	-	-	4/11	4.3451e-008
Rotation Scale 10.00	7/122	1.08e-007	-	-	-	-
Rotation Scale 15.00	1/29	8.34e-002	-	-	-	-
Rotation Scale 30.00	8/748	2.19e-003	-	-	-	-
Rotation Scale 45.00	10/740	1.05e-004	-	-	-	-
Rotation Scale 90.00	5/111	2.39e-005	-	-	-	-
Scaling 50%	2/104	3.94e-002	0/100	1.00e-000	-	-
Scaling 75%	6/323	4.89e-004	0/100	1.00e-000	-	-
Scaling 90%	5/77	4.01e-006	2/100	1.82e-010	-	-
Scaling 110%	5/132	5.48e-005	-	-	-	-
Scaling 150%	4/328	1.78e-002	-	-	-	-
Scaling 200%	7/119	9.13e-008	-	-	-	-
Shearing x-0% y-1%	8/111	2.20e-009	-	-	-	-
Shearing x-1% y-0%	9/110	6.96e-011	2/100	1.82e-010	-	-
Shearing x-1% y-1%	5/114	2.72e-005	-	-	4/11	4.34e-008
Shearing x-0% y-5%	10/109	1.92e-012	-	-	3/11	6.37e-006
Shearing x-5% y-0%	6/106	9.62e-007	-	-	-	-
Shearing x-5% y-5%	6/103	8.13e-007	0/100	1.00e-000	0/11	1.00e-000
Random Bending	6/116	1.63e-006	0/100	1.00e-000	-	-

0.0034 is obtained. On the other hand, the false positive probability derived from an image is defined in [33, eq. (6)] as

$$\begin{cases} P_{False-alarm\ on\ one\ image} \\ = \sum_{i=m}^{N} \binom{N}{i} \cdot (P_{False-alarm\ on\ one\ disk})^i \\ \cdot (1 - P_{False-alarm\ on\ one\ disk})^{N-i} \end{cases} \tag{16}$$

where N is total number of disks in an image, and at least m disks are detected as "successful."

As surveyed above, the false positive probabilities of [29], [33] and ours are all calculated based on the Binomial distribution, as shown in (14), (16), and (13), respectively. However, the major difference between them is the probability of determining whether an embedding unit (either a mesh or disk) has been watermarked or not, which is $P_{FA-\text{disk}}$ in Seo and Yoo's method [29], $P_{False-alarm\ on\ one\ image}$ in Tang and Hang's method [33], and p_M in our method. In this study, robustness comparisons among our method, Seo and Yoo's method [29], and Tang and Hang's method [33] were conducted by taking the derived false positive probabilities into consideration. The results will be reported in the next section.

V. EXPERIMENTAL RESULTS

In order to thoroughly verify the robustness of the proposed scheme, the standard benchmark, Stirmark 3.1 [26], [27], and the WEAs [15] were adopted. Three standard images, Baboon, Lena, and Pepper, were used here as cover images, and the size of each one was 512×512. After mesh-based watermark embedding was performed, the PSNR values between the cover image and its stego image for Baboon, Lena, and Pepper were 36.06, 38.44, and 38.32 dB, respectively. No perceptual differences could be observed. Although the PSNR of stego Baboon was smaller than 38 dB, it was still hard to find any quality degradation because the Baboon image was rather noisy. As described previously, two thresholds, $\text{Th}_{\text{mesh}} = 0.375$ and

TABLE VI
GEOMETRIC ATTACKS FOR LENA

attack	proposed method		[29]		[33]	
	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}
1 column, 1 row removed	34/100	7.99e-060	-	-	-	-
5 column, 1 row removed	35/104	2.38e-061	-	-	3/8	2.18e-006
1 column, 5 row removed	29/104	2.59e-048	-	-	-	-
17 column, 5 row removed	24/110	2.33e-037	5/100	1.95e-026	0/8	1.00e-000
5 column, 17 row removed	10/100	8.00e-013	-	-	-	-
Cropping 1% off	27/106	7.14e-044	-	-	-	-
Cropping 2% off	22/103	3.78e-034	-	-	-	-
Cropping 5% off	17/86	4.23e-026	-	-	2/8	3.19e-004
Cropping 10% off	16/77	4.95e-025	-	-	2/8	3.19e-004
Cropping 15% off	15/65	2.58e-024	6/100	5.93e-032	-	-
Cropping 20% off	12/68	3.31e-018	-	-	-	-
Cropping 25% off	12/53	1.27e-019	4/100	5.30e-021	-	-
Cropping 50% off	5/21	4.75e-009	-	-	-	-
Cropping 75% off	2/99	3.60e-002	-	-	-	-
Linear(1.007, 0.010, 0.010, 1.012)	32/104	9.60e-055	6/100	5.93e-032	5/8	2.53e-011
Linear(1.010, 0.013, 0.009, 1.011)	39/104	2.05e-070	7/100	1.52e-037	4/8	9.29e-009
Linear(1.013, 0.008, 0.011, 1.008)	28/100	9.28e-047	7/100	1.52e-037	4/8	9.29e-009
Aspect ratio change(0.80, 1.00)	6/87	2.99e-007	-	-	-	-
Aspect ratio change(0.90, 1.00)	15/94	1.07e-021	-	-	-	-
Aspect ratio change(1.00, 0.80)	3/97	3.23e-003	-	-	-	-
Aspect ratio change(1.00, 0.90)	7/104	3.60e-008	-	-	-	-
Aspect ratio change(1.00, 1.20)	18/121	3.69e-025	-	-	-	-
Aspect ratio change(1.00 1.10)	31/104	1.40e-052	-	-	-	-
Aspect ratio change(1.10, 1.00)	19/122	7.10e-027	-	-	-	-
Aspect ratio change(1.20, 1.00)	13/132	3.69e-016	-	-	-	-
Rotation 1.00	21/109	1.24e-031	-	-	3/8	2.18e-006
Rotation 2.00	21/93	3.15e-033	-	-	0/8	1.00e-000
Rotation 5.00	18/78	6.94e-029	-	-	0/8	1.00e-000
Rotation 10.00	15/77	4.25e-023	-	-	-	-
Rotation 15.00	12/73	8.25e-018	-	-	-	-
Rotation 30.00	9/57	1.56e-013	-	-	-	-
Rotation 45.00	6/38	1.85e-009	2/100	1.82e-010	-	-
Rotation 90.00	23/108	1.34e-035	-	-	-	-
Flipping	19/108	5.95e-028	-	-	-	-
Rotation Scale 1.00	24/105	6.72e-038	-	-	0/8	1.00e-000
Rotation Scale 10.00	7/98	2.38e-008	-	-	-	-
Rotation Scale 15.00	5/89	8.18e-006	-	-	-	-
Rotation Scale 30.00	1/115	2.92e-001	-	-	-	-
Rotation Scale 45.00	0/96	1.00e+000	-	-	-	-
Rotation Scale 90.00	23/108	1.34e-035	-	-	-	-
Scaling 50%	10/110	2.11e-012	2/100	1.82e-010	-	-
Scaling 75%	3/58	7.36e-004	3/100	1.14e-015	-	-
Scaling 90%	4/94	1.99e-004	4/100	5.30e-021	-	-
Scaling 110%	19/120	5.08e-027	-	-	-	-
Scaling 150%	3/57	7.00e-004	-	-	-	-
Scaling 200%	32/102	4.61e-055	-	-	-	-
Shearing x-0% y-1%	23/100	1.88e-036	-	-	-	-
Shearing x-1% y-0%	33/102	2.94e-057	5/100	1.95e-026	-	-
Shearing x-1% y-1%	23/100	1.88e-036	-	-	4/8	9.29e-009
Shearing x-0% y-5%	15/92	7.57e-022	-	-	2/8	3.19e-004
Shearing x-5% y-0%	20/94	3.81e-031	-	-	-	-
Shearing x-5% y-5%	12/78	1.92e-017	1/100	1.91e-005	1/8	2.69e-002
Random Bending	17/110	3.83e-024	4/100	5.30e-021	-	-

$\mathrm{Th}_{\mathrm{image}} = 3.50e-004$, were adopted in our method. In this section, experimental results will be demonstrated with respect to resistance to removal (non-geometric) attacks, resistance to geometric attacks, and resistance to watermark-estimation attacks. The reasons that may lead to the obtained results will also be identified.

In order to demonstrate the superiority of our method, we compared it with other feature-based watermarking methods [2], [29] [33]. In digital watermarking, it has been recognized that robustness is meaningful only if false positives are taken into consideration. Although false positive analyses were conducted in [29], [33] the detection results did not show the impact of this factor, so the reported results are not fully convincing. Therefore, in this study the false positive probability was derived using (14) for the method in [29] and (16) for the method in [33]. To avoid tedious comparisons, the parameters that could produce better results in [29], [33] were adopted here. In [29], the authors declared that when $\mu = 1$ and $P_{FA-\mathrm{image}} = 0.1918e-004$ are used, $P_{FA-\mathrm{disk}} = 0.1918e-006$ is obtained according to (14). The number of disks, μ, detected to contain a watermark and the number of total disks, N, in (14) are denoted in the following tables as D_M and T_M, respectively. $N = 100$ was adopted in [29]. In [33], $n = 16$, $T_1 = 10$, $T_2 = 10$, and $T = 24$ were used, leading to $P_{False-alarm\ on\ one\ disk} = 0.0034$.[2] The number of disks, m, found to contain watermarks and the number of total disks, N, in (16) are denoted in the following tables as D_M and T_M, respectively. "D_M/T_M" in the following tables denotes "the number of detected watermarked meshes(disks)/the number of total meshes(disks)."

[2]This value is very close to P_M of (13) in our method. Thus, the comparisons conducted here are quite fair to avoid any possible parameter selections that may deviate the final false positive probabilities.

TABLE VII
GEOMETRIC ATTACKS FOR PEPPER

attack	proposed method		[29]		[33]	
	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}
1 column, 1 row removed	45/111	6.59e-083	-	-	-	-
5 column, 1 row removed	42/108	1.56e-076	-	-	3/4	1.57e-007
1 column, 5 row removed	39/105	3.25e-070	-	-	-	-
17 column, 5 row removed	34/104	3.96e-059	5/100	1.95e-026	1/4	1.35e-002
5 column, 17 row removed	34/104	3.96e-059	-	-	-	-
Cropping 1% off	34/110	3.81e-058	-	-	-	-
Cropping 2% off	24/110	2.33e-037	-	-	-	-
Cropping 5% off	17/94	2.19e-025	-	-	2/4	6.91e-005
Cropping 10% off	17/88	6.47e-026	-	-	2/4	6.91e-005
Cropping 15% off	14/74	1.84e-021	2/100	1.82e-010	-	-
Cropping 20% off	14/59	5.61e-023	-	-	-	-
Cropping 25% off	6/60	3.18e-008	2/100	1.82e-010	-	-
Cropping 50% off	4/19	3.03e-007	-	-	-	-
Cropping 75% off	4/109	3.51e-004	-	-	-	-
Linear(1.007, 0.010, 0.010, 1.012)	41/111	1.31e-073	5/100	1.95e-026	1/4	1.35e-002
Linear(1.010, 0.013, 0.009, 1.011)	46/108	5.65e-086	7/100	1.52e-037	1/4	1.35e-002
Linear(1.013, 0.008, 0.011, 1.008)	45/110	3.93e-083	5/100	1.95e-026	0/4	1.00e-000
Aspect ratio change(0.80, 1.00)	17/94	2.19e-025	-	-	-	-
Aspect ratio change(0.90, 1.00)	31/97	1.10e-053	-	-	-	-
Aspect ratio change(1.00, 0.80)	9/89	1.01e-011	-	-	-	-
Aspect ratio change(1.00, 0.90)	27/100	1.18e-044	-	-	-	-
Aspect ratio change(1.00, 1.20)	18/128	1.07e-024	-	-	-	-
Aspect ratio change(1.00 1.10)	30/130	4.02e-047	-	-	-	-
Aspect ratio change(1.10, 1.00)	38/110	5.43e-067	-	-	-	-
Aspect ratio change(1.20, 1.00)	20/138	1.51e-027	-	-	-	-
Rotation 1.00	33/106	1.33e-056	-	-	2/4	6.91e-005
Rotation 2.00	20/101	1.85e-030	-	-	1/4	1.35e-002
Rotation 5.00	13/90	2.11e-018	-	-	0/4	1.00e-000
Rotation 10.00	12/74	9.82e-018	-	-	-	-
Rotation 15.00	13/61	9.16e-021	-	-	-	-
Rotation 30.00	12/55	2.07e-019	-	-	-	-
Rotation 45.00	5/46	3.01e-007	1/100	1.91e-005	-	-
Rotation 90.00	25/111	3.10e-039	-	-	-	-
Flipping	25/109	1.87e-039	-	-	-	-
Rotation Scale 1.00	29/106	4.90e-048	-	-	2/4	6.91e-005
Rotation Scale 10.00	7/102	3.15e-008	-	-	-	-
Rotation Scale 15.00	4/84	1.29e-004	-	-	-	-
Rotation Scale 30.00	4/85	1.35e-004	-	-	-	-
Rotation Scale 45.00	3/91	2.69e-003	-	-	-	-
Rotation Scale 90.00	25/111	3.10e-039	-	-	-	-
Scaling 50%	13/101	1.02e-017	2/100	1.82e-010	-	-
Scaling 75%	4/66	5.03e-005	6/100	5.93e-032	-	-
Scaling 90%	22/94	4.03e-035	6/100	5.93e-032	-	-
Scaling 110%	22/136	2.89e-031	-	-	-	-
Scaling 150%	5/65	1.73e-006	-	-	-	-
Scaling 200%	48/105	1.45e-091	-	-	-	-
Shearing x-0% y-1%	43/110	1.94e-078	-	-	-	-
Shearing x-1% y-0%	37/110	9.39e-065	4/100	5.30e-021	-	-
Shearing x-1% y-1%	32/111	1.10e-053	-	-	1/4	1.35e-002
Shearing x-0% y-5%	30/95	8.05e-052	-	-	1/4	1.35e-002
Shearing x-5% y-0%	30/98	2.42e-051	-	-	-	-
Shearing x-5% y-5%	16/94	1.58e-023	0/100	1.00e-000	0/4	1.00e-000
Random Bending	26/109	1.81e-041	3/100	1.14e-015	-	-

The experimental comparisons in terms of resistance to Stirmark attacks between our approach and [29], [33] will be reported in Sections V-A and V-B, respectively.

On the other hand, since Bas *et al.*'s scheme [2] was not evaluated using Stirmark, we implement their approach for the purpose of comparison. We will discuss the robustness comparisons in Section V-C.

Finally, the results about the resistance of our proposed hash-dependent watermarking to estimation attacks will be reported in Section V-D.

A. Resistance to Nongeometric Attacks

The watermark detection results with respect to nongeometric attacks are shown in Tables II–IV for the three standard images, respectively. In Table II, the method in [29] can only survive FMLR and color reduce attacks, while our method and that in [33] can tolerate JPEG compression up to a quality factor of 40%. Furthermore, only our method can survive the Sharpening attack. As shown in Table III, our method can survive almost all attacks except for JPEG10 and FMLR attacks, so it is more robust than the other two. A similar result can also be found in Table IV. On a whole, our method when compared with those in [29], [33] can survive most of the nongeometric attacks of Stirmark 3.1. We also note that it is challenging to extract robust feature points from complex images such as Baboon. Thus, the overall performance with respect to Baboon is not as robust as that for other smoothing images. This phenomenon was observed in [2], [29], and [33] as well as in our study.

B. Resistance to Geometric Attacks

The results of comparisons of resistance to geometric distortions are shown in Tables V–VII. Basically, it can be observed that our method and that in [29] provide p_{fp} that is sufficiently

TABLE VIII
NONGEOMETRIC ATTACKS FOR BABOON. SCHEME 1: PROPOSED FEATURE EXTRACTOR; SCHEME 2: FEATURE EXTRACTOR IN [2]; SCHEME 3: PROPOSED FEATURE EXTRACTOR + SYMMETRIC WATERMARKING PARADIGM

attack	Scheme 1		Scheme 2		Scheme 3	
	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}
Median filter 2x2	20/105	4.31e-030	3/99	3.42e-003	20/101	1.85e-030
Median filter 3x3	22/105	6.03e-034	3/102	3.72e-003	18/109	4.99e-026
Median filter 4x4	12/109	1.27e-015	1/94	2.46e-001	12/105	8.02e-016
Gaussian filter 3x3	28/99	6.70e-047	1/59	1.62e-001	26/101	1.93e-042
JPEG 90	24/101	2.37e-038	18/96	4.24e-027	30/101	7.00e-051
JPEG 80	31/105	1.98e-052	19/101	1.50e-028	34/103	2.67e-059
JPEG 70	38/101	1.02e-068	15/95	1.26e-021	32/103	6.67e-055
JPEG 60	25/99	1.24e-040	7/96	2.06e-008	28/103	2.41e-046
JPEG 50	25/105	6.55e-040	3/101	3.61e-003	24/103	4.01e-038
JPEG 40	15/103	4.57e-021	7/96	2.06e-008	18/103	1.67e-026
JPEG 30	13/101	1.02e-017	7/89	1.22e-008	22/101	2.34e-034
JPEG 20	9/101	3.21e-011	2/95	3.34e-002	17/105	1.65e-024
JPEG 10	1/105	2.71e-001	0/102	1.00e+000	2/103	3.87e-002
FMLR	22/101	2.34e-034	9/96	2.02e-011	31/99	2.32e-053
Color reduce	31/103	9.85e-053	21/100	1.70e-032	32/103	6.67e-055
Sharpening 3x3	14/110	6.71e-019	2/137	6.43e-002	13/112	4.11e-017

TABLE IX
NONGEOMETRIC ATTACKS FOR LENA. SCHEME 1: PROPOSED FEATURE EXTRACTOR; SCHEME 2: FEATURE EXTRACTOR IN [2]; SCHEME 3: PROPOSED FEATURE EXTRACTOR + SYMMETRIC WATERMARKING PARADIGM

attack	Scheme 1		Scheme 2		Scheme 3	
	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}
Median filter 2x2	32/112	1.54e-053	19/112	1.25e-027	28/114	6.14e-045
Median filter 3x3	40/110	1.61e-071	11/117	1.15e-013	36/112	3.38e-062
Median filter 4x4	26/109	1.81e-041	5/107	2.00e-005	30/113	3.45e-049
Gaussian filter 3x3	32/108	3.96e-054	11/88	4.59e-015	31/110	1.06e-051
JPEG 90	53/104	2.59e-104	41/106	1.25e-074	50/108	1.12e-095
JPEG 80	46/108	5.65e-086	26/110	2.37e-041	40/108	6.52e-072
JPEG 70	43/109	1.19e-078	20/101	1.85e-030	36/114	7.22e-062
JPEG 60	33/108	2.75e-056	21/110	1.53e-031	36/109	1.05e-062
JPEG 50	23/107	1.06e-035	9/103	3.83e-011	30/109	9.96e-050
JPEG 40	29/102	1.35e-048	9/105	4.56e-011	19/108	5.95e-028
JPEG 30	24/103	4.01e-038	9/99	2.67e-011	17/107	2.32e-024
JPEG 20	11/105	3.41e-014	5/101	1.51e-005	11/105	3.41e-014
JPEG 10	2/109	4.29e-002	1/125	3.13e-001	2/104	3.94e-002
FMLR	13/94	3.83e-018	12/94	2.02e-016	16/95	1.90e-023
Color reduce	44/108	3.20e-081	42/103	1.27e-077	46/110	1.67e-085
Sharpening 3x3	39/111	4.63e-069	7/145	3.52e-007	37/117	1.44e-063

TABLE X
NONGEOMETRIC ATTACKS FOR PEPPER. SCHEME 1: PROPOSED FEATURE EXTRACTOR; SCHEME 2: FEATURE EXTRACTOR IN [2]; SCHEME 3: PROPOSED FEATURE EXTRACTOR + SYMMETRIC WATERMARKING PARADIGM

attack	Scheme 1		Scheme 2		Scheme 3	
	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}
Median filter 2x2	63/113	3.65e-127	19/103	2.25e-028	56/111	8.69e-110
Median filter 3x3	55/112	5.66e-107	15/115	2.60e-020	57/110	1.23e-112
Median filter 4x4	43/114	1.32e-077	14/122	3.02e-018	38/114	2.79e-066
Gaussian filter 3x3	53/110	1.50e-102	33/99	9.05e-058	56/110	4.32e-110
JPEG 90	75/108	3.39e-162	55/115	4.05e-106	77/108	5.54e-168
JPEG 80	70/110	3.61e-147	47/110	6.84e-088	73/110	1.56e-155
JPEG 70	62/114	3.28e-124	39/105	3.25e-070	65/112	8.30e-133
JPEG 60	56/118	9.12e-108	35/105	3.56e-061	56/116	2.51e-108
JPEG 50	58/108	7.53e-116	30/115	6.30e-049	60/108	4.72e-121
JPEG 40	60/106	9.26e-122	27/116	1.11e-042	55/110	1.46e-107
JPEG 30	36/104	1.37e-063	11/111	6.37e-014	38/102	1.62e-068
JPEG 20	23/112	3.37e-035	8/107	1.64e-009	21/108	1.00e-031
JPEG 10	2/114	4.65e-002	1/124	3.11e-001	3/111	4.71e-003
FMLR	17/97	3.89e-025	20/96	6.06e-031	23/103	4.00e-036
Color reduce	72/110	9.98e-153	54/111	9.26e-105	76/112	4.62e-163
Sharpening 3x3	64/111	1.61e-130	7/143	3.20e-007	65/113	1.95e-132

lower than that in [33] for line removal, cropping attacks, and general linear transformations. Our method also consistently provides much lower p_{fp}'s for shearing and random bending attacks. For other attacks, our method was thoroughly evaluated and found to provide low p_{fp}'s, while [29], [33] did not. This is particularly obvious for resistance to change of the aspect ratio and large degree of shearing (e.g., Shearing $\text{x}-5\%\ \text{y}-5\%$) because the circular disk adopted in [29], [33] could not accommodate such an attack. In addition, for those attacks involving rotations (e.g., shearing, rotation), the method [33] basically generates poor results since the used filtering is rotation-sensitive. In order to better adapt to varied attacks, a rotation-invariant filtering technique together with a triangular mesh-based watermarking scheme are adopted in this study.

C. Comparisons With Bas et al.'s Scheme [2]

Experimental comparisons were conducted based on two scenarios. In the first scenario, the only difference is that the

TABLE XI
GEOMETRIC ATTACKS FOR BABOON. SCHEME 1: PROPOSED FEATURE EXTRACTOR; SCHEME 2: FEATURE EXTRACTOR IN [2]; SCHEME 3: PROPOSED FEATURE EXTRACTOR + SYMMETRIC WATERMARKING PARADIGM

attack	Scheme 1		Scheme 2		Scheme 3	
	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}
1 column, 1 row removed	23/103	4.00e-036	19/102	1.84e-028	29/101	9.69e-049
5 column, 1 row removed	22/98	1.12e-034	16/109	1.97e-022	25/98	9.33e-041
1 column, 5 row removed	24/101	2.37e-038	18/111	7.08e-026	21/102	2.69e-032
17 column, 5 row removed	16/95	1.90e-023	10/115	3.31e-012	16/95	1.90e-023
5 column, 17 row removed	17/95	2.65e-025	11/123	2.01e-013	15/93	9.00e-022
Cropping 1% off	27/108	1.27e-043	20/115	3.09e-029	29/104	2.59e-048
Cropping 2% off	23/106	8.33e-036	15/105	6.20e-021	24/104	5.20e-038
Cropping 5% off	20/101	1.85e-030	13/107	2.22e-017	22/97	8.73e-035
Cropping 10% off	16/85	2.79e-024	13/91	2.46e-018	16/83	1.84e-024
Cropping 15% off	21/75	1.88e-035	5/80	4.84e-006	19/73	1.59e-031
Cropping 20% off	15/71	1.12e-023	10/60	3.88e-015	15/69	6.99e-024
Cropping 25% off	13/59	5.68e-021	6/54	1.66e-008	11/58	3.54e-017
Cropping 50% off	6/19	1.91e-011	0/11	1.00e+000	6/19	1.91e-011
Linear(1.007, 0.010, 0.010, 1.012)	30/103	1.39e-050	9/99	2.67e-011	30/101	7.00e-051
Linear(1.010, 0.013, 0.009, 1.011)	29/105	3.57e-048	12/116	2.74e-015	28/103	2.41e-046
Linear(1.013, 0.008, 0.011, 1.008)	27/105	5.34e-044	7/108	4.68e-008	28/105	4.47e-046
Aspect ratio change(0.80, 1.00)	8/83	2.12e-010	2/73	2.05e-002	11/83	2.35e-015
Aspect ratio change(0.90, 1.00)	8/82	1.92e-010	9/94	1.66e-011	8/82	1.92e-010
Aspect ratio change(1.00, 0.80)	4/80	1.07e-004	7/79	5.25e-009	6/80	1.81e-007
Aspect ratio change(1.00, 0.90)	11/93	8.62e-015	6/85	2.60e-007	16/93	1.32e-023
Aspect ratio change(1.00, 1.20)	12/125	6.85e-015	6/99	6.43e-007	13/127	2.21e-016
Aspect ratio change(1.00 1.10)	17/110	3.83e-024	16/106	1.23e-022	20/108	7.96e-030
Aspect ratio change(1.10, 1.00)	27/114	6.57e-043	2/107	4.15e-002	25/116	1.05e-038
Aspect ratio change(1.20, 1.00)	8/127	6.40e-009	6/99	6.43e-007	13/131	3.33e-016
Rotation 1.00	18/101	1.14e-026	8/105	1.41e-009	23/101	2.42e-036
Rotation 2.00	25/98	9.33e-041	2/116	4.79e-002	25/100	1.65e-040
Rotation 5.00	14/95	7.79e-020	9/100	2.93e-011	15/92	7.57e-022
Rotation 10.00	18/91	1.49e-027	5/81	5.15e-006	15/89	4.45e-022
Rotation 15.00	14/76	2.76e-021	3/69	1.22e-003	10/74	3.56e-014
Rotation 30.00	8/55	7.05e-012	3/46	3.72e-004	7/55	3.91e-010
Rotation 45.00	9/39	3.85e-015	3/38	2.11e-004	7/39	3.09e-011
Rotation 90.00	18/103	1.67e-026	13/96	5.11e-018	17/101	8.14e-025
Flipping	11/105	3.41e-014	7/99	2.56e-008	10/101	8.85e-013
Rotation Scale 1.00	20/103	2.84e-030	6/109	1.13e-006	18/103	1.67e-026
Rotation Scale 10.00	9/119	1.41e-010	0/98	1.00e+000	12/117	3.04e-015
Rotation Scale 15.00	3/130	7.28e-003	0/87	1.00e+000	3/126	6.68e-003
Rotation Scale 30.00	0/140	1.00e+000	0/72	1.00e+000	0/140	1.00e+000
Rotation Scale 45.00	1/139	3.41e-001	0/69	1.00e+000	1/137	3.37e-001
Rotation Scale 90.00	18/103	1.67e-026	13/96	5.11e-018	17/101	8.14e-025
Scaling 50%	0/13	1.00e+000	0/25	1.00e+000	0/13	1.00e+000
Scaling 75%	1/56	1.55e-001	3/64	9.81e-004	1/56	1.55e-001
Scaling 90%	9/78	2.98e-012	11/92	7.62e-015	11/78	1.15e-015
Scaling 110%	13/130	3.01e-016	8/109	1.90e-009	17/128	5.80e-023
Scaling 150%	1/330	6.29e-001	0/97	1.00e+000	2/324	2.54e-001
Scaling 200%	3/703	3.53e-001	1/146	3.55e-001	2/713	6.31e-001
Shearing x-0% y-1%	26/101	1.93e-042	8/89	3.73e-010	23/99	1.45e-036
Shearing x-1% y-0%	22/103	3.78e-034	16/113	3.63e-022	23/101	2.42e-036
Shearing x-1% y-1%	23/103	4.00e-036	6/102	7.67e-007	24/101	2.37e-038
Shearing x-0% y-5%	22/99	1.44e-034	8/102	1.12e-009	21/97	8.40e-033
Shearing x-5% y-0%	20/94	3.81e-031	12/93	1.77e-016	17/94	2.19e-025
Shearing x-5% y-5%	15/91	6.35e-022	5/98	1.31e-005	15/89	4.45e-022
Random Bending	15/105	6.20e-021	1/93	2.44e-001	13/108	2.52e-017

feature point extractors used in our approach and [2] are different, while the watermark size and the mesh size are the same, and the mesh-based watermarking paradigm [2] (called asymmetric embedding/detection process here) is adopted. Let our feature extractor+asymmetric embedding/detection be scheme 1 and Bas *et al.*'s feature extractor + asymmetric embedding/detection be scheme 2. The goal is to demonstrate the geometric resilience of the proposed feature point extractor by comparing schemes 1 and 2. The second scenario is similar to the first scenario. The only difference is that the watermark embedding/detection process is different, i.e., the so-called symmetric watermarking paradigm proposed here and the asymmetric watermarking paradigm used in [2] are compared. Let our feature extractor + symmetric embedding/detection be scheme 3. In both scenarios, the PSNR values of stego images generated from our approach and [2] were controlled to approximately the same. Again, the standard benchmark, Stirmark, was used for robustness evaluation.

The detection results are measured in terms of the proposed false positive probability p_{fp} (13) and are summarized in Table VIII–XIII. In the first scenario, we observe that the p_{fp}'s obtained from scheme 1 are significantly smaller than those from scheme 2 except for very few exceptions, which implies that the proposed geometric-invariant feature extractor is rather helpful in achieving resistance to geometric attacks. In the second scenario, we also observe that the detection results obtained from the two different watermarking paradigms (schemes 2 and 3) are comparable. This means that both the symmetric watermark paradigm (proposed here) and asymmetric watermarking paradigm (used in [2]) equally contribute to robustness. However, as described previously, the proposed symmetric watermarking embedding and detection processes are required to achieve anti-estimation attacks if the hash-dependent watermarks are embedded. Overall, through experimental comparisons we can confirm that our method

TABLE XII
GEOMETRIC ATTACKS FOR LENA. SCHEME 1: PROPOSED FEATURE EXTRACTOR; SCHEME 2: FEATURE EXTRACTOR IN [2]; SCHEME 3: PROPOSED FEATURE EXTRACTOR + SYMMETRIC WATERMARKING PARADIGM

attack	Scheme 1		Scheme 2		Scheme 3	
	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}
1 column, 1 row removed	48/110	2.70e-090	35/104	2.38e-061	48/110	2.70e-090
5 column, 1 row removed	41/108	3.25e-074	32/116	5.64e-053	45/104	1.54e-084
1 column, 5 row removed	51/110	1.34e-097	30/105	2.72e-050	45/110	3.93e-083
17 column, 5 row removed	36/108	7.07e-063	26/111	3.08e-041	38/110	5.43e-067
5 column, 17 row removed	27/100	1.18e-044	26/117	1.42e-040	26/104	4.57e-042
Cropping 1% off	37/109	6.25e-065	33/120	1.58e-054	27/109	1.68e-043
Cropping 2% off	31/112	2.02e-051	35/115	1.57e-059	42/106	5.83e-077
Cropping 5% off	29/88	8.75e-051	33/113	1.58e-055	35/84	2.29e-065
Cropping 10% off	27/75	1.22e-048	32/112	1.54e-053	27/81	1.50e-047
Cropping 15% off	18/66	2.31e-030	27/109	1.68e-043	20/66	1.24e-034
Cropping 20% off	20/74	1.76e-033	30/95	8.05e-052	14/75	2.26e-021
Cropping 25% off	14/55	1.86e-023	31/87	1.89e-055	16/51	2.80e-028
Cropping 50% off	5/24	9.85e-009	14/35	1.05e-026	8/24	4.62e-015
Linear(1.007, 0.010, 0.010, 1.012)	43/104	9.22e-080	23/115	6.57e-035	34/115	2.25e-057
Linear(1.010, 0.013, 0.009, 1.011)	34/105	5.84e-059	21/119	9.14e-031	36/110	1.56e-062
Linear(1.013, 0.008, 0.011, 1.008)	33/101	2.00e-057	25/115	8.29e-039	32/101	3.17e-055
Aspect ratio change(0.80, 1.00)	11/89	5.22e-015	8/101	1.03e-009	16/89	6.18e-024
Aspect ratio change(0.90, 1.00)	28/90	2.96e-048	17/108	2.75e-024	18/94	2.81e-027
Aspect ratio change(1.00, 0.80)	3/91	2.69e-003	15/102	3.92e-021	3/85	2.22e-003
Aspect ratio change(1.00, 0.90)	21/88	8.59e-034	21/101	2.14e-032	24/90	1.03e-039
Aspect ratio change(1.00, 1.20)	17/122	2.46e-023	21/115	4.21e-031	15/128	1.38e-019
Aspect ratio change(1.00 1.10)	38/110	5.43e-067	15/111	1.49e-020	34/108	1.82e-058
Aspect ratio change(1.10, 1.00)	28/122	5.19e-044	23/100	1.88e-036	27/132	5.34e-041
Aspect ratio change(1.20, 1.00)	23/140	8.82e-033	10/101	8.85e-013	20/140	2.05e-027
Rotation 1.00	39/109	1.95e-069	20/127	2.59e-028	40/113	6.00e-071
Rotation 2.00	36/90	2.22e-066	27/123	6.50e-042	34/94	6.20e-061
Rotation 5.00	21/81	1.20e-034	23/118	1.26e-034	22/83	1.82e-036
Rotation 10.00	21/69	2.46e-036	14/114	1.13e-018	21/74	1.36e-035
Rotation 15.00	19/60	2.11e-033	10/87	1.91e-013	16/60	5.69e-027
Rotation 30.00	13/54	1.58e-021	7/69	2.01e-009	10/56	1.85e-015
Rotation 45.00	9/35	1.30e-015	7/58	5.75e-010	8/34	1.11e-013
Rotation 90.00	32/104	9.60e-055	24/107	1.11e-037	28/108	1.10e-045
Flipping	19/108	5.95e-028	14/105	3.40e-019	16/110	2.30e-022
Rotation Scale 1.00	35/113	7.60e-060	29/110	1.68e-047	42/115	4.11e-075
Rotation Scale 10.00	11/94	9.73e-015	11/116	1.04e-013	13/92	2.86e-018
Rotation Scale 15.00	10/89	2.42e-013	12/114	2.21e-015	10/89	2.42e-013
Rotation Scale 30.00	2/91	3.09e-002	5/116	2.96e-005	2/92	3.15e-002
Rotation Scale 45.00	0/93	1.00e+000	1/103	2.66e-001	1/93	2.44e-001
Rotation Scale 90.00	32/104	9.60e-055	24/107	1.11e-037	28/108	1.10e-045
Scaling 50%	0/28	1.00e+000	0/43	1.00e+000	0/28	1.00e+000
Scaling 75%	3/61	8.53e-004	10/73	3.09e-014	3/62	8.95e-004
Scaling 90%	19/92	2.16e-029	14/100	1.66e-019	19/92	2.16e-029
Scaling 110%	23/120	1.92e-034	14/106	3.91e-019	27/120	3.09e-042
Scaling 150%	0/238	1.00e+000	0/87	1.00e+000	0/240	1.00e+000
Scaling 200%	0/445	1.00e+000	0/68	1.00e+000	2/461	4.02e-001
Shearing x-0% y-1%	36/102	5.88e-064	38/112	1.24e-066	35/102	1.05e-061
Shearing x-1% y-0%	34/107	1.25e-058	32/105	1.38e-054	40/110	1.61e-071
Shearing x-1% y-1%	27/104	3.97e-044	20/116	3.73e-029	29/100	6.93e-049
Shearing x-0% y-5%	23/85	2.73e-038	28/112	3.50e-045	26/90	6.13e-044
Shearing x-5% y-0%	30/93	3.77e-052	22/104	4.78e-034	24/97	7.94e-039
Shearing x-5% y-5%	20/79	7.81e-033	17/115	8.53e-024	19/83	2.48e-030
Random Bending	43/111	3.16e-078	26/100	1.44e-042	37/108	4.14e-065

consistently outperforms Bas *et al.*'s method in terms of robustness against nongeometric attacks, geometric attacks, and watermark-estimation attacks.

D. Resistance to Watermark-Estimation Attacks (WEAs)

The collusion attack and copy attack were used to verify the resistance achieved by our method to WEAs [15]. Table XIV and XV show the results of resisting collusion attack for CDW embedding and non-CDW embedding, respectively. After a collusion attack was performed, the number of detected meshes as shown in Table XV was smaller than that shown in Table XIV, which implies that our proposed scheme with CDW embedding efficiently defends against the collusion attack. It should also be noted that mesh-based collusion does not increase the PSNRs of colluded images as block-based collusion does [15]. This may be due to the fact that the interpolation errors involving in mesh warping neutralize the expected PSNR improvement of collusion. Table XVI and XVII show the results of resisting copy attack for CDW embedding and non-CDW embedding, respectively. After a copy attack was performed, the number of detected meshes as shown in Table XVII was larger than that shown in Table XVI, which implies our proposed scheme with CDW embedding efficiently defends against the copy attack. However, the content-independent watermarking methods [2], [29], [33] cannot survive WEAs [15].

To summarize, extensive experiment results verify that our method indeed outperforms all the other feature-based watermarking methods.

E. Discussions

In this section, we shall discuss the impact of each step in our method on the detection results and identify which step mostly affects the overall performance. As described previously

TABLE XIII
GEOMETRIC ATTACKS FOR PEPPER. SCHEME 1: PROPOSED FEATURE EXTRACTOR; SCHEME 2: FEATURE EXTRACTOR IN [2]; SCHEME 3: PROPOSED FEATURE EXTRACTOR + SYMMETRIC WATERMARKING PARADIGM

attack	Scheme 1		Scheme 2		Scheme 3	
	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}
1 column, 1 row removed	66/112	1.78e-135	52/111	8.56e-100	70/110	3.61e-147
5 column, 1 row removed	70/108	4.73e-148	41/110	8.27e-074	74/108	2.49e-159
1 column, 5 row removed	59/108	1.92e-118	48/118	2.05e-088	62/110	1.35e-125
17 column, 5 row removed	49/106	9.41e-094	34/116	3.17e-057	52/104	8.77e-102
5 column, 17 row removed	54/105	1.51e-106	35/116	2.24e-059	54/105	1.51e-106
Cropping 1% off	55/111	2.89e-107	42/118	1.55e-074	54/111	9.26e-105
Cropping 2% off	46/112	4.81e-085	42/120	3.66e-074	42/112	1.05e-075
Cropping 5% off	37/98	5.08e-067	35/113	7.60e-060	34/98	3.48e-060
Cropping 10% off	30/100	4.93e-051	20/97	7.62e-031	31/98	1.60e-053
Cropping 15% off	27/77	2.89e-048	20/89	1.13e-031	25/75	3.86e-044
Cropping 20% off	24/63	3.95e-044	17/82	1.74e-026	22/63	1.47e-039
Cropping 25% off	20/60	1.30e-035	13/70	6.46e-020	20/60	1.30e-035
Cropping 50% off	9/21	5.60e-018	6/27	2.04e-010	10/21	2.02e-020
Linear(1.007, 0.010, 0.010, 1.012)	71/110	6.12e-150	35/117	3.18e-059	71/112	4.62e-149
Linear(1.010, 0.013, 0.009, 1.011)	63/113	3.65e-127	38/123	8.68e-065	69/111	5.40e-144
Linear(1.013, 0.008, 0.011, 1.008)	70/108	4.73e-148	34/112	7.84e-058	65/110	1.45e-133
Aspect ratio change(0.80, 1.00)	14/87	2.11e-020	13/99	7.76e-018	20/87	6.84e-032
Aspect ratio change(0.90, 1.00)	40/96	1.76e-074	32/109	5.60e-054	40/102	3.80e-073
Aspect ratio change(1.00, 0.80)	14/88	2.50e-020	20/102	2.29e-030	15/88	3.71e-022
Aspect ratio change(1.00, 0.90)	36/99	1.58e-064	37/105	1.17e-065	37/99	8.09e-067
Aspect ratio change(1.00, 1.20)	28/128	2.32e-043	20/112	1.75e-029	26/130	2.93e-039
Aspect ratio change(1.00 1.10)	41/118	2.81e-072	29/110	1.68e-047	39/120	1.81e-067
Aspect ratio change(1.10, 1.00)	47/119	7.57e-086	35/114	1.09e-059	43/117	5.25e-077
Aspect ratio change(1.20, 1.00)	30/135	1.42e-046	23/114	5.27e-035	28/127	1.82e-043
Rotation 1.00	46/114	1.35e-084	32/111	1.10e-053	46/112	4.81e-085
Rotation 2.00	40/102	3.80e-073	30/108	7.24e-050	38/104	4.03e-068
Rotation 5.00	35/88	1.80e-064	21/98	1.07e-032	36/88	7.97e-067
Rotation 10.00	24/80	3.90e-041	15/97	1.76e-021	24/80	3.90e-041
Rotation 15.00	22/67	7.48e-039	7/81	6.26e-009	22/63	1.47e-039
Rotation 30.00	15/61	8.89e-025	2/62	1.51e-002	13/61	9.16e-021
Rotation 45.00	13/54	1.58e-021	1/55	1.52e-001	12/54	1.62e-019
Rotation 90.00	39/112	7.09e-069	28/117	1.39e-044	40/110	1.61e-071
Flipping	23/112	3.37e-035	13/109	2.85e-017	23/110	2.14e-035
Rotation Scale 1.00	42/117	1.00e-074	39/108	1.26e-069	43/119	1.29e-076
Rotation Scale 10.00	13/91	2.46e-018	9/101	3.21e-011	13/93	3.31e-018
Rotation Scale 15.00	12/84	4.92e-017	6/89	3.42e-007	10/86	1.70e-013
Rotation Scale 30.00	5/95	1.12e-005	5/78	4.28e-006	5/95	1.12e-005
Rotation Scale 45.00	6/101	7.24e-007	4/68	5.66e-005	5/95	1.12e-005
Rotation Scale 90.00	39/112	7.09e-069	28/117	1.39e-044	40/110	1.61e-071
Scaling 50%	0/37	1.00e+000	0/39	1.00e+000	0/37	1.00e+000
Scaling 75%	7/71	2.46e-009	13/80	4.17e-019	3/69	1.22e-003
Scaling 90%	32/89	2.43e-057	22/104	4.78e-034	28/89	2.04e-048
Scaling 110%	37/128	7.15e-062	25/106	8.55e-040	36/126	4.95e-060
Scaling 150%	2/209	1.31e-001	12/85	5.71e-017	3/209	2.56e-002
Scaling 200%	1/348	6.49e-001	0/82	1.00e+000	2/340	2.72e-001
Shearing x-0% y-1%	61/109	2.53e-123	50/117	2.11e-093	64/109	2.86e-131
Shearing x-1% y-0%	57/112	5.12e-112	54/116	2.26e-103	54/112	1.78e-104
Shearing x-1% y-1%	52/111	8.56e-100	40/117	3.25e-070	54/110	4.77e-105
Shearing x-0% y-5%	57/101	1.21e-115	36/117	2.18e-061	54/99	1.72e-108
Shearing x-5% y-0%	48/101	1.21e-092	37/108	4.14e-065	50/105	1.71e-096
Shearing x-5% y-5%	32/96	4.57e-056	27/107	9.52e-044	31/94	3.43e-054
Random Bending	39/111	4.63e-069	30/108	7.24e-050	40/115	1.41e-070

TABLE XIV
COLLUSION ATTACK ON CDW EMBEDDING

image	CDW stego image		PSNR	colluded image		PSNR
	D_M/T_M	p_{fp}	(dB)	D_M/T_M	p_{fp}	(dB)
Baboon	7/111	5.65e-008	36.06	5/107	2.00e-005	33.07
Lena	31/108	5.48e-052	38.44	17/97	3.89e-025	35.35
Pepper	57/109	5.95e-113	38.32	29/109	1.24e-047	35.21

TABLE XV
COLLUSION ATTACK ON NON-CDW EMBEDDING

image	Non-CDW stego image		PSNR	colluded image		PSNR
	D_M/T_M	p_{fp}	(dB)	D_M/T_M	p_{fp}	(dB)
Baboon	20/103	2.84e-030	36.06	0/99	1.00e+000	34.64
Lena	56/105	1.13e-111	38.43	7/111	5.65e-008	38.17
Pepper	65/119	2.57e-130	38.31	12/109	1.27e-015	38.42

TABLE XVI
COPY ATTACK ON CDW EMBEDDING

image	CDW stego image		PSNR	copy attacked image		PSNR
	D_M/T_M	p_{fp}	(dB)	D_M/T_M	p_{fp}	(dB)
Baboon	7/111	5.65e-008	36.06	3/105	4.03e-003	36.02
Lena	31/108	5.48e-052	38.44	1/103	2.66e-001	38.39
Pepper	57/109	5.95e-113	38.32	1/115	2.92e-001	38.27

TABLE XVII
COPY ATTACK ON NON-CDW EMBEDDING

image	Non-CDW stego image		PSNR	copy attacked image		PSNR
	D_M/T_M	p_{fp}	(dB)	D_M/T_M	p_{fp}	(dB)
Baboon	20/103	2.84e-030	36.06	24/105	6.72e-038	36.02
Lena	56/105	1.13e-111	38.43	53/99	6.71e-106	38.38
Pepper	65/119	2.57e-130	38.31	59/105	1.75e-119	38.27

in Section III, in addition to media hashing, feature point extraction and denoising-based blind detection are recognized as two main factors that may affect the performance of our method. Since the robustness of our media hashing has been verified in [19], it is not discussed here again. According to the experimental results shown in the above tables, it is important to

TABLE XVIII
IMPACT OF FEATURE POINT EXTRACTION AND DENOISING-BASED BLIND DETECTION ON THE PERFORMANCE OUR WATERMARKING METHOD (AD: AVERAGE DISPLACEMENT OF FEATURE POINTS IN PIXELS)

image	Condition (i)		Condition (ii)		AD
	D_M/T_M	p_{fp}	D_M/T_M	p_{fp}	
Baboon	67/103	6.08e-142	7/113	6.40e-008	4.13
Lena	88/100	9.83e-208	32/106	1.97e-054	2.59
Pepper	95/107	5.07e-225	55/109	7.33e-108	1.60

know how many meshes of a stego image, under the absence of attacks, can be detected to contain watermarks. Two experiments were performed based on the conditions that (i) the feature points and media hashes extracted from the original image are directly applied to the stego image, which means that feature point extraction is perfect and we are only interested in understanding the effect of Wiener filtering and (ii) all the processes are the same as those described in Section III, which means that by comparing the results obtained from conditions (i) and (ii), we can understand the effect of feature point extraction (and media hashing). The results of these two experiments are depicted in Table XVIII.

As we can see from Table XVIII that when condition (i) is considered, denoising-based blind detection slightly affects the detection results. For example, the number, T_M, of total meshes in Baboon is 103 and the number of meshes, D_M, detected to contain watermarks is 67. The similar results can also be found in Lena and Pepper. However, when condition (ii) is considered, D_M for each stego image, when compared with the results obtained in condition (i), is dramatically reduced. This obviously implies that the correctness of extracted points plays a major role in the performance of our watermarking method. More specifically, it can be observed from Table XVIII that the average displacements (in pixels) of feature points illustrate the obtained detection results. As a consequence, we can conclude that the stability of feature point extraction mainly affects the overall performance of our watermarking method. This conclusion is also consistent with the robustness verifications described in the above subsections that resistance to attacked Baboon images is apparently inferior to resistance to other smoother images.

VI. CONCLUSIONS

Although multiple watermarks can be embedded into an image to provide resistance to geometric distortions, we found in our companion study [15] that they are, unfortunately, vulnerable to watermark estimation attacks (including collusion and copy attacks) such that the desired geometric invariance is lost. In view of this fact, a mesh-based content-dependent image watermarking method that can resist extensive geometric attacks and watermark estimation attacks simultaneously has been proposed here. There are three major contributions of our method. First, robust mesh extraction is designed to enhance the feasibility of feature-based watermarking methods. Second, a media hash-based content-dependent watermark that is composed of a watermark and a hash is used to resist watermarking-estimation attack. Third, a false positive-oriented watermark detection mechanism is applied to determine the existence of a watermark so as to achieve a tradeoff between correct detection and false detection. The performance of our scheme in enhancing robustness has been thoroughly verified using the standard benchmark, Stirmark, and watermark estimation attacks.

However, the major weakness of our method is its high complexity since most of the time is spent on mesh warping, which makes the method in its current state unsuitable for real-time applications. By keeping the achievable robustness, reducing the complexity of our method deserves further researching. In addition, as described in Section V-E, enhancing the stability of feature point extraction can further improve the overall performance of the proposed method. Finally, the important issue of security against protocol attacks based on the proposed method was also investigated. Due to limits of space, the results were reported elsewhere [18].

REFERENCES

[1] M. Alghoniemy and A. H. Tewfik, "Geometric invariance in image watermarking," *IEEE Trans. Image Process.*, vol. 13, no. 2, pp. 145–153, Feb. 2004.

[2] P. Bas, J. M. Chassery, and B. Macq, "Geometrically invariant watermarking using feature points," *IEEE Trans. Image Proces.*, vol. 11, no. 9, pp. 1014–1028, Sep. 2002.

[3] S. Craver, N. Memon, B. L Yeo, and M. M. Yeung, "Resolving rightful ownerships with invisible watermarking techniques: limitations, attacks, and implications," *IEEE J. Select. Areas Commun.*, vol. 16, no. 4, pp. 573–586, Apr. 1998.

[4] R. Durstenfeld, "Algorithm 235: random permutation [G6]," *Commun. ACM*, p. 420, 1964.

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

[6] F. Hartung and B. Girod, "Watermarking of uncompressed and compressed video," *Signal Process.*, vol. 66, no. 3, pp. 283–302, 1998.

[7] A. J. Hayter, *Probability and Statistics for Engineers and Scientists*. Boston, MA: PWS Publishing Company, 1995.

[8] J. R. Hernandez and F. Perez-Gonzalez, "Statistical analysis of watermarking schemes for copyright protection of images," *Proc. IEEE*, vol. 87, no. 7, pp. 1142–1143, Jul. 1999.

[9] A. Herrigel, S. Voloshynovskiy, and Y. Rytsar, "The watermark template attack," in *Proc. SPIE Security and Watermarking of Multimedia Contents III*, San Jose, CA, Jan. 2001, vol. 4314.

[10] H. S. Kim and H. K. Lee, "Invariant image watermark using zernike moments," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 8, pp. 766–775, Aug. 2003.

[11] D. Knuth, *The Art of Computer Programming*, 3rd ed. Reading, MA: Addison-Wesley, 1997, vol. 2.

[12] M. Kutter, "Watermarking resisting to translation, rotation and scaling," in *Proc. SPIE Int. Symp. Voice, Video, and Data Communication*, Boston, MA, Nov. 1998.

[13] M. Kutter, S. K. Bhattacharjee, and T. Ebrahimi, "Toward second generation watermarking schemes," in *Proc. IEEE Int. Conf. Image Processing*, 1999, vol. I, pp. 320–323.

[14] 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 Process.*, vol. 10, no. 5, pp. 767–782, May 2001.

[15] C. S. Lu and C. Y. Hsu, "Content-dependent anti-disclosure image watermark," in *Proc. 2nd Int. Workshop on Digital Watermarking*, Seoul, Korea, 2003, vol. 2939, LNCS, pp. 61–76.

[16] C. S. Lu, S. W. Sun, and P. C. Chang, "Robust mesh-based content-dependent image watermarking with resistance to both geometric attack and watermark-estimation attack," in *Proc. SPIE: Security, Steganography, and Watermarking of Multimedia Contents VII (EI120)*, San Jose, CA, 2005, pp. 147–163.

[17] C. S. Lu, "Towards robust image watermarking: combining content-dependent watermark, moment normalization, and side-informed embedding," *Signal Process.: Image Commun.*, vol. 20, no. 2, pp. 129–150, 2005.

[18] C. S. Lu and C. M. Yu, "On the security of mesh-based media hash-dependent watermarking against protocol attacks," in *Proc. IEEE Int. Conf. Multimedia and Expo*, The Netherlands, 2005.

[19] C.-S. Lu and C.-Y. Hsu, "Geometric distortion-resilient image hashing scheme and its applications on copy detection and authentication,," *ACM Multimedia Syst. J., Special Issue on Multimedia and Security,*, vol. 11, no. 2, pp. 159–173, 2005.
[20] A. R. Manuel and P. G. Fernando, "Analysis of pilot-based synchronization algorithms for watermarking of still images," *Signal Process.: Image Commun.*, vol. 17, pp. 611–633, 2002.
[21] K. Mikolajczyk and C. Schmid, "An affine invariant interest point detector," in *Proc. ECCV*, 2002, vol. 2350, LNCS, pp. 128–142.
[22] A. Nikolaidis and I. Pitas, "Region-based image watermarking," *IEEE Trans. Image Process.*, vol. 10, no. 11, pp. 1726–1740, 2001.
[23] J. O'Ruanaidh and T. Pun, "Rotation, scale and translation invariant spread spectrum digital image watermarking," *Signal Process.*, vol. 66, no. 3, pp. 303–317, May 1998.
[24] S. Pereira and T. Pun, "Robust template matching for affine resistant image watermarks," *IEEE Trans. Image Process.*, vol. 9, no. 6, pp. 1123–1129, Jun. 2000.
[25] ——, "An iterative template matching algorithm using the chirp-Z transform for digital image watermarking," *Pattern Recognit.*, vol. 33, pp. 173–175, 2000.
[26] F. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Attacks on copyright marking systems," in *Proc. Int. Workshop on Information Hiding*, 1998, vol. 1575, LNCS, pp. 219–239.
[27] F. Petitcolas, "Watermarking schemes evaluation," *IEEE Signal Process. Mag.*, vol. 17, no. 5, pp. 58–64, 2000.
[28] M. Ramkumar and A. N. Akansu, "A robust scheme for oblivious detection of watermarks/data hiding in still images," in *Proc. SPIE Multimedia Systems and Applications*, 1998, vol. 3528, pp. 474–481.
[29] J. S. Seo and C. D. Yoo, "Localized image watermarking based on feature points of scale-space representation," *Pattern Recognit.*, vol. 37, pp. 1365–1375, 2004.
[30] D. Simitopoulos, D. E. Koutsonanos, and M. G. Strintzis, "Robust image watermarking based on generalized radon transformations," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 8, pp. 732–745, Aug. 2003.
[31] V. Solachidis and I. Pitas, "Circularly symmetric watermark embedding in 2-D DFT domain," *IEEE Trans. Image Process.*, vol. 10, no. 11, pp. 1741–1753, Nov. 2001.
[32] S. Stankovic, I. Djurovic, and I. Pitas, "Watermarking in the space/spatial-frequency domain using two-dimensional Radon-Wigner distribution," *IEEE Trans. Image Process.*, vol. 10, no. 4, pp. 650–658, Apr. 2001.
[33] C. W. Tang and H. M. Hang, "A feature-based robust digital image watermarking scheme," *IEEE Trans. Signal Process.*, vol. 51, no. 4, pp. 950–958, Apr. 2003.
[34] S. Voloshynovskiy, A. Herrigel, N. Baumgartner, and T. Pun, "A stochastic approach to content adaptive digital image watermarking," in *Proc. Int. Workshop on Information Hiding*, 1999, vol. 1768, LNCS, pp. 211–236.
[35] S. Voloshynovskiy, F. Deguillaume, and T. Pun, "Multibit digital watermarking robust against local nonlinear geometrical distortions," in *Proc. IEEE Int. Conf. Image Processing*, Thessaloniki, Oct. 2001, pp. 999–1002.
[36] S. Voloshynovskiy, S. Pereira, V. Iquise, and T. Pun, "Attack modeling: towards a second generation watermarking benchmark," *Signal Process.*, vol. 81, pp. 1177–1214, 2001.
[37] D. Zheng, J. Zhao, and A. El Saddik, "RST-invariant digital image watermarking based on log-polar mapping and phase correlation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 8, pp. 753–765, Aug. 2003.
[38] P. Zhu and P. M. Chirlian, "On critical point detection of digital shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 8, pp. 737–748, Aug. 1995.

Chun-Shien Lu (M'99) received the Ph.D. degree in electrical engineering from National Cheng-Kung University, Tainan, Taiwan, R.O.C., in 1998.

Since August 2002, he has been an Assistant Research Fellow at the Institute of Information Science, Academia Sinica, Taipei, Taiwan. His current research interests focus on multimedia and networking. He has two U.S. patents, two R.O.C. patents, and one Canadian patent in digital watermarking. He is the editor of *Multimedia Security: Steganography and Digital Watermarking Techniques for Protection of Intellectual Property* (Idea Group, January 2004).

Dr. Lu organized a special session on Multimedia Security in the Second and Third IEEE Pacific-Rim Conference on Multimedia, respectively (2001–2002). He co-organized two special sessions (in the area of media identification and DRM) in the Fifth IEEE International Conference on Multimedia and Expo (ICME), 2004. He was a Guest Co-Editor of the *EURASIP Journal on Applied Signal Processing* (Special Issue on Visual Sensor Networks) in 2005. He has received numerous paper awards from the Image Processing and Pattern Recognition Society of Taiwan for his work on data hiding. He was a co-recipient of a National Invention and Creation Award in 2004. He is a member of the ACM.

Shih-Wei Sun received the B.S. degree in electrical engineering from Yaun Ze University, Chung-Li, Taiwan, R.O.C., in 2001. He is currently pursuing the Ph.D. degree at the Video/Audio Processing Laboratory in Electrical Engineering, National Central University, Chung-Li.

His research interests include image and video signal processing, multimedia security, multimedia communications, and digital watermarking.

Chao-Yong Hsu is currently pursuing the Ph.D. degree in the Graduate Institute of Communication Engineering, National Taiwan University, Taipei.

He is a Research Assistant in the Institute of Information Science, Academia Sinica, Taipei, His research interests focus on multimedia.

Pao-Chi Chang received the B.S. and M.S. degrees from National Chiao-Tung University, Taiwan, R.O.C., in 1977 and 1979, respectively, and the Ph.D. degree from Stanford University, Stanford, CA, in 1986, all in electrical engineering.

From 1986 to 1993, he was a Research Staff Member in the Department of Communications, IBM T. J. Watson Research Center, Hawthorne, NY, where his work centered on high-speed switching systems, efficient network design algorithms, and multimedia conferencing. In 1993, he joined the faculty of National Central University, Taiwan, where he is presently a Professor in the Department of Communication Engineering. In 1994, he established and has headed the Video-Audio Processing Laboratory (VAPLab) in the Electrical Engineering Department and Communication Department, National Central University. He is the Principle Investigator for many joint projects with the National Science Council (NSC), Institute of Information Industry (III), Chung Hwa Telecommunication Laboratories (TL), and many other companies. His research interests include speech/audio coding, video/image compression, scalable coding, error-resilient coding, digital watermarking and data hiding, and multimedia delivery over packet and wireless networks. He has published more than 70 journal and conference papers in these areas.