A Geometrically Resilient Digital Image Watermarking Scheme Based on SIFT and Extended Template Embedding

Po-Chyi Su
Dept. of Computer Science and Information Engineering
National Central University, Jhongli, Taiwan
Email: pochyisu@csie.ncu.edu.tw

Yu-Chuan Chang
Dept. of Computer Science and Information Engineering
National Central University, Jhongli, Taiwan
Email: 995202026@cc.ncu.edu.tw

***Abstract*—This research presents a feature-based still image watermarking approach. Scale-Invariant Feature Transform (SIFT) is first applied to locate the interest points, from which we form the invariant regions for watermark embedding. To resist geometrical transformations, the extended synchronization templates, which help to ensure that reasonably large invariant regions will be available for carrying the watermark payload and/or for increasing the confidence of watermark detection, will also be embedded. In the detection phase, after SIFT, the template is first determined locally by adjusting the related affine parameters of the grid to match with the possible hidden template signal so that the watermark can be retrieved afterwards. Experimental results show the feasibility of the proposed method.**

***Keywords*—digital watermark; geometrical transformations; SIFT; StirMark.**

I. INTRODUCTION

Digital watermarking has been considered as a potential solution to providing further protection of digital content. The close integration of the hidden signal, *i.e.*, digital watermark, with the host media can be used for declaring/verifying the ownership of the content, controlling the software/hardware operations or for the trailer tracking purposes. In most of the related applications, the digital watermark signal has to be robust against the "watermark attacks," including lossy compression, signal processing procedures and even malicious watermark-removal operations, etc. For still images, the watermark surviving geometrical transformations is always required since such manipulations as cropping, rotation and scaling are so common. Nevertheless, these procedures cause challenging synchronization problems for watermark detection and special care must be taken such that the watermark can resist such attacks to meet the requirements of applications.

Existing methods to resist geometrical transformations can be classified into four types, *i.e.*, the exhaustive search [1], embedding the watermark in invariant domains [2], [3], embedding synchronization templates [4], [5], [6] and employing feature detections for locating the watermark [7]. In our opinions, exhaustive search has to be coupled with certain side information to reduce its computational load. The algorithms of watermarking in invariant domains seem elegant but they may not perform well under all kinds of possible geometrical attacks. Employing synchronization templates may be a more flexible method to deal with attacks. However, the so-called "template attack" [8] may detect and remove the template if it is used repeatedly. The feature-based approaches thus gain more and more attention. Kutter *et al.* [7] first claimed that the feature-based approaches are the second-generation watermarking schemes. They illustrated this concept by applying the Mexican-hat wavelet to extract features and the Voronoi diagram to define the local characteristic regions for watermarking. Several methods have been proposed in recent years [9], [10], [11], [12], [13]. The basic idea of these approaches is applying such feature-point extraction as Harris corner detection [14] or Scale-Invariant Feature Transform (SIFT) [15], etc. to determine the interest areas, which are then transformed into the regions with known shape, size and orientation for the subsequent watermark embedding and detection. Since the interest points are extracted according to the content, the process of locating the embedded areas can be facilitated. Nevertheless, according to our observation, the feature-based watermarking may encountered some problems. First, in order to detect one watermark in a small invariant area, such transforms as Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT) or Discrete Fourier Transform (DFT) are usually applied and the middle-frequency components may be modified considerably to achieve the robustness and/or payload. Compared with the neighboring areas without watermarks, the quality of embedded area may be affected a lot, especially when the perceptual model is not employed. Second, deviations of the position, scale and/or orientation of the watermarked regions may appear under attacks, or even right after the watermark embedding without any attack. Therefore, almost all the schemes have to try a few different shapes of their invariant regions for the watermark detection. Then, the watermark here looks more like a template or pilot signal, instead of a watermark signal carrying the necessary hidden information. The false positive rate of watermark detection may also become higher. Furthermore, if image transforms are used, they have to be applied several times and the computational load will be increased. Third, there are

 ISBN: 978-1-61208-204-2

usually many feature points extracted from an image. We thus have to make sure that the watermark embedder and detect choose the same ones for processing and this is not a trivial issue. In our opinions, certain searching should be applied to make the feature-based approaches more practical.

