Region-Based Image Watermarking

Athanasios Nikolaidis and Ioannis Pitas, *Senior Member, IEEE*

***Abstract*—We introduce a novel method for embedding and detecting a chaotic watermark in the digital spatial image domain, based on segmenting the image and locating regions that are robust to several image manipulations. The robustness of the method is confirmed by experimental results that display the immunity of the embedded watermark to several kinds of attacks, such as compression, filtering, scaling, cropping, and rotation.**

***Index Terms*—Chaos, copyright protection, correlators, feature extraction, image segmentation, signal detection.**

I. Introduction

PROTECTION of multimedia information has attracted a lot of attention during the last few years. The aim of such methods is to protect the copyright of broadcast or publicly exposed multimedia data. Attackers have the freedom to obtain copies of copyrighted electronic material via the Internet and manipulate them at will. The most popular method to protect copyright information is watermarking. The main requirements for an acceptable technique of watermarking are [1], [2] as follows.

1) *Imperceptibility*: the watermark should not be easily noticed by simple visual inspection.
2) *Key uniqueness*: different keys should produce different, statistically independent watermarks.
3) *Noninvertibility*: it should not be computationally feasible to find the watermark by possessing a watermarked image.
4) *Image dependency*: a single key produces a single watermark; however, this watermark should be adapted to the image content.
5) *Reliable detection*: the watermark should be efficiently detected for any value of false alarm probability up to a certain threshold.
6) *Robustness*: the watermark should be efficiently detected after most common signal processing operations.

Usually a tradeoff is necessary between watermark imperceptibility and robustness. Most of the proposed techniques easily meet the imperceptibility demand. However, most of them do not consider simultaneous robustness to several kinds of attacks. Many of them focus on robustness against JPEG or other compression techniques, noise addition, and lowpass filtering [3]–[5], while others only attempt to face geometric distortions efficiently [6]. None of them has covered the entire range of different processing attacks at the same time, except when resorting to the original image [7]. These techniques are either applied in the spatial digital image domain or in some image transform domain (e.g., DCT, DFT, DWT, etc.). In [8], a set of attacks is proposed that any watermarking scheme should survive.

Previous methods have failed in providing robust behavior under many commonly considered attacks mainly, because they attempted to face the image, audio or video signal in a global sense, without exploiting their local characteristics. In the case of image watermarking, employing spatial characteristics is essential for ensuring immunity to geometric transformations. When a watermark is embedded on the entire image, scaling, rotation or cropping will result in the destruction of the watermark because no reference points exist that would lead in finding the amount of scaling, rotation or cropping. The use of an image transform, with the exception of the Fourier transform, will suffer the same problems. The Fourier transform is theoretically rotation, translation and scale invariant, but the robustness to filtering or compression depends on the range of frequencies that are used for watermarking.

In this paper, we propose the use of salient spatial features resulting from image segmentation, so that they will be used as reference for compensating usual geometric attacks. Image segmentation is a powerful image processing tool that can provide us with useful information about the spatial image content. The produced regions are arranged according to their size and the largest of them are selected for watermark embedding. The selected regions are approximated by ellipsoids, by using a neural network technique. Most likely this representation will not be seriously distorted after image manipulation and, thus, the ordering of the regions will not be affected either. The bounding rectangle of each region is used for watermark embedding. The parameters of this rectangle include its center coordinates, orientation, width and height. These characteristics prove significantly immune to the considered attacks, and ensure robustness to many geometric distortions. A local search for a small range of values for each of the previously referenced parameters is sufficient for watermark retrieval. Finally, the use of a certain chaotic system that produces trajectories of controlled lowpass properties is suggested in order to preserve the robustness of the method to manipulations such as filtering, noise addition and compression. This choice is preferred to the use of a usual pseudo-random number generator, because the latter produces uniformly distributed trajectories that have a white spectrum and cannot withstand lowpass filtering.

An approach that is somehow related to our method was proposed in [9], where the core of the technique was to find image points that could be warped according to their distance to specific line segments that form the watermark. However, the cost

Manuscript received June 17, 1999; revised July 26, 2001. This work was carried out within the framework of the LTR-ESPRIT European Project 31103-INSPECT. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Naohisa Ohta.

