# Geometric-invariant image watermarking by object-oriented embedding

*Yu-Tzu Lin, Ja-Ling Wu, and Yu-Feng Kuo*

National Taiwan University, Taipei, Taiwan

**Summary**

Traditional digital watermarking techniques often treat the image as a set of pixels rather than a set of objects. Under these schemes, the resistance to geometrical attacks is a difficult problem because the image pixels are sensitive to geometrical distortions. Once the watermarked image is altered by geometrical operations, pixels are disturbed and the recovery is difficult especially when the blind watermarking is a must.

In this paper, we propose a simple watermarking scheme which is blind-detectable, computational-efficient, and robust against geometrical attacks and signal processing, in addition, the embedding capacity is reasonable. The proposed object-oriented watermarking scheme is based on the assumption that even the malicious attackers would not destroy the user-attentive objects (the so-called principal objects) in the images too much, so embedding watermark in these objects are more robust to geometrical alternations or signal processing than in the whole image. We extract segments contained in the principal object, and embed watermarks in one of the extracted segments. Experiments show promising results. This fact encourages us to do further researches on object-oriented watermarking which analyzes the host images from the viewpoints of human vision

***Key words:***

*Watermarking, geometric attacks, affine-resistance, image normalization, blind watermarking.*

## Introduction

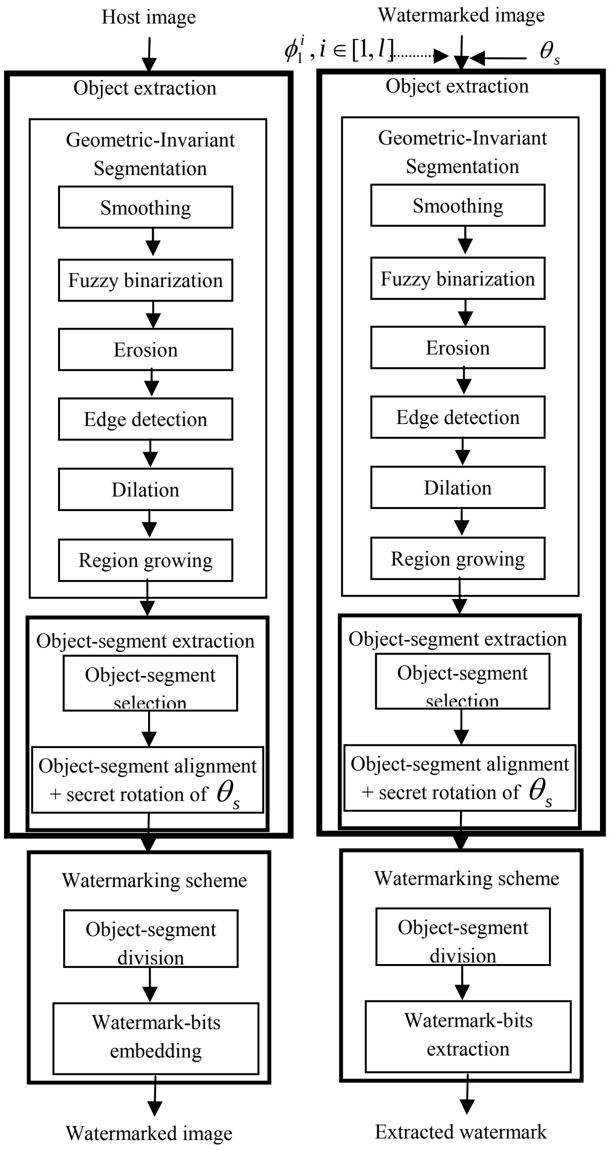
Many digital watermarking schemes have been proposed for many different purposes, such as copy protection, ownership verification, and fingerprinting. But most of them are sensitive to geometrical alternations (e.g., affine transformations and cropping) because traditional watermarking techniques often treat the image as a set of pixels rather than a set of objects. Once the watermarked image is attacked by geometrical distortions, pixels are disturbed and the recovery is difficult especially when the blind watermarking is a must. Some existing affine-resistant watermarking algorithms [1,2,3] embedded watermarks in the Fourier-Mellin transform domain so as to take advantages of some useful properties of the transform, such as rotation-invariance and scaling-invariance. However, the corresponding large amount of computation required for applying the transform to the whole image handicap the method for real-time applications. [2] added a template in the Fourier transform domain to render the method robust against general linear transformations. In [3], the watermark is embedded into a 1-D signal obtained by taking the Fourier transform of the image, and resampling the Fourier magnitudes into log-polar coordinates. Some watermarking schemes [4] disturbed the watermark sequence in order to prevent cropping attacks. Another strategy [5] tries to embed a mark besides the main watermark for identifying the distortions, but this will impair the image fidelity and it is not workable when the mark is distorted by attacks. Feature based watermarking schemes [6,7] extracted geometrically invariant features from the image for watermark embedding and detection. [6] described the content by salient points and bound the mark with the content descriptor. The mark is embedded using a classical additive scheme inside each triangle of the tessellation formed by salient points. [7] extracted feature points and normalized the disks derived from feature points for watermark embedding. Another trend of affine-resistant watermarking is based on the image normalization: [8] reconstructed images with respect to the angle orientation and the flipping condition by comparing the central moments. [9] used 11 moment-invariants to provide solutions for designing the geometric-invariant authentication systems. [10,11] designed geometrically robust watermarking systems based on Zernike moments. There are still many other approaches [12,13] addressing this problem but affine-resistance is still a challenging topic in the watermarking research.

In this work, we try to achieve the affine-resistance by proposing an object-oriented watermarking scheme based on the assumption that even if the malicious attackers would not destroy the user-attentive objects in the images, so embedding watermarks in these objects are more robust against various geometrical alternations than in the whole image. The segments contained in the principal object must be extracted first, and the watermark is then embedded in one of the extracted segments.

In the rest of this paper, we will first briefly describe our system in Section 2. In Section 3, we present a Geometric-

Manuscript received March 25, 2006.
Manuscript revised March 30 , 2006.

Invariant Segmentation algorithm and detail the extraction process of the user-attentive object. Section 4 states the embedding and extraction approach of the proposed spatial-domain quantization watermarking and moment-based watermarking scheme. Section 5 presents our experiment results and Section 6 gives the concluding remarks.



(a) (b)

Fig. 1 System overview: (a) watermark embedding procedure and (b) watermark extraction procedure.

## 2. System Overview

Fig. 1 shows our system block diagram. Fig. 1 (a) illustrates the watermark embedding procedure. The host image is first segmented into several areas from which we select the user-attentive one (called object-segment) for embedding the watermarks. In order to obtain an object-segment invariant to distortions which are resulted from various attacks, a reliable segmentation algorithm is a must. After conducting a stable segmentation, the object-segment which we select for watermarking should be aligned by the eigenvector corresponding to the largest eigenvalue of the pixels in the object. The aligned object is then divided into sections for embedding bits of the watermark. Two embedding techniques: spatial-domain quantization watermarking [14] and moment-based watermarking are applied to the watermarking procedure. Pixels inside the object-segment, derived in the previous step, are modified according to the pre-defined quantization step or the needed values of the moment-invariant.

The watermark extraction process is illustrated in Fig. 1 (b), in which the first two steps are the same as those in the watermark embedding procedure. In the third step, the watermark sequence is simply extracted by calculating the ratio of 1s in each of the object-divisions if spatial-domain embedding is used, and is decided by comparing values of the moment-invariant if the moment-based one is used.

## 3. Object Extraction

In the proposed object-oriented watermarking scheme, an object-segment (the segment contained in the user-attentive objects) should be extracted from the image before watermark embedding. The watermark is then embedded in the extracted segment. It is clear that if the extraction algorithm is sensitive to alternations on images, large errors will be generated in the watermark extraction. So, the stability of the object-segment extraction is important.

### 3.1 Geometric-Invariant Segmentation

In the object-extraction process, the image has to be firstly segmented into several segments for locating candidates of the user-attentive object-segment. As mentioned above, in order to resist various attacks, the segmentation algorithm must be insensitive to slight distortion of the image so that the object-extraction can be stable. Unfortunately, the segmentation results of gray-level images are often sensitive to the changes of the image pixels. In this section, we propose a reliable segmentation technique, called "Geometric-Invariant Segmentation" (GIS), which is invariant to unintentional or malicious attacks of certain degrees.

Image pixels are firstly smoothed and binarized to reduce the noise possibly introduced in the edge detection step of the proposed segmentation algorithm. Instead of binarizing the image by a hard decision method, we propose a binarization approach using fuzzy membership functions, which binarizes the image by classifying image pixels into two classes: one is the class of pixels close to minima graylevel of the image ($G_{min}$), and another is close to maxima graylevel ($G_{max}$). Two membership functions $m_{min}(x)$ and $m_{max}(x)$ evaluate the certainty of re-classifying graylevel x to the classes of $G_{min}$ and $G_{max}$, respectively, and are obtained by computing $Ra(x)$ and $R_b(x)$. As shown in Fig. 2, $R_a(x)$ represents the accumulative histogram from $G_{min}$ to graylevel $x$ while $R_b(x)$ represents the accumulative histogram from graylevel $x$ to $G_{max}$. In fact, Fuzzy Binarization histogram equalizes the graylevles and then binarizes them based on the obtained new graylevel-values. The following pseudo-codes present the proposed Fuzzy Binarization algorithm.

```
Fuzzy_Binarization(image)
{
   for i=1 to image.width
      for j=1 to image.height {
         x= image.graylevel[i][j];
         $m_{min}(x)= R_b(x)/( R_a(x)+ R_b(x))$;
         $m_{max}(x)= R_a(x)/( R_a(x)+ R_b(x))$;
         image.graylevel[i][j]=
                        $G_{min} * m_{min}(x)+ G_{max} * m_{max}(x)$;

      }
   for i=1 to image.width
      for j=1 to image.height {
         if (image.graylevel[i][j]>=( $G_{min}$ + $G_{max}$)/2)
            image.graylevel[i][j]=black;
         else
            image.graylevel[i][j]=white;
      }
}
```

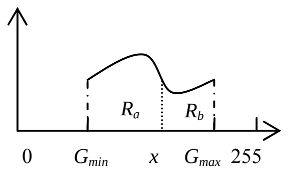


Fig. 2 $R_a$ and $R_b$ used in Fuzzy Binarization.

After the binarization step, the morphological erosion and a noise-removal operation are performed on the binarized image for removing black and white noises. Then we detect edges in the image and the morphological dilation is employed to connect small gaps so that more stable results are obtained. Finally, region growing is applied to non-edge areas to mark all the segments in the image. Segments with too small sizes (less than a predefined threshold) will be skipped when they are isolated and will be merged if they are neighboring to other segments. Fig. 3 illustrates the results of the segmentation process. It can be shown from the experiments that the segmentation results are invariant to various attacks of certain degrees. To measure the invariance of the segmentation results, $Sim_{xy}$ is defined as in (1) and evaluates the similarity between the segmented image of the original image $x$ and the segmented image of the attacked image $y$.

$$Sim_{xy} = 1 - \frac{Dis(F(x), F(y))}{|F(x)|}, \tag{1}$$

in which $Dis(a,b)$ is the Euclidean distance between the feature vector a and b, and $F(\cdot)$ is the first 64 Discrete Fourier Transform (DFT) magnitudes of the segmented image (obtained by setting the edge pixels of the segmented image to white ones and the other pixels to black). Fig. 4 illustrates the evaluation of the segmentation performance.

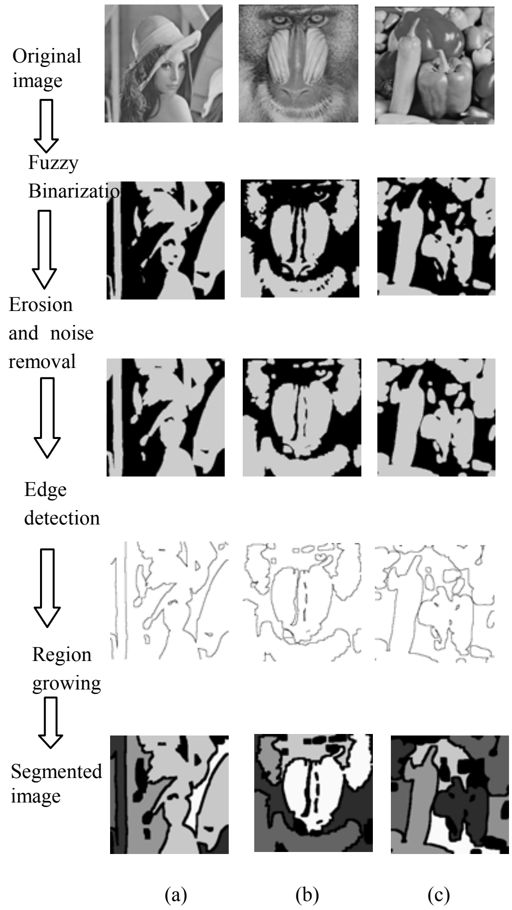


Fig. 3 Examples for the pre-scribed segmentation procedures: (a) Lena, (b) Baboon, and (c) Peppers.

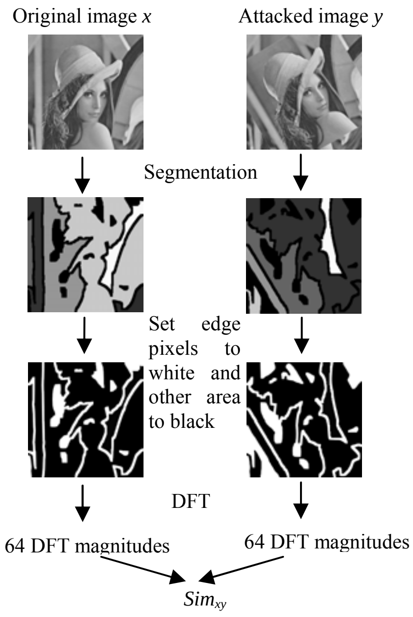


Fig. 4 The evaluation of the segmentation process.

Compared with another well-known segmentation method, Fuzzy Kohonen Clustering Network (FKCN) [15], we can demonstrate the practicability of the proposed algorithm in the watermarking application by experiments (segmentation results and corresponding *Sim* values are shown in Fig. 5). Even though FKCN-based approaches produce good results in many applications [16,17,18], FKCN does not work very well in watermarking. One of the reasons is that FKCN is sensitive to noises, in other words, only slight alternations will influence the segmentation results (as shown in Fig. 5(b), different operations on Lena result in different segmented images), unless the used features are adequate and the training time is long enough to converge. This is because fluctuations occurred during the training process. To obtain more stable segmentation results, large amount of training iterations are needed. Therefore, the computational overhead is another drawback of FKCN. Moreover, some of the parameters in FKCN are not content-independent, which is impractical in watermarking applications. In the implementation of FKCN, the features of a given pixel are the graylevels of that pixel and its nearby pixels, and the maximum number of clusters *c* in the algorithm is set to 6. Other details of FKCN are omitted because they are not the focus of this paper.

## 3.2 Object-Segment Selection

The segmentation step produces several segments, from which we should select user-attentive object-segment as the area for watermark embedding. This idea relies on the truth that even the malicious attackers do not tamper the user-attentive area severely or the content will become valueless.

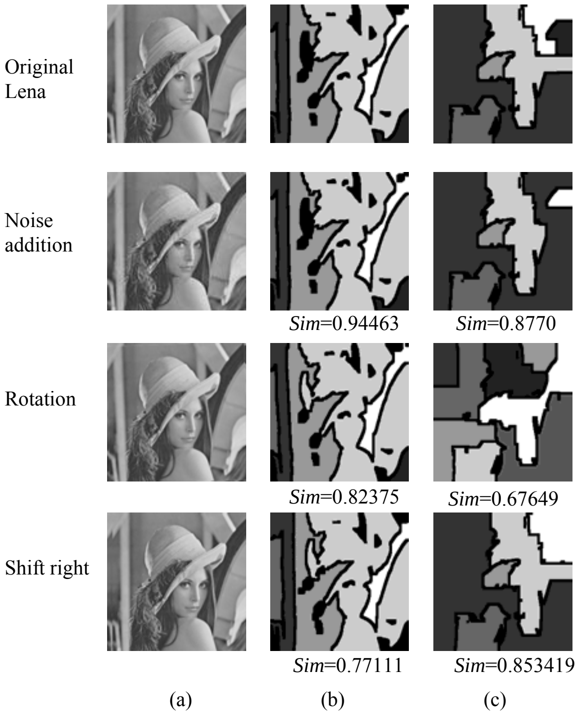


Fig. 5 The comparison between Geometric-Invariant Segmentation (GIS) and FKCN. Corresponding *Sim* values of GIS and FKCN after applying various attacks are listed below the attacked image: (a) images manipulated by geometrical operations or signal processing, (b) the resulting images segmented by GIS, and (c) by FKCN.

Human Visual System (HVS) has been introduced in many watermarking related researches. In [19] and [20], HVS has been used to improve the quality of watermarked image. In [21], HVS has been used to skip bits without influencing the visual perceptibility in video encoding applications. We apply the user-attentive model proposed in [20] to build an evaluation function which is used as the criterion for selecting of object-segments. In this user-attentive model, segments with mid-graylevels will have a high score for selection because areas with very high or low graylevels are less noticeable to humans. In addition, the strongly textured segments should get low scores. The distances to the image center are also considered because humans often focus on the area near the center of an image. Therefore, we evaluate user-attentive levels of segments by the values of $C_S$, $L_S$ and $T_S$, which estimate the closeness to the center points, the moderate of graylevels and the roughness of segment *S*, respectively. That is,

$$C_S = 1 - \frac{\sum_{(x,y)\in S} \sqrt{\left(\frac{x-x_c}{w}\right)^2 + \left(\frac{y-y_c}{h}\right)^2}}{|S|}, \qquad (2)$$

$$L_S = 1 - \frac{\sum_{(i,j)\in S} \left\lfloor \left( {}^{128 - I_{i,j}} \big/ {}_{128} \right)^2 \right\rceil^m}{|S|}, \tag{3}$$

$$T_S = \sum_{(k,l)\in S_B} \left( \frac{(C_{k,l})^2 - (C_{0,0})^2}{(C_{0,0})^2} \right), \tag{4}$$

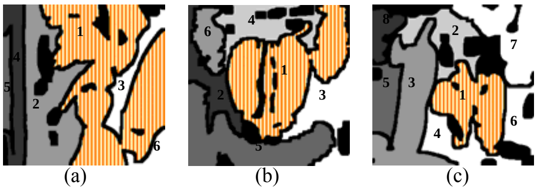
in which $I_{i,j}$ is the graylevel of the pixel (*i*,*j*), $C_{i,j}$ is the DCT coefficient of the positions (*k*,*l*) after applying the DCT to the minimum block containing *S* (denoted as $S_B$), *m* is an empirically determined constant, which adjusts the degree of the graylevel's influence (large *m* lengthens the distance among values of the base number), and |*S*| denotes the number of pixels in the segment *S*. The final evaluation function is

$$E = \alpha C_s + \beta L_s + \gamma T_s, \tag{5}$$

where $\alpha$, $\beta$, and $\gamma$ are also empirically decided constants, and result in different weights for $C_S$, $L_S$ and $T_S$. In our experiments, $\alpha$ is set to 0.6, $\beta$ is to 0.2, and $\gamma$ is to 0.2. After the Geometric-Invariant Segmentation, as described in Section 3.1, is done the function *E* is applied to each of the obtained segments, and the segment with the highest *E* is selected as the object-segment. Fig. 6 shows the experimental results of the proposed object-segment selection scheme, in which the numbers inside segments represent ranks of them. It can be found that the extracted object-segments are close to the attention area for humans, so the possibility of tampering these areas is very low; therefore, the embedded watermarks will have higher probability of surviving various attacks.

### 3.3 Object-Segment Alignment

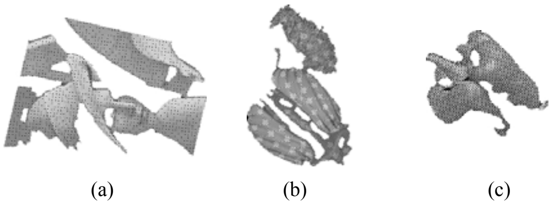
After deciding the user-attentive object-segment, the obtained object-segment is then aligned with its principal axes, which is derived by computing the corresponding eigenvector of the largest eigenvalue of the pixels in this segment. The purpose of aligning the object segment is to rotate the objects extracted from the host image and the watermarked image to the same direction. By means of object-segment alignment, even the watermarked image is distorted by the rotation attack, the embedded watermark can still be extracted correctly. After the alignment with the principle axes, an extra rotation of angle $\theta_s$ is performed for the purpose of security. $\theta_s$ can be seen as a secret key for decoding the watermark sequence.



(a) (b) (c)

Fig. 6 The results of the proposed object-segment selection scheme for three test images: (a) Lena, (b) Baboon, and (c) Peppers. Striped areas are object-segments and the attached numbers represent the ranks of segments on the basis of *E* values.

Some examples of the object alignment are illustrated in Fig. 7, in which the object-segments for images Lena, Baboon, and Peppers are aligned with their principle axes. The watermarking step then proceeds on the aligned object-segments, which will be detailed in the next section.



(a) (b) (c)

Fig. 7. Aligned object-segments for three test images: (a) Lena, (b) Baboon, and (c) Peppers.

## 4. Watermarking Scheme

In the proposed watermarking scheme, the embedding and extraction procedures are performed on the object-segment obtained in Section 3, which should be divided into *l* parts at first (*l* is the length of the watermark sequence). The watermark symbols are then embedded in the *l* sections in sequence. We propose three methods of object-segment division, the first one is the quarter-wise division, the second is the sector-wise division, and the third is the circle-wise division. Experiment results of the proposed watermarking scheme under three different object-segment division approaches will be given in Section 5.

### 4.1 Object-Segment Division

### 4.1.1 Quarter-Wise Division

The quarter-wise division firstly divides the object-segment into 4 parts with nearly the same size, and then divides each of the 4 parts recursively to obtain *l* parts for embedding the watermark sequence of *l* symbols. The division algorithm is stated as the following.

```
QuarterWise_Division(segment)
{
   ($x_{dc}$,$y_{dc}$)=Division_Center(segment);
   quarter_segments=Divide(($x_{dc}$,$y_{dc}$),segment);
   for $i$=1 to 4
      QuarterWise_Division(quarter_segments[$i$]);
}
```

The division-center ($x_{dc}$,$y_{dc}$) is computed by moving a point until the four divisions, which are divided by the horizontal and vertical line intersecting at that point, are nearly of equal-size (see Fig. 8). The "equaling" of the sizes of the 4 sub-segments in segment $S$ is measured by $\sum_{x\in S}(|x| - \frac{|S|}{4})^2$ . Fig. 9 gives some examples of the quarter-wise division.

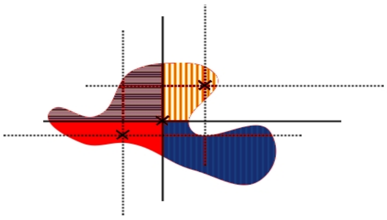


Fig. 8 Find the division center to divide the segments into 4 nearly equal-size parts.

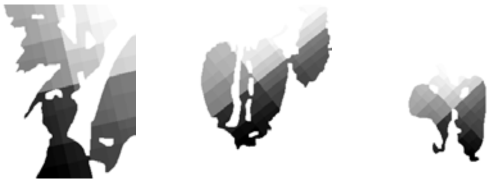


Fig. 9 Results of the quarter-wise division for three test images (the watermark-length $l$ is 64): (a) Lena, (b) Baboon, and (c) Peppers. Different sections are filled with different colors.

### 4.1.2 Sector-wise division

The sector-wise division divides the object-segment into $l$ equal-size sectors. First, drawing a minima circle, centers on the centre of the segment, and the radius is half of the image width. Then, the circle is split into $l$ sectors, in other words, the sector angle is $\frac{360°}{l}$. Fig. 10 shows the results of the sector-wise division.

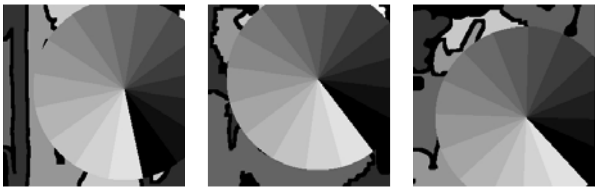


Fig. 10 Results of the sector-wise division for three test images (the watermark-length $l$ is 16): (a) Lena, (b) Baboon, and (c) Peppers. Different sections are filled with different colors.

### 4.1.3 Circle-Wise Division

The circle-wise division divides the object-segment into $l$ equal-size areas in a circular way. The reason for divide the object-segment into concentric circles is that rotations of the object will not affect the division results severely. The division steps are as the following:

First, drawing a circle, centers on the centre of the segment, and the radius $r_0$ is

$$r_0 = \frac{1}{\sqrt{l}} R , \tag{6}$$

where $R$ is the radius of the circular embedded region, which is set to half of the image width.

Second, a larger circle with radius $r_1 = (\sqrt{2} - \sqrt{1})r_0$ is drawn, and so on, then the radius of the $k$th circle is $r_{k-1} = (\sqrt{k} - \sqrt{k-1})r_0$ . Fig. 11 shows the results of the circle-wise division.

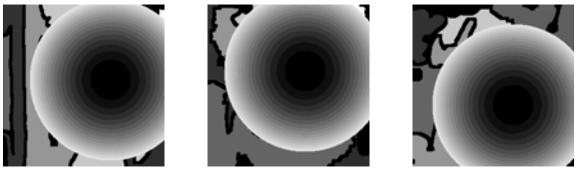


Fig. 11 Results of the circle-wise division for three test images (the watermark-length $l$ is 16): (a) Lena, (b) Baboon, and (c) Peppers. Different sections are filled with different colors.

## 4.2 Watermark Embedding

In this paper, we devise two watermark embedding algorithms: a quantization watermarking scheme in the spatial domain and a moment-based watermarking method. The first scheme embeds watermarks in the spatial domain by two quantization watermarking techniques, and the second one embeds watermarks in the moment-invariant. The embedding procedures are detailed in next two sections.

### 4.2.1 Quantization Watermarking in the Spatial-Domain

Spread-spectrum watermarking scheme [22] provides good performance on digital watermarking. However, sequential pseudorandom bits are involved in the embedding and extraction process. Consequently, the synchronization between original images and attacked images should be considered because some attacks (e.g. rotation and shifting) cause the disorder of the pseudorandom sequence. Therefore, quantization watermarking approaches and a majority decision method are utilized in this section rather than the spread-spectrum scheme. Two quantization watermarking techniques are employed in the embedding process:

First, *q*-ary symbols in the watermark are encoded by *q* levels of 1's sequences. That is, if the *i*th section of the object-segment has $l_i$ pixels, the step size of the quantization levels is $l_i/(q\text{-}1)$. We then embed $l_i\cdot(k\text{-}1)/(q\text{-}1)$ 1s for the *k*th symbol in the alphabet. For example, if the watermark is a binary sequence (i.e., *q*=2), the symbol "0" is encoded by a string 00...0 with length $l_i$ for the ith section, while the symbol "1" is encoded by 11...1.

The second quantization procedure is used for encoding 0 or 1 when embedding the binary string corresponding to one of the *q*-ary symbols. The graylevel is quantized to 255/*s* levels if the quantization step is *s*. As shown in Fig. 12, the odd zones represent 0 while the even ones represent 1. For instance, if 1 has to be embedded and the graylevel value of this pixel does not fall in even zones, then the graylevel value must be adjusted to the nearest even zone; but if the graylevel of current pixel is exactly in the even zone, the pixel value remains unchanged.

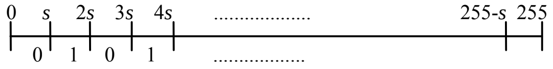


Fig. 12. The quantization of the graylevel in the watermark embedding.

### 4.2.2 Moment-Based Watermarking

Moment-based watermarking systems [8,9,10,11] take advantage of the affine-invariant properties of selected moments. [9] showed 11 moment-based functions are invariant to orthogonal or affine transformations, and used the good characteristics to construct an authentication system. But the system only embedded one-bit information in images. [8] normalized images before embedding and extraction, an additional comparison of central moments should be done in the detection step to find the geometric distortion of watermarked images. Furthermore, they improved the embedding capacity to 50 bits. The reason why those two schemes do not embed information directly in the selected moment-invariants is these invariants are not invertible. In the watermarking schemes in [10,11], Zernike moments are employed, which are invertible. Although Zernike moments are invertible, Kim and Lee [10] declared that images reconstructed from Zernike moments even with order 30 can hardly be recognized. In addition, it was very time consuming. Instead of embedding the watermark by modifying Zernike moments of the input image directly, they added watermarks into the intensity of the image using Zernike polynomials.

In the proposed moment-based watermarking, we choose the first and simplest orthogonal-invariant function with second order moments derived in [23] as the geometric-invariant feature (later we will use the same notations as [9] and the used moment-invariant is $\phi_1$). That is,

$$\phi_1 = \eta_{2,0} + \eta_{0,2}\,, \tag{7}$$

$$\text{where } \eta_{p,q} = \frac{\mu_{p,q}}{(\mu_{0,0})^{\gamma}}, \gamma = \frac{(p+q+2)}{2}, \tag{8}$$

$\eta_{p,q}$ is the normalized central moments and the central moment is

$$\mu_{p,q} = \iint_{\Gamma} (x-\bar{x})^p (y-\bar{y})^q f(x,y)dxdy\,, \tag{9}$$

where $\bar{x} = \frac{m_{1,0}}{m_{0,0}}, \bar{y} = \frac{m_{0,1}}{m_{0,0}}$, $m_{p,q}$ is the geometric moments of a grayscale image $f(x,y)$, and $\Gamma$ is the support of the image. $m_{p,q}$ is defined as

$$m_{p,q} = \iint_{\Gamma} x^p y^q f(x,y)dxdy\,. \tag{10}$$

Unfortunately, unlike the Zernike moments, $\phi_1$ is not invertible. However, it costs less computational overhead than Zernike moments. The watermark bit for each of the *l* sections is embedded by modifying graylevels of pixels inside that section toward the expected $\phi_1$. But how to change pixel graylevels to tune values of $\phi_1$?

It should be noted that if the variability of the graylevel *f*(*x*,*y*) is omitted, both of the two normalized central moments of $\phi_1$ involve variables of only one dimension. That is, $\eta_{2,0}$ has only *x*-dimension variables and $\eta_{0,2}$ has only *y*-dimension variables. We firstly consider the case with respect to the dimension *x*:

$$\begin{aligned} \eta_{2,0} &= \frac{\mu_{2,0}}{\mu_{0,0}^{\;2}}, \\ \mu_{2,0} &= \iint (x-\bar{x})^2 f(x,y)dxdy, \\ \mu_{0,0} &= \iint f(x,y)dxdy, \\ \bar{x} &= \frac{\iint xf(x,y)dxdy}{\iint f(x,y)dxdy}. \end{aligned} \tag{11}$$

Since values of all above 4 functions change only with *x* if *f*(*x*,*y*) is fixed, we reduce the problem to 1-dimensional case:

$$\eta_p = \frac{\mu_p}{\mu_0^{\gamma}}, \gamma = \frac{(p+2)}{2},$$

$$\mu_p = \int (x-\bar{x})^p f(x)dx\,, \tag{12}$$

$$m_p = \int x^p f(x) dx ,$$

then $\eta_2 = \frac{\mu_2}{\mu_0{}^2} = (m_2 - \frac{m_1{}^2}{m_0}) / m_0{}^2 = \frac{m_2 m_0 - m_1{}^2}{m_0{}^3}$. (13)

Without lose of generality, we firstly consider the case when *x*=*x'*. If the value of $f(x')$ is adjusted while $\forall x \in \Gamma \setminus \{x'\}$, $f(x)$ remains unchanged , then

$$\begin{aligned} m_0' &= m_0 + \Delta m_0 = m_0 + \Delta f(x'), \\ m_1' &= m_1 + \Delta m_1 = m_1 + x' \Delta f(x'), \\ m_2' &= m_2 + \Delta m_2 = m_2 + x'^2 \Delta f(x'), \end{aligned} \tag{14}$$

and

$$\eta_2' = \frac{m_2' m_0' - m_1'^2}{m_0'^3} = \frac{(m_2 + x'^2 \Delta)(m_0 + \Delta) - (m_1 + x' \Delta)^2}{(m_0 + \Delta)^3} \quad .(15)$$

(where $\Delta = \Delta f(x')$ for convenience)

$$\frac{d}{d\Delta} \eta_2' =$$

$$\Rightarrow \frac{1}{(m_0 + \Delta)^4} [(m_0{}^2 x'^2 - 2 m_0 m_1 x' - 2 m_0 m_2 + 3 m_1{}^2) \quad (15)$$
$$- 2(m_0 x^2 - 2 m_1 x + m_2) \Delta]$$

It can be derived that when

$\Delta = \Delta' = \frac{m_0{}^2 x'^2 - 2 m_0 m_1 x - 2 m_0 m_2 + 3 m_1{}^2}{2(m_0 x'^2 - 2 m_1 x' + m_2)}$ , $\eta_2'$ has the maximum or minimum value. In the implementation, the extreme value is maximal or minimal can be decided by comparing the value of $\eta_2'(\Delta')$ and $\eta_2'(\Delta), \Delta \neq \Delta'$ , rather than the complex analysis of the curve of $\eta_2'$. In fact, the absolute value of $\Delta'$ is often very large for real images, which is impractical for implementation. Nevertheless, we simply want to do the fine tuning instead of obtaining the maximum value, or the imperceptibility can not be preserved. Therefore, only the sign of $\Delta'$ is needed for determining the direction toward the maximum.

The analysis of dimension *y* can be done in the similar way. After iteratively modifying graylevels along *x* and *y* axes, the expected new $\eta_{2,0}$ and $\eta_{0,2}$ are obtained.

## 4.3 Watermark Extraction

In the proposed quantization watermarking scheme, the watermark can simply be extracted by the majority voting, that is, counting the number of 1s in each of the *l* sections in the object-segment. Obviously, the extraction procedure does not need the original image, that is, this scheme is blind-detectable.

In the extraction procedure for the proposed moment-based watermarking, *l* $\hat{\phi}_1$ s for every section in the object-segment of the suspected image have to be computed. Besides, $\phi_1$ s of the original image should be stored in advance for comparing with those of the suspected one to decide bits of the watermark. It should be noted that the whole host image is not needed in the extraction because the comparisons are not occurred in the pixels or other transform domains, only *l* values of $\phi_1$ s are necessary for detection.

The existence of the watermark is decided by the ratio of correctly detected bits $r_{cb}$ ( $r_{cb} = \frac{n_{cb}}{l_b}$ , where $n_{cb}$ is the number of correctly detected bits and lb is the total number of bits for the binary form of the watermark). That is,

$$\begin{cases} \text{if } r_{cb} \geq T, \text{ then the watermark is existed,} \\ \text{else there's no watermark in the image,} \end{cases} \tag{17}$$

where *T* is a detection threshold which determines the minimum votes of the majority.

# 5. Experiment Results

Three 256×256 graylevel images (Lena, Baboon, and Peppers) are used as the test data for the proposed watermarking system. We have three groups of experiments: one is for the quarter-wise division, another is for the sector-wise division, and the other is for the circle-wise approach. For each of these three division approaches, cooperating with two proposed embedding techniques, we test both the corresponding imperceptibility and robustness. The robustness of the resistance for geometric or signal processing is tested by performing some geometrical operations and signal processing on the embedded image which are shown in Fig. 13.

## 5.1 Results of the Quarter-Wise Division

We use a binary sequence of length 64 as the watermark for the quantization embedding and length 16 for the moment-based embedding, so the object-segment must be divided into 64 and 16 sections, respectively. PSNRs for the three test images are given in Table 1, which

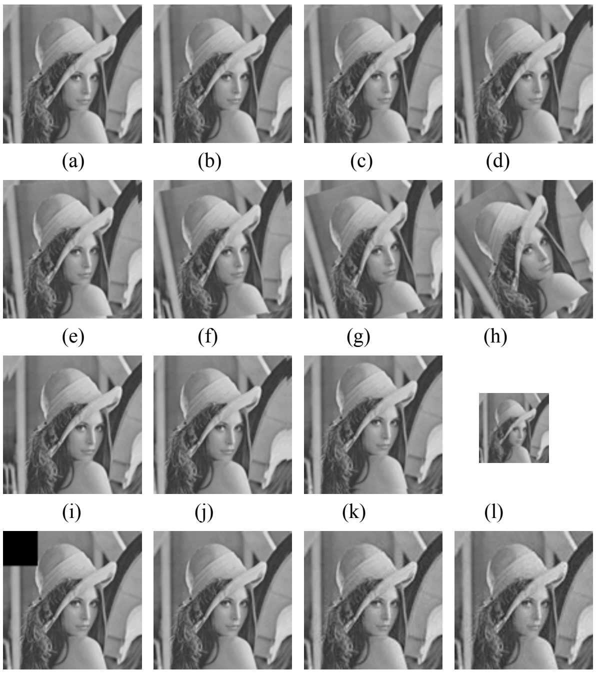


Fig. 13 Examples of images attacked by various geometrical alternations or image processing: (a) rotation 1°, (b) rotation 2°, (c) rotation 3°, (d) rotation 4°, (e) rotation 5°, (f) rotation 10°, (g) rotation 15°, (h) rotation 30°, (i) shifting right 10 pixels, (j) shifting left 10 pixels, (k) shifting up 10 pixels, (l) scaling $\frac{1}{2} \times \frac{1}{2}$ (m) cropping 1/16, (n) JPEG 50%, (o) noise addition 5%, and (p) noise addition 10%.

shows the quality of the watermarked image is rather good (Q and M denotes the quantization watermarking and moment-based watermarking, respectively). Table 2 presents the fidelity of this watermarking scheme. It can be found that the watermark can survive most of the geometric and signal processing manipulations. For the quantization watermarking, an additional experiment is performed by randomly selecting 10 unwatermarked images (3 of them are Lena, Baboon, and Peppers), and applying the same watermark extraction on them, the resulting average $r_{cb}$ is 48.44%. Then this watermarking scheme is performed on all 64 test images (in which 10 are unwatermarked images, 3 are watermarked images with no attacks, and 51 are attacked watermarked images) to examine the overall performance. If the detection threshold $T$ is set to 55%, the miss ratio is 5.56% and the false alarm rate is 0. It should be noted that 51 of the all 64 test images are attacked by various geometric or signal processing manipulations.

Table 1: The PSNRs in experiments for the quarter-wise division.

| Test Image | Lena | | Baboon | | Peppers | |
|---|---|---|---|---|---|---|
| Embedding method | Q | M | Q | M | Q | M |
| PSNR(dB) | 41.67 | 37.77 | 43.67 | 39.75 | 46.70 | 43.06 |

Table 2: The values of $r_{cb}$ in experiments for the quarter-wise division.

| Test image | Lena | | Baboon | | Peppers | |
|---|---|---|---|---|---|---|
| Embeddi ng method | Q | M | Q | M | Q | M |
| rotation 1° | 95.31% | 81.25% | 79.69% | 87.50% | 95.31% | 87.50% |
| rotation 2° | 96.88% | 81.25% | 89.06% | 81.25% | 92.19% | 87.50% |
| rotation 3° | 92.18% | 87.50% | 89.06% | 81.25% | 95.31% | 68.75% |
| rotation 4° | 93.75% | 81.25% | 87.5% | 75.00% | 93.75% | 68.75% |
| rotation 5° | 93.75% | 87.50% | 89.06% | 81.25% | 90.63% | 68.75% |
| rotation 10° | 81.25% | 62.50% | 89.06% | 81.25% | 64.06% | 50.00% |
| rotation 15° | 57.81% | 50.00% | 79.69% | 81.25% | 59.38% | 43.75% |
| rotation 30° | 70.31% | 37.50% | 50% | 31.25% | 59.38% | 43.75% |
| shifting right 10 | 95.31% | 75.00% | 100% | 81.25% | 100% | 87.50% |
| shifting left 10 | 76.56% | 75.00% | 100% | 56.25% | 100% | 87.50% |
| shifting up 10 | 93.75% | 62.50% | 92.19% | 81.25% | 100% | 68.75% |
| scaling 2×2 | 96.88% | 81.25% | 90.63% | 93.75% | 98.44% | 68.75% |
| scaling $\frac{1}{2} \times \frac{1}{2}$ | 84.38% | 100% | 56.25% | 81.25% | 65.63% | 68.75% |
| cropping 1/16 | 92.19% | 81.25% | 59.38% | 56.25% | 79.69% | 68.75% |
| JPEG 50% | 96.88% | 93.75% | 100% | 93.75% | 100% | 87.50% |
| noise additio | 93.75% | 87.50% | 100% | 75.00% | 100% | 93.75% |
| noise additio | 60.94% | 93.75% | 51.56% | 56.25% | 51.56% | 81.25% |

## 5.2 Results of the Sector-Wise Division

The areas nearby the centre of the embedding circle for each of $l$ sections are so small that few errors from segmentations or object-alignments will cause incorrect detection. Therefore, in the sector-wise division, we shorten the watermark sequence to 16 bits to ensure every section has enough space for watermark embedding.