In this research, we propose a feature-based still image watermarking approach based on SIFT and the extended template embedding to alleviate the above-mentioned problems. The spatial template signal will be embedded for achieving the synchronization. The template detection is based on the local search of the hidden templates to recover the regions for the subsequent watermark detection. The rest of the paper is organized as follows. The proposed scheme is detailed in Section II, including the reasons of such design, the determination of the invariant areas, the signal embedding and detection processes. Section III shows some experimental results and Section IV presents the conclusion and future work.

II. The Proposed Watermarking Approach

Fig. 1 illustrates the flowchart of the proposed scheme. To begin with, the image is applied with SIFT to extract the feature points, which help to determine the invariant regions by their descriptors, including location, scale, and orientation. Some inappropriate feature points are removed in the preprocessing step. The invariant regions for signal embedding will be formed around the remaining points. Basically, we segment the image into areas associated with different feature points and each invariant region extends to a large area for the signal embedding. We choose to embed the watermark that contains the necessary hidden information in DCT coefficients as an illustration while the template signal will be embedded in the spatial/pixel domain. Both signals will be weighted according to the perceptual model for guaranteeing the image quality. In other words, the template will serve as the pilot signal for synchronization. In the signal detection, SIFT is applied and the local searching will be performed around the extracted feature points to determine the possible areas with watermark. The preprocessing step may be omitted if the detection efficiency is not the major issue. Through the template detection, we can roughly locate the target areas and then perform the watermark detection. This design may look a bit strange to many people since the feature-point extraction should have solved the synchronization problem and the template or pilot signals seem unnecessary. However, the watermark detection usually requires pretty strict synchronization for signal matching. As mentioned before, the feature detection will be affected by geometrical attacks more or less and the accuracy may not be enough. The use of template will help to not only ensure the synchronization but speed up the detection process. Therefore, the major contribution of our research is to combine the different types of watermarking approaches in a reasonable way to achieve the feasibility. In other words, local searching will be applied to accommodate the possible deviations of extracted features. The searching is based on template matching but different (known) templates can be used to avoid the "template attacks." As the template or pilot signals may not carry enough information for the target applications, we only use them for synchronization and the watermark carrying the information can thus be embedded and detected successfully. Furthermore, compared with the existing methods, our algorithm demonstrates better performances against the random bending attack in StirMark [16], which applies different affine transformations on different areas. Next, we will detail each step in the following.

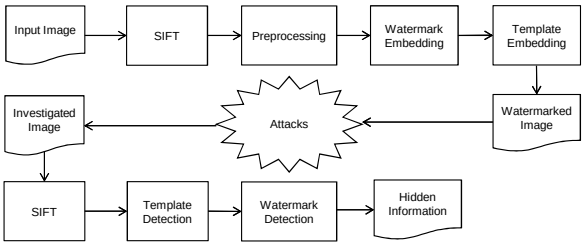


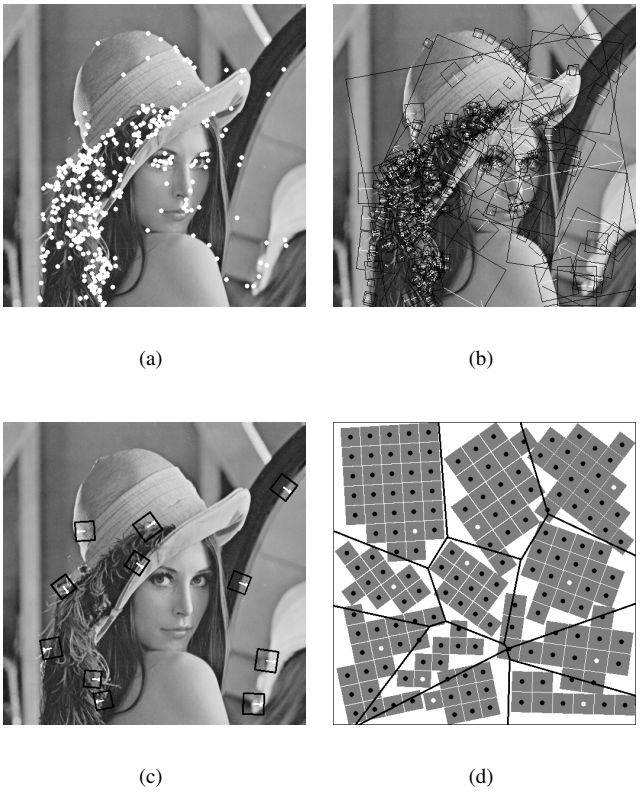
Fig. 1. The flowchart of the proposed watermarking approach.

A. Invariant Area Determination

The first step is to determine the areas for signal embedding/detection so that the synchronized detection can be achieved. Here, we use the Lena image to explain the procedures. Fig. 2(a) illustrates the interest or feature points extracted by SIFT and marked with white dots. Fig. 2(b) shows the invariant areas associated with all the feature points in Lena. The invariant regions are squares centered at the interest points. The square shape is chosen since the watermark will be embedded after a block transform. Besides, we will try to embed the signal in larger areas and these squares can help to cover a broader portion of an image by tiling. The length of one side of this square is determined by the multiplication of the characteristic scale of the corresponding SIFT feature point, λ, and a predefined positive value, τ. The orientation of the invariant area is also decided by the gradient information of the SIFT interest point. Since SIFT usually generates a large number of interest points and certain invariant regions are even overlapped, we choose to reject some interest points from signal embedding. First, we check the robustness of interest points and delete weaker ones. We apply JPEG compression with quality factor equal to 30, followed by Gaussian filtering, and pick those interest points that are still matched between the original image and attacked image. Some points are further eliminated according to their characteristic scales. It should be noted that the extracted invariant area will be embedded with the signal with a fixed size so the scaling/normalization of either our hidden signal pattern or the image content is inevitable. If the size of invariant area is too different from the size of the hidden pattern, the scaling itself may affect the embedded signal severely. For example, if the fixed size is 32×32, we will pick the feature point with its $\lambda \times \tau$ within the range of $[28, 36]$. Furthermore, we expect that the selected points should be separated from each other by a reasonable distance so we adopt the maximum distance algorithm to

 ISBN: 978-1-61208-204-2

disperse the feature points. To be more specific, the location of interest points can be seen as a set of 2-dimensional vectors, $\mathcal{V} = \{\mathbf{v_i}\}$. We calculate the centroid of the $|\mathcal{V}|$ vectors and choose the one closest to the centroid as our first feature point, $\mathbf{fp_1}$. From the $|\mathcal{V}| - 1$ vectors, choose the one that has the largest distance from the first feature point as the second feature point, $\mathbf{fp_2}$. To find the third feature point from the remaining $|\mathcal{V}| - 2$ vectors, we calculate the distance $Dist_i$ of each vector, $\mathbf{v_i}$, and the already chosen feature points, (*i.e.*, $Dist_i = Min(Dist(\mathbf{fp_1}, \mathbf{v_i}), Dist(\mathbf{fp_2}, \mathbf{v_i}))$). Again, we will find the one with the largest distance as $\mathbf{fp_3}$. Repeat the process to increase the number of feature points until the additional feature point will yield the distance from the selected points smaller than a distance threshold, T_{Dist}. These feature points will then be used to form the grids for the signal embedding. Fig. 2(c) illustrates the chosen invariant regions. Some existing schemes may only employ these regions for the watermark embedding/detection. However, if the regions are small, the payload or the detection confidence will be affected. On the other hand, if a larger size is used, the embedded watermark may be affected by the local distortion easily. In our scheme, the invariant areas or grids will be extended to cover a larger area for signal embedding. The major advantages of expanding are enhancing the detection confidence, even though weak signals are embedded, and the increased robustness against the random bending by StirMark. Before expanding, we use Voronoi diagram to set up the boundaries for separating the image into subregions and each subregion belongs to one selected feature point. The expansion of invariant area can then be applied. For an invariant grid, four extended grids are generated. The grids associated with the same feature point will thus have the same size and orientation. The same process will be applied on each extended grid too and the expansion from one initial grid will be limited in the Voronoi subregion. A few grids can still be added on the boundaries of Voronoi subregions as long as the added ones will not overlap others. The grids for signal embedding can then be generated almost all over the host image as illustrated in Fig. 2(d).



(a) (b) (c) (d)

Fig. 2. (a) The extracted feature points by SIFT on Lena and (b) the associated invariant areas. (c) The chosen invariant regions and (d) the extended grids.

B. Signal Embedding

It should be noted that the signals will be embedded into almost all the grids but the grids that cover the initially selected feature points as shown in Fig. 2(c). The reason of such omission is to avoid the embedded signals from modifying the descriptors of interest points and from making these points undetectable. According to our observation, the signal embedding will change the image data more or less and we cannot fully prevent the feature-point extraction from being affected. Although we may apply SIFT on the embedded area right after the embedding to see whether the selected feature point is reliable or not, we choose not to take this risk so that the embedding process can be simplified. In other words, the signals will only be embedded into the extended grids. Therefore, each extended grid will be embedded with the watermark first and then with the template signal. For the watermark embedding, we employ a pretty traditional spread spectrum approach in DCT as an example. A pseudo-random sequence W taking two values, *i.e.*, ± 1, is generated as the watermark signal and embedded into the middle-low frequency DCT coefficients. To be more specific, we randomize Hadamard sequences to generate the watermark sequences since they are mutually orthogonal. This bipolar watermark signal is then weighted according to Watson's perceptual model [17]. Although Watson's model should be able to be applied on blocks with variable sizes, some questioned that it can only ensure the invisibility of the noises within blocks. We thus choose small blocks for the watermark embedding. A grid with its side length equal to $\lambda \times \tau$ will be rotated and scaled into a 32×32 block, which will be divided into 8×8 subblocks for the watermark embedding. According to the zig-zag scan, the lowest three DCT coefficients are skipped and the next three coefficients are chosen as an illustration for watermark embedding/detection. The lowest frequency components are excluded to maintain the high image quality while the high-frequency components are not chosen to reduce the interference from the template signals. It should be noted that other methods, such as quantization index modulation, may also work under our framework since the synchronization issue will be settled. The Watson's model basically takes two masking effects into account, *i.e.*, the luminance masking

 ISBN: 978-1-61208-204-2

and contrast masking. The luminance masking refers to the dependency of the visual threshold and the mean luminance of the local image region while the contrast masking indicates that the threshold for a visual pattern would be reduced in the presence of other patterns. The Just Noticeable Difference (*JND*) of a DCT coefficient, $m_{i,j,h}$, is computed as

$$m_{i,j,h} = Max[a_{i,j,h}, |c_{i,j,h}|^{s_{i,j}} \times a_{i,j,h}^{(1-s_{i,j})}], \quad (1)$$

where $a_{i,j,h}$ is the luminance-adjusted threshold related to the global display and perceptual parameters, such as the viewing distance, display resolution and luminance. $c_{i,j,h}$ is the DCT coefficient and $s_{i,j}$ is the exponent that typically is set as 0.7.

As mentioned before, for a given grid, $\mathbf{G}$, we will transfer it to a 32×32 square $\hat{\mathbf{G}}$ and the watermark signals will be embedded into its 8×8 DCT coefficients $c_{i,j,h}$ by

$$c'_{i,j,h} = \begin{cases} c_{i,j,h} + w_{i,j,h} \cdot m_{i,j,h}, \\ \quad \text{if } sgn(c'_{i,j,h}) = sgn(c_{i,j,h}) \\ 0, \quad \text{otherwise,} \end{cases} \quad (2)$$

where $c'_{i,j,h}$ is the watermarked coefficient and the watermarked grid, $\mathbf{G}'$, is formed by the inverse DCT. We calculate the difference grid $\bar{\mathbf{G}} = \mathbf{G}' - \hat{\mathbf{G}}$. For each pixel in the grid, $\mathbf{G}$, we determine its deviation by checking $\bar{\mathbf{G}}$ with the inverse mapping. Since the watermark sequence is long, we will embed it into several grids. A selected feature point will serve as an anchor point and the watermark sequence will be embedded into the corresponding locations.

Then, each pixel in an extended grid will be embedded with the component of a 32×32 template taking ± 1. Since the template signal can be viewed as a spatial-domain watermark, to maintain the quality of the image, we employ the noise visibility function (*NVF*) [18] to determine the embedded energy. For each pixel, $I(i,j)$, its *NVF* is derived from:

$$NVF(I(i,j)) = \frac{1}{1 + \sigma_I^2(i,j)}, \quad (3)$$

where $\sigma_I^2(i,j)$ denotes the local variance in a window centered on the pixel. The template embedding is applied by

$$I^t(i,j) = I^r(i,j) + (1 - NVF(I(i,j))) \cdot \alpha_s \cdot W_t(i,j), \quad (4)$$

where $I^r(i,j)$ denotes the DCT watermarked pixel and $\alpha_s = 3$ is a predefined embedding strength. $W_t(i,j)$ is the template component and $I^t(i,j)$ is the resulting pixel.

C. Signal Detection

For the signal detection, an investigated image will be applied with SIFT to extract the feature points to locate the areas for the hidden signal detection. The correlation between the retrieved signal and the host template/watermark will be computed to verify whether the hidden signal exists. If the template is found, the watermark will be extracted and the expanding procedure will be performed to find other areas for detections. Since the grid covering the initial feature point is not embedded with any signal, we adopt a strategy of "delayed detection." Given a selected interest point, we use its characteristic orientation and scale to help extract the four adjacent grids for template detection. For each extended grid, we slightly adjust the parameters to form various affine matrices and grid centers. A set of compensative grids, $\hat{\mathbf{G}}_\mathbf{u}$, are generated for tests. The positions do not need to be integers so that the accuracy can be further achieved. Then we apply the interpolation to form a 32×32 block for the template detection. Since the detection in the extended grids around the feature point is very important, we test 3×3 positions (± 1 pixels in horizontal and vertical directions), $\pm 6^\circ, \pm 4^\circ, \pm 2^\circ$ rotations, and $\pm 6, \pm 4, \pm 2$ pixel differences of the grid side. Totally 441 detections will be applied for each grid. The detection is based on the correlation coefficient of the template $\mathbf{W_t}$ and the filtered grid by

$$\rho_u = \frac{\mathbf{W_t} \cdot \tilde{\mathbf{G}}_\mathbf{u}}{\sqrt{\mathbf{W_t} \cdot \mathbf{W_t}} \sqrt{\tilde{\mathbf{G}}_\mathbf{u} \cdot \tilde{\mathbf{G}}_\mathbf{u}}}, \quad (5)$$

where $\tilde{\mathbf{G}}_\mathbf{u}$ is obtained by filtering $\hat{\mathbf{G}}_\mathbf{u}$ with

$$\mathbf{H} = \begin{bmatrix} 0 & 1 & -2 & 1 & 0 \\ 1 & 2 & -6 & 2 & 1 \\ -2 & -6 & 16 & -6 & -2 \\ 1 & 2 & -6 & 2 & 1 \\ 0 & 1 & -2 & 1 & 0 \end{bmatrix}. \quad (6)$$

The largest response, ρ_u^{max}, in an extended grid is compared with a threshold, T_1, to determine if the template exists. The sphere model [19] is used to evaluate the false positive rate. For a single detection, the false positive rate is estimated by

$$P_{fp}^{single}(\rho) = \frac{\int_0^{\cos(\rho)} \sin^{N-2}(x) dx}{2 \int_0^{\pi/2} \sin^{N-2}(x) dx}, \quad (7)$$

where ρ is the response of a single detection and $N = 32 \times 32$. The false positive rate of a grid is

$$P_{fp}^{grid}(\rho) = 1 - (1 - P_{fp}^{single}(\rho))^K, \quad (8)$$

where $K = 411$ is the number of detections in a grid. We can thus set T_1 according to the target false alarm rate.

Once the first extended grid associated with a selected feature point is marked as being embedded with the template pattern, we can proceed to expand the invariant area by the similar method. Again, the delayed detection will be applied. That is, we employ/adjust the determined affine matrix and grid center that result in the largest response in the already detected grid for testing the current extended grid. The slight difference is that less trials are used in the extended grids to speed up the process. To be more specific, we check 3×3 positions and ± 2 degrees so totally $K = 27$ compensated grids will be tested. If the largest response among the K trials is not large enough, the expanding procedure from this grid will be stopped. The unique idea in our approach is that we do not use a single fixed threshold for determining the existence of a template. Since we adopt the strategy of extended grids, after we find a large response, it is quite possible that the adjacent grids will be embedded with the signal. We thus adopt a lower threshold T_2 for the further extended grids and this trick is quite helpful in correctly finding more grids for the

 ISBN: 978-1-61208-204-2

subsequent watermark detection. However, a lower threshold may introduce a higher false alarm rate. Our strategy is to set up another adaptive threshold, T_3. From our observations, the responses in grids of a subregion are usually related as they are similarly large or small, T_3 is designed as $\rho_m - 2 \times \rho_\sigma$, where ρ_m is the mean of detected template responses in a subregion so far, and ρ_σ is the standard deviation. The rule of template detection is thus as follows. If ρ_u^{max} of a grid is larger than T_1, the template is ruled as being detected. If ρ_u^{max} is smaller than T_1 but larger than $\max\{T_2, T_3\}$, the responses of its neighbors will be checked. If there is at least one neighbor with its ρ_u^{max} larger than T_1 or there are at least two neighbors with their ρ_u^{max} larger than T_2, the template will also be ruled as being detected. For the grids around the selected feature point, $T_1 = 0.1425$ and $T_2 = 0.1089$. For other extended grids, $T_1 = 0.1233$ and $T_2 = 0.0937$. The corresponding false alarm rates of T_1 and T_2 are 0.001 and 0.033 respectively. Basically, the expansion is applied in a recursive way but we always check whether a response of a grid has been computed before to speed up the process.

After the template detection helps to achieve the synchronization, the detection of watermark can be executed in a straightforward manner. The 32×32 affine-transformed grid will be divided into sixteen 8×8 subblocks for calculating DCT. The same coefficients will be considered for the watermark detection. Similarly, the response, ρ_b, is calculated by

$$\rho_b = \frac{\sum_h \sum_{(i,j)\in B} c^*_{i,j,h} \times w^b_{i,j,h}}{\sqrt{\sum_h \sum_{(i,j)\in B} (c^*_{i,j,h})^2} \sqrt{\sum_h \sum_{(i,j)\in B} (w^b_{i,j,h})^2}}, \quad (9)$$

where $c^*_{i,j,h}$ is the DCT coefficient and B is the set of selected DCT coefficients and $w^b_{i,j,h} = \pm 1$ is the component of tested watermark sequence. After all the grids are detected, the existence of watermark is claimed if ρ_b is larger than a threshold, T_{nc}, which is also set according to a pre-determined false positive rate by Eq. (7) with N equal to the number of considered DCT coefficients. Large ρ_b indicates the existence of a certain watermark signal. To embed more information, we may simply divide the DCT coefficients into m parts and each part is embedded with one of 2^n watermark sequences so that $m \times n$ bits are embedded.

III. Experimental Results

We demonstrate some results to show the feasibility of the proposed approach by using 512×512 Lena image. The size of a template pattern is set to be 32×32 as mentioned before. The marked image is shown in Fig. 3 with Peak Signal to Noise Ratio (PSNR) equal to 36.43 dB. Then we test several kinds of attacks on the proposed watermarking scheme. We use StirMark benchmark 4.0 to generate the attacked images, including rotating $1°$, $2°$, $5°$, $10°$, $15°$, $30°$, $45°$, $90°$, scaling to 50%, 60%, 70%, 75%, 80%, 90%, 110%, 120% of the size, cropping off 15%, 25%, 50%, 75% of the size, horizontally/vertically shearing by 5%, JPEG with quality factor equal to 90, 70, 50, Gaussian filtering, sharpening, and rotating with cropping by $15°$ and $45°$. Table I shows the results of the attacked Lena images. The second column shows the number of detected SIFT interest points. The two values shown in the third column are the numbers of correctly determined interest points for watermarking and those should be detected. The fourth column lists the numbers of detected grids, followed by the numbers of false positive detections. The fifth column shows the responses of watermark detections. The sixth column shows the threshold T_{nc}, which corresponds to the false alarm rate equal to 10^{-8}. The execution time evaluated in seconds is listed in the seventh column as the reference. The first row shows the results of marked image without any attack for comparison. All the embedded grids can be correctly determined. As we can see, the watermarks are detected in almost all the cases except the attack of scaling by 50% and cropping by 75%. Cropping by a large scale may result in fewer feature points left and the number of DCT coefficients may not be large enough to generate a higher response than the adaptive threshold.

Compared with the existing works, such as [9] and [10], our method demonstrates more consistent performances under various attacks. We can compare the numbers of detected regions, as shown in the third column of Table I, with those listed in the tables of [10]. The values in their five methods vary in different cases. If we use the ratio, *i.e.*, the number of detected areas divided by the number of embedded areas, as an indication, the ratios of five approaches in [10] are $71/168 = 0.42$, $20/132 = 0.15$, $84/276 = 0.30$, $53/324 = 0.16$ and $52/156 = 0.33$. The ratio of [9] is $73/468 = 0.16$ and ours is $68/132 = 0.52$, which is the highest. In addition, our approach has reasonable performances in all the attack cases except the aspect-ratio changes, from which no template can be determined. This problem may be solved by employing more flexible templates, instead of squares only. It is worth noting that our scheme may outperform others under the random bending attack by StirMark. Since this attack is applied in a random way, each time a different outcome appears. Table II illustrates the performances of our scheme against such attack. We employ two parameters, weaker (1.0) and stronger (2.0) geometrical modifications. The tests are run six times in each case. We can see that our scheme has no problem dealing with the random bending on the Lena image. In fact, according to our experiments, strong attacks sometimes cause misses of detections in other images. For stronger random bending attacks, two challenges may appear. The first challenge is the stability of SIFT interest points. It seems that either the descriptors or positions of interest points are changed. The same problem may happen when sharpening is applied. We think that developing a more suitable feature extraction method for digital watermarking may be an interesting research topic. The second challenge is that using only square grids for matching may not be enough, as mentioned before. Further adjusting the parameters of grids, such as modifying the positions of four corners in different ways, may provide more diverse forms of grids for detection. However, heavier computational load will be expected and may hinder the feasibility.

 ISBN: 978-1-61208-204-2

Fig. 3. The watermarked Lena with PSNR equal to 36.43 dB.

TABLE I
RESULTS OF ATTACKED LENA IMAGES

Attacks	Pts.	Regions	Grids	Response	T_{nc}	Time
NoAttack	77	11/11	161(0)	0.36	0.06	116
Rot. 1	97	10/11	143(0)	0.34	0.07	151
Rot. 2	82	10/11	124(2)	0.32	0.07	126
Rot. 5	90	10/11	143(1)	0.33	0.07	142
Rot. 10	94	8/11	112(1)	0.31	0.08	166
Rot. 15	79	9/11	124(0)	0.32	0.07	126
Rot. 30	95	10/11	119(1)	0.32	0.07	169
Rot. 45	91	7/11	85(0)	0.34	0.09	181
Rot. 90	81	11/11	161(0)	0.36	0.06	116
Scale 0.5	137	3/11	38(0)	0.06	0.13	252
Scale 0.6	167	7/11	32(1)	0.25	0.14	301
Scale 0.7	114	6/11	49(0)	0.31	0.12	206
Scale 0.75	104	8/11	111(2)	0.26	0.08	187
Scale 0.8	94	11/11	146(1)	0.32	0.07	156
Scale 0.9	70	8/11	95(0)	0.33	0.08	125
Scale 1.1	102	9/11	98(1)	0.32	0.08	175
Scale 1.2	136	8/11	107(1)	0.30	0.08	234
Crop 15%	73	8/ 8	92(0)	0.36	0.08	108
Crop 25%	60	7/ 7	65(0)	0.35	0.10	78
Crop 50%	29	3/ 3	19(0)	0.36	0.18	32
Crop 75%	6	1/ 1	4(0)	0.33	0.39	5
ShearX 5%	82	10/11	151(2)	0.30	0.07	116
ShearY 5%	92	9/11	131(1)	0.32	0.07	141
JPEG 90	80	11/11	161(0)	0.35	0.06	118
JPEG 70	88	10/11	131(1)	0.33	0.07	141
JPEG 50	95	7/11	49(0)	0.31	0.12	179
Gaussian	93	8/11	95(0)	0.35	0.08	137
Sharpen	75	4/11	57(0)	0.44	0.11	126
Rot.Crop15	76	8/8	81(0)	0.33	0.09	99
Rot.Crop45	61	5/5	54(1)	0.31	0.11	101

TABLE II
RANDOM GEOMETRICAL ATTACKS FROM STIRMARK

Attacks	Pts.	Regions	Grids	Response	T_{nc}	Time
Lena/1.0/(1)	98	10/11	124(2)	0.32	0.07	147
Lena/1.0/(2)	102	7/11	93(0)	0.33	0.11	159
Lena/1.0/(3)	97	10/11	138(0)	0.30	0.08	151
Lena/1.0/(4)	100	10/11	150(0)	0.37	0.11	142
Lena/1.0/(5)	97	8/11	109(1)	0.29	0.07	138
Lena/1.0/(6)	90	11/11	134(0)	0.31	0.11	137
Lena/2.0/(1)	107	8/11	52(0)	0.36	0.07	183
Lena/2.0/(2)	97	8/11	55(1)	0.29	0.07	166
Lena/2.0/(3)	108	8/11	59(1)	0.28	0.08	192
Lena/2.0/(4)	99	10/11	132(3)	0.28	0.09	147
Lena/2.0/(5)	111	8/11	85(0)	0.34	0.07	181
Lena/2.0/(6)	101	8/11	66(0)	0.21	0.10	170

IV. CONCLUSION

We developed a feature based image watermarking method enabling the spread spectrum based schemes to resist geometric distortions. SIFT is used to help solve the synchronization problem. The embedding regions are extended to increase the detection confidence. The experimental results show that the scheme is effective against rotation, scaling, cropping, shearing, and random bending. The current version mainly illustrates the feasibility and novel ideas of combining SIFT and extended template embedding. We may integrate this idea with the parallel computing to speed up the processing.

REFERENCES

[1] M. Barni, "Effectiveness of exhaustive search and template matching against watermark desynchrnizaion," *IEEE Signal Processing Letters*, vol. 12, no. 2, pp. 158–161, 2005.

[2] J. Ruanaidh and T. Pun, "Rotation, scale and translation invariant spread spectrum digital image watermarking," *Signal processing*, vol. 66, no. 3, pp. 303–317, 1998.

[3] 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. on Image Processing*, vol. 10, no. 5, pp. 767–782, 2001.

[4] S. Pereira and T. Pun, "Robust template matching for affine resistant image watermarks," *IEEE Trans. on Image Processing*, vol. 9, no. 6, pp. 1123–1129, 2000.

[5] M. Kutter, "Watermarking resisting to translation, rotation and scaling," in *Proc. of SPIE: Multimedia systems and applications*, Boston, MA, Nov. 1998, pp. 423–431.

[6] P.-C. Su and C.-C. J. Kuo, "Synchronized detection of the block-based watermark with invisible grid embedding," in *SPIE Photonics West*, San Jose, CA, Jan. 2001, pp. 423–431.

[7] M. Kutter, S. Bhattacharjee, and T. Ebrahimi, "Towards second generation watermarking schemes," in *IEEE International Confernece on Image Process*, vol. 1, 1999, pp. 320–323.

[8] A. Herrigel, S. Voloshynovsky, and Y. Rytsar, "The watermark template attack," in *SPIE Photonics West*, San Jose, CA, Jan. 2008, pp. 394–405.

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

[10] J. Seo and C. Yoo, "Image watermarking based on invariant regions of scale-space representation," *IEEE Trans. on Signal Processing*, vol. 54, no. 4, pp. 1537–1549, 2006.

[11] X. Gao, C. Deng, X. Li, and D. Tao, "Geometric Distortion Insensitive Image Watermarking in Affine Covariant Regions," *IEEE Trans. on Systems, Man and Cybernetics. Part C, Applications and reviews*, vol. 40, no. 3, pp. 278–286, 2010.

[12] X. Wang, J. Wu, and P. Niu, "A new digital image watermarking algorithm resilient to desynchronization attacks," *IEEE Trans. on Information Forensics and Security*, vol. 2, no. 4, pp. 655–663, 2007.

[13] D. Zheng, S. Wang, and J. Zhao, "RST invariant image watermarking algorithm with mathematical modeling and analysis of the watermarking processes," *IEEE Trans. on Image Processing*, vol. 18, no. 5, pp. 1055–1068, 2009.

[14] C. Harris and M. Stephens, "A combined corner and edge detector," in *Alvey vision conference*, vol. 15. Manchester, UK, 1988, p. 50.

[15] D. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[16] F. Petitcolas, M. Steinebach, F. Raynal, J. Dittmann, C. Fontaine, and N. Fates, "A public automated web-based evaluation service for watermarking schemes: StirMark benchmark," in *Proc. SPIE*, 2001, pp. 575–584.

[17] A. Watson, "Visually optimal DCT quantization matrices for individual images," in *Data Compression Conference*, 1993, pp. 178–187.

[18] S. Voloshynovskiy, A. Herrigel, N. Baumgaertner, and T. Pun, "A stochastic approach to content adaptive digital image watermarking," in *Information Hiding*. Springer, 1999, pp. 211–236.

[19] M. Miller and J. Bloom, "Computing the probability of false watermark detection," in *The Third International Workshop on Information Hiding*, Dresden, Germany, Sep. 1999, pp. 146–158.

 ISBN: 978-1-61208-204-2