The authors are with the Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki 540 06, Greece (e-mail: pitas@zeus.csd.auth.gr).

Publisher Item Identifier S 1057-7149(01)09364-2.

of producing a theoretically infinite set of different line patterns in the detection stage, together with the questionable prominence of the selected points to be warped, render this technique unsuitable. The main weak aspect of the technique [9] is that the chosen pattern is completely random and does not take any spatial properties into consideration. Another method that tries to face the problem of geometric attacks is presented in [10], where a template consisting of a fixed number of points is inserted in the DFT domain of the image. However, these points can easily be distorted by calculating the local mean value and standard deviation of the coefficients and change accordingly the strength of the predominant DFT coefficients. In [11] a block-based attack-resilient watermark decoder is proposed, that is based on a sliding correlator window that is scaled, rotated, and translated until the maximum correlation to the original pseudo-random watermark is found. This implies an increased computation time because of the complete ignorance about the spatial localization of the watermark that results in a multidimensional search space of quite many parameters spanning a wide range of values. The approach presented in [12] suggests a scheme that modifies only the blue channel of a color image for imperceptibility reasons. Each watermark bit is embedded in many positions to ensure robustness. However, these positions are randomly chosen according to the watermark key and, thus, the immunity of their luminance values to attacks is not ensured.

This paper is organized as follows: Section II presents the watermarking requirements and the techniques that are used to meet them. Section III presents the adaptive K-means clustering and region approximation technique, which is used as a preprocessing step for the determination of the spatial constraints to be used in the embedding and detection stages. In Section IV, the general class of chaotic watermarks is presented together with adaptations for digital images, followed by an explanation of the connection between the spatial features and the watermark to be embedded on the image. Section V presents the watermark detection procedure. Simulation results for several watermarked image manipulations are presented and explained in Section VI. Finally, conclusions are drawn and future work is addressed in Section VII.

II. Technique Outline

The current paper aims at providing a watermarking technique that faces the image copyright protection problem from the viewpoint of selecting only interesting image regions for embedding. We perform region-based image watermarking for the following reasons.

1) This watermarking technique is consistent with the object-based coding/description approach followed in MPEG4 and MPEG7 (although the proposed method is described for still images only).
2) In some cases, only certain image regions have to be protected (e.g., in portraits). However, more regions than the required ones can be used to embed the same prototype watermark after the proper adaptation to each region, thus enhancing the performance of the detection stage. If any of these regions are cropped and pasted to another image, the respective watermark should remain intact.
3) The output of certain region segmentation techniques can be proven robust to certain geometrical transforms and other image processing operations. The outline of the developed technique is as follows.
   - *Feature selection*: This stage is concerned with the preprocessing that is necessary to extract the spatial image characteristics that are needed for the watermark embedding/detection stage.
     —*Image segmentation*: In this step, a clustering technique based on the well-known ICM (iterated conditional modes) method is used, having a minimum number of required parameters. The employed technique is a variation of the one proposed in [13], where both spatial constraints and local intensity variations are used in order to enhance the performance of the classic K-means algorithm. This technique provides a segmentation of the image into a rather small number of large regions that are suitable for watermarking.
     —*Feature detection*: After the image has been segmented, its inner segments are sorted in the order of decreasing area (measured in number of pixels) and the largest of them are selected for the watermarking process. The selected regions are afterwards coarsely approximated by ellipsoids that are constructed by employing a neural network technique. The orientation, center and dimensions of the bounding rectangles of these ellipsoids are finally used as input for watermark embedding.
   - *Watermark embedding/detection*:
     —*Chaotic watermark embedding*: A chaotic watermark that is constructed by Peano-scanning of a one-dimensional (1-D) chaotic trajectory [14] is embedded according to the geometric information arising from the previous stage. The watermark is embedded to each of the bounding rectangles corresponding to the selected regions of the previous stage.
     —*Chaotic watermark detection*: A watermark detector based on the correlation of a watermark template with the possibly watermarked and processed image is proposed. This detector acts on the regions segmented from the watermarked and processed image. Consequently the robustness of the localization of the spatial features after various attacks on the watermarked image ensures the robustness of the detection process. In this way, only small local searches in the geometric parameter space are required to find the correct position of the embedded watermark.