Table 3: The PSNRs in experiments for the sector-wise division.

| Test Image | Lena | | Baboon | | Peppers | |
|---|---|---|---|---|---|---|
| Embedding method | Q | M | Q | M | Q | M |
| PSNR(dB) | 39.17 | 38.31 | 39.40 | 39.21 | 39.54 | 35.13 |

Table 4: The values of $r_{cb}$ in experiments for the sector-wise division.

| Attacks | Lena | | Baboon | | Peppers | |
|---|---|---|---|---|---|---|
| Embedding method / Attacks | Q | M | Q | M | Q | M |
| rotation 1° | 100% | 100% | 100% | 81.25% | 100% | 93.75% |
| rotation 2° | 100% | 87.5% | 100% | 87.5% | 100% | 93.75% |
| rotation 3° | 100% | 93.75% | 100% | 68.75% | 100% | 75% |
| rotation 4° | 100% | 100% | 100% | 62.5% | 100% | 43.75% |
| rotation 5° | 100% | 100% | 93.75% | 68.75% | 100% | 43.75% |
| rotation 10° | 100% | 100% | 93.75% | 43.75% | 43.75% | 31.25% |
| rotation 15° | 62.5% | 56.25% | 87.5% | 50% | 37.5% | 43.75% |
| rotation 30° | 100% | 81.25% | 43.75% | 37.5% | 43.75% | 43.75% |
| shifting right 10 | 100% | 68.75% | 100% | 100% | 100% | 93.75% |
| shifting left 10 | 100% | 85.5% | 100% | 68.75% | 100% | 100% |
| shifting up 10 | 100% | 75% | 100% | 50% | 100% | 93.75% |
| scaling 2×2 | 100% | 100% | 100% | 81.25% | 100% | 93.75% |
| scaling $\frac{1}{2}\times\frac{1}{2}$ | 100% | 100% | 68.5% | 75% | 100% | 93.75% |
| cropping 1/16 | 100% | 68.75% | 56.25% | 31.25% | 100% | 37.5% |
| JPEG 50% | 100% | 100% | 100% | 75% | 100% | 93.75% |
| noise addition | 100% | 100% | 100% | 93.75% | 100% | 93.75% |
| noise addition | 100% | 100% | 75% | 31.25% | 87.5% | 93.75% |

Table 3 shows the PSNRs of test images and Table 4 presents the robustness of this watermarking scheme. Although the capacity of this scheme is less than the previous one, the fidelity is superior because the embedding area for each of the watermark bits is much larger than the quarter-wise method and the inherent circular features of the sector-wise approach can aid the resistance to rotation attacks. In the experiments for the quantization watermarking, we also select 10 unwatermarked images and the average $r_{cb}$ is 48.13%. The test data is the same 64 images as Section 5.1. If the detection threshold $T$ is set to 57%, the miss ratio is 9.3% and the false alarm rate is 10%.

Table 5: The PSNRs in experiments for the circle-wise division.

| Test Image | Lena | | Baboon | | Peppers | |
|---|---|---|---|---|---|---|
| Embedding method | Q | M | Q | M | Q | M |
| PSNR(dB) | 39.35 | 44.14 | 39.39 | 45.18 | 39.54 | 49.25 |

Table 6: The values of $r_{cb}$ in experiments for the circle-wise division.

| Attacks | Lena | | Baboon | | Peppers | |
|---|---|---|---|---|---|---|
| Embedding method / Attacks | Q | M | Q | M | Q | M |
| rotation 1° | 81.25% | 75.00% | 93.75% | 81.25% | 100% | 68.75% |
| rotation 2° | 87.5% | 56.25% | 100% | 62.50% | 100% | 100.00% |
| rotation 3° | 100% | 68.75% | 100% | 87.50% | 100% | 81.25% |
| rotation 4° | 100% | 50.00% | 100% | 62.50% | 100% | 62.50% |
| rotation 5° | 100% | 50.00% | 100% | 56.25% | 100% | 56.25% |
| rotation 10° | 100% | 68.75% | 100% | 68.75% | 100% | 62.50% |
| rotation 15° | 56.25% | 31.25% | 93.75% | 50.00% | 100% | 62.50% |
| rotation 30° | 68.75% | 31.25% | 43.75% | 56.25% | 100% | 50.00% |
| shifting right 10 | 62.5% | 50.00% | 100% | 81.25% | 100% | 93.75% |
| shifting left 10 | 75% | 56.25% | 100% | 50.00% | 100% | 87.50% |
| shifting up 10 | 62.5% | 25.00% | 100% | 62.50% | 100% | 93.75% |
| scaling 2×2 | 100% | 93.75% | 100% | 87.50% | 100% | 100.00% |
| scaling $\frac{1}{2}\times\frac{1}{2}$ | 81.25% | 93.75% | 75% | 81.25% | 100% | 81.25% |
| cropping 1/16 | 81.25% | 37.50% | 50% | 68.75% | 100% | 62.50% |
| JPEG 50% | 100% | 75.00% | 100% | 75% | 100% | 93.75% |
| noise addition | 100% | 93.75% | 100% | 56.25% | 100% | 81.25% |
| noise addition | 87.5% | 43.75% | 81.25% | 62.5% | 43.75% | 87.50% |

## 5.3 Results of the Circle-Wise Division

In the circle-wise division, errors introduced from segmentation and object-extractions may affect the watermarking accuracy severely if we divide the embedding area into too many donuts. It is because the deviation of the object-segment's center in the attacked image produces concentric circles different from those of the original image, and the ith donut may cover the area of jth donut in the original image ( $i \neq j$ ). Therefore, in the experiments, the watermark sequence has the same length as the sector-wise division (i.e. 16 bits) to avoid the mentioned problem.

Table 5 shows the PSNRs of test images and Table 6 presents the robustness of this watermarking scheme. Compared with the sector-wise scheme, which uses watermark sequences with the same length as the circle-wise approach, it can be found that the circle-wise method has higher performance on the resistance to rotation, but is weaker against shifting attacks. In the experiments for the quantization watermarking, the average $r_{cb}$ of 10 unwatermarked images is 47.16%. If the same 64 images are tested and the detection threshold $T$ is set to 57%, the miss ratio is 5.56% and the false alarm rate is 0.

### 5.4 Discussions

From the experiment results, it can be found that the proposed watermarking scheme with quantization embedding provides good performance on both imperceptibility and robustness against geometrical distortions and digital processings. In addition, unlike many geometric-invariant watermarking schemes based on moment invariants or in Fourier domain, the proposed watermarking techniques are performed without high computational complexity and difficulty in embedding information. Although traditional spatial domain watermarking suffers the risk of fragility if watermarked images are altered by digital processings, the applied object-oriented watermarking method prevents malicious attackers from tamper the information embedding area. Therefore, the watermark will not be tampered severely, or the image quality will decay. Moreover, the proposed quantization watermarking approach provides the robustness against various attacks and experiment results support this proposition.

The watermarking scheme with another proposed embedding technique, i.e. the moment-based watermarking, also has acceptable correctness of watermark detection, even though it does not have so good performance as the quantization watermarking. Moreover, its time complexity is not so high as traditional moment-based watermarking approaches because only 2-order moments are employed and the computation is only performed on the small zone of the embedding section.

## 6. Conclusions

In this paper, we have described a new watermarking scheme which is simple, computational-efficient, blind-detectable, robust, and has a reasonable embedding capacity. The proposed watermarking scheme is based on the object-oriented embedding, which treats an image as a set of objects, rather than pixels. We first extract the user-attentive object-segments from the image, and then the watermark is embedded in this area, which will be tampered by attackers with low possibility. In the object-extraction step, we propose a Geometric-Invariant Segmentation, which is stable and insensitive to geometrical operations and signal processing. In the watermark embedding procedure, object-segments should be divided into several sections. We devise three dividing methods: the quarter-wise division, the sector-wise division, and the circle division. After the object-segment is divided, the watermark sequence is embedded by quantization or moment-based embedding. Experiment results show the proposed watermarking scheme indeed has good performances on imperceptibility, robustness, computational efficiency, and capacity, which implies the applicability of this method.

### Acknowledgments

This work was partially supported by the National Science Council and the Ministry of Education of ROC under the contract No. NSC92-2622-E-002-002, NSC92-2213-E-002-023, 89E-FA06-2-4-8 and NSC93-2752-E-002-006-PAE . We would also like to acknowledge Dr. Chun-Hsiang Huang for his helpful discussions and valuable suggestions.

## References

[1] J.J. K. Ŏ Ruanaidh and Pun, "Rotation, Scale and Translation Invariant Spread Spectrum Digital Image Watermarking", *Signal Processing*, Vol. 66, pp. 303-317, 1998.

[2] 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.

[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 Iimages", *IEEE Trans. on Image Processing*, Vol. 10, No. 5, pp. 767-782, 2001.

[4] C. T. Hsu and J. L. Wu, "Hidden Digital Watermarks in Images", *IEEE Trans. on Image Processing*, Vol. 8, No. 1, pp.58-68, 1999.

[5] D. Y. Chen, M. Ouhyoung, and J. L. Wu, "A shift-resisting Public Watermark System for Protecting Image Processing Software", *IEEE Trans. on Consumer Electronics*, Vol. 46, No. 3, pp.404-414, 2000.

[6] P. Bas, J. M. Chassery, and B. Macq, "Geometrically Invariant Watermarking Using Feature Points", *IEEE Trans. on Image Processing*, Vol. 11, No. 9, pp. 1014-1-027, 2002.

[7] C. W. Tang and H. M. Hang, "A Feature-Bbased Robust Digital Image Watermarking scheme", *IEEE Trans. on Signal Processing*, Vol. 51, No. 4, pp.950-959, 2003.

[8] H. I. Kang and E. J. Delp, "An Image Normalization Based Watermarking Scheme Robust to General Affine Transfromation", *ICIP 2004*, pp. 1553-1556.

[9] M. Alghoniemy and A. H. Tewfik, "Geometric Invariance in Image Watermarking", *IEEE Trans. on Image Processing*, Vol. 13, No. 2, pp. 145-153, 2004.

[10] H. S. Kim and H. K. Lee, "Invariant Image Watermark Using Zernike Moments", *IEEE Trans. on Circuits and Systems for Video Technology*, Vol. 13, No. 8, pp. 766-775, 2003.

[11] Y. Xin, S. Liao, and M. Pawlak, "A Multibit Geometrically Robust Image Watermark Based on Zernike Moments", *IEEE ICPR 2004.*

[12] Y. Wang and A. Pearmain, "Blind MPEG-2 Video Watermarking Robust Against Scaling", *IEEE ICIP 2004*, pp. 2159-2162.

[13] C. V. Serdean, M. A. Ambroze, M. Tomlinson and J. G. Wade, "DWT-Based High-Capacity Blind Video Watermarking Invariant to Geometrical Attacks", *IEE Proc. VISP*, Vol. 150, No. 1, pp. 51-58, 2003.

[14] B. Chen and G. W. Wornell, "Quantization Index Modulation: A Class of Provably Good Methods for Digital Watermarking and Information Embedding", *IEEE Trans. Information Theory*, Vol. 47, pp. 1423-1443, May 2001.

[15] J. C. Bezdek, E. C.-L. Tsao and N. R. Pal, "Fuzzy Kohonen Clustering Networks", *IEEE ICFS 1992*, pp.1035 – 1043, 1992.

[16] H. Atmca, M. Bulut, and D. Demir, "Histogram Based Fuzzy Kononen Clustering Network for Image Segmentation", *ICIP 1996*, Vol. 2, pp. 951-954.

[17] C. C. Lin, J. R. Duann, H. C. Cheng, and J. H. Chen, "A Cascade Algorithm Combined Kohonen Feature Map with Fuzzy C-Means Applied in MR Brain Image Segmentation", *18th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Amsterdam 1996*, Vol. 3, pp. 1079-1080, 1996.

[18] L. Cinque, G. Foresti, and L. Lombardi, "A Clustering Fuzzy Approach for Image Segmentation", *Pattern Recognition*, Vol. 37, pp. 1797-1807, 2004.

[19] M. D. Swanson, B. Zhu, and A. H. Tewfik, "Multiresolution Scene-Based Video Watermarking Using Perceptual Models", *IEEE J. SAC*, Vol. 16, pp.540-550, 1998.

[20] M. S. Kankanhalli and K. R. Ramakrishnan, "Content Based Watermarking of Images", *ACM Multimedia'98*, pp.61-70, 1998.

[21] W. S. Geisler and J. S. Perry, "A Real-Time Foveated Multiresolution System for Low-Bandwidth Video Communication", *SPIE proceedings: Human Vision and Electronic Imaging* (*VCIP'98*), pp.294-305, 1998.

[22] I. Cox, J. Kilian, F. Leighton, and T. Shamoon, "Secure Spectrum Watermarking for Multimedia", *IEEE Trans. on Image Processing*, Vol. 6, No. 12, pp. 1673-1687, Dec. 1997.

[23] M. K. Hu, "Visual Pattern Recognition by Moment Invariants", *IRE Trans. on Information Theory*, Vol. 8, pp. 179-187, 1962.

**Yu-Tzu Lin** received the B.S. and M.S. degrees in information and computer education from National Taiwan Normal University, Taipei, Taiwan, R.O.C., in 1994 and 1997. She is currently pursuing the Ph.D. degree in the Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, R.O.C. Her research interests include copyright protection for multimedia, pattern recognition, and image processing.

**Ja-Ling Wu** received the B.S. degree in electronic engineering from TamKang University, Tamshoei, Taiwan, R.O.C., in 1979, and the M.S. and Ph.D degrees in electrical engineering from Tatung Institute of Technology, Taipei, Taiwan, in 1981 and 1986. Since 1987, he has been with the Department of Computer Science and Information Engineering, National Taiwan University, where he is presently a Professor. He has published more than 200 journal and conference papers. His research interests include algorithm design for DSP, data compression, digital watermarking and multimedia systems. Prof. Wu was the recipient of the Excellent Research Award from NSC, Taiwan, in 1999, 2001 and 2004.

**Yu-Feng Kuo** received the B.S. in computer science and information engineering from National Taiwan University, Taipei, Taiwan, R.O.C., in 2005, where he is currently pursuing the M.S. degree in the Communication and Multimedia Laboratory , CSIE, NTU.