III. Adaptive K-Means Clustering and Region-Based Spatial Feature Determination

The first stage of the proposed technique concerns finding a segmentation or clustering technique that will provide us with

a robust region representation under image processing, in the sense that it will not be seriously affected by usual geometric image manipulations (e.g., rotation, translation, compression, filtering). Many classes of segmentation techniques have been studied to this end and extensive experimentation has been made. Morphological watershed techniques [15]–[17], for instance, produce oversegmentation, because they are very sensitive to local luminance changes and, thus, catchment basins are very densely constructed. This is against our need for producing large regions that cover a significant percentage of the total image area. Region-growing methods start up from seed pixels and advance based on luminance similarity of neighboring pixels to each growing region. This, however, may result in noncompact regions. Split-merge techniques [18], provide rather coarse region estimates, which are very prone to changes in luminance. This results in many regions either being split to others of quite smaller size, or merged to regions of quite larger size, because of the hierarchical nature of this method. Finally, histogram-based techniques are not reliable because, if constant split values are chosen on the histogram, they will produce noncompact regions and many spurious pixels. Furthermore, the method will not be robust against attacks, because after manipulation the peaks correspond to different greylevels on the histogram.

The technique which was chosen after experimentation as the most robust one is a multilevel implementation of the adaptive clustering method proposed in [13]. This is a variation of the ICM algorithm that was introduced by Besag [19]. This technique works well especially on images containing objects with smooth surfaces. The algorithm may not be optimal in the case of some textured images, because no clear distinction between objects is possible in this case. However, the merging step that completes the algorithm may provide a set of regions that do not correspond to real objects. This is of no concern to us, because we do not need a representation that is as detailed as possible. The merging step serves to provide a final segmentation containing a quite reduced number of regions that are approximately as large as required. A modified approach of the clustering technique is presented in [20] for color images, where first-order and second-order derivatives are incorporated. We will not employ this refined technique, because we need an output that is not too sensitive to changes in discontinuities and homogeneity constraints.

The initial image is first subsampled by a factor that depends on its original size. A classical K-means algorithm is then performed on the luminance component of the decimated image version. The initial values for the cluster centers are not randomly defined, as in the classical implementation. They are chosen among the most prominent image histogram peaks over a sufficient neighborhood of histogram values. This ensures that the segmentation algorithm will converge always to the values that provide a classification that is as close to the original as possible, since any possible attack would affect the image greylevels in a way that will translate the peak values. Even though the peak values may be modified after certain attacks, they are only used to produce the region seeds and do not change segmentation results significantly.

Let K denote the number of clusters in which the image pixels are to be classified. The classical K-means algorithm provides a coarse segmentation estimate which is noisy, because spurious pixels are assigned to different clusters and image regions that are somehow interrelated may be disconnected. We wish to obtain a smoothened segmentation output containing a rather small number of large regions that would be suitable for spatial watermark embedding. The approach in [13] presents an adaptive method that takes under consideration both similarity potentials between current and neighboring pixel cluster assignments, as well as greylevel relation between current pixel and possible centers. The second constraint, when employed alone, describes the classical K-means algorithm. The similarity potentials are defined in such a way that two neighboring pixels are more likely to belong to the same cluster than to different clusters, especially when they are 4-neighbors. The distance metric employed in our case is the Euclidean one. Bayes theorem can provide us with a model of the *a posteriori* probability density function that describes the desired segmentation. By maximizing this probability with respect to the cluster center, each pixel is assigned to a certain cluster. By choosing proper values for the potential parameter β, we can achieve a segmentation result that is noise free and contains regions having quite smooth borders.

After the ICM step, a region merging process according to the mean value similarity between adjacent regions is employed in order to eliminate useless small regions. We consider that a region should be eliminated if it covers an area that is less than a certain percentage of the total image area (e.g., 10%). A segmentation result is shown in Fig. 1. Fig. 1(a) shows the original image of size 800×800 and Fig. 1(b) shows the final segmentation result, after the small region elimination stage. The number of clusters is $K = 4$ and the potential parameter is $\beta = 2$. The several regions (which are seven in this case) are represented by different grey levels.

After the segmentation process, the resulting regions are ordered according to their size, excluding the ones that are on the image boundaries, in order to avoid problems arising from image cropping along borders. The largest regions are preferred for watermarking, in order to preserve as much of the watermark power as possible. These regions also tend to be more robust as far as their size, shape and orientation are concerned. Fig. 1(c) shows the two largest regions of the above referenced image, excluding the regions lying at the image borders. Certainly there are images (e.g., landscape images) for which it would not be suitable to exclude border regions. Cropping can be either symmetric or nonsymmetric. Some rows or columns may be cut away, provided that at least one of the initially watermarked regions is not affected.

We propose a robust representation of the selected regions by ellipsoid approximation. The resulting ellipsoids are more easily described than the regions themselves, by means of their center coordinates, width, height and orientation. In addition, since the embedded watermark is a rectangular pattern, we choose to use the bounding rectangle of each ellipsoid for watermark embedding/detection. An α-trimmed Mean Radial Basis Function network as defined in [21] was employed in order to approximate each selected region by an ellipsoid. Each region corresponds to a hidden unit of the network. The marginal data samples are first ordered according to their Mahalanobis distance

$$r^2 = (\mathbf{X} - \hat{\mu}_k)' \hat{\mathbf{\Sigma}}_k^{-1} (\mathbf{X} - \hat{\mu}_k) \tag{1}$$

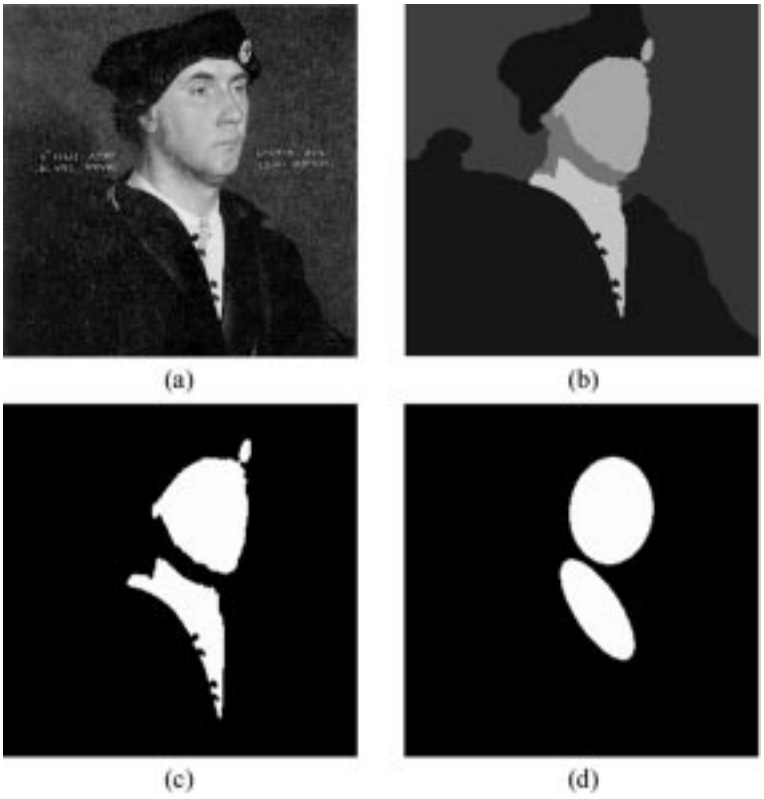

Fig. 1. Segmentation and region approximation by ellipsoid. (a) Original image. (b) Segmentation result. (c) Two largest regions of the segmented image. (d) Ellipsoid approximation of the regions in (c).

where $\mathbf{X}$ is the marginal data sample, $\hat{\mu}_k$ is the center of the kth cluster (or, equivalently, the kth hidden unit of the network), and $\hat{\mathbf{\Sigma}}_k$ is its covariance matrix. After ordering the samples, a percentage α of them can be trimmed away from both ends of the data distribution. The center and covariance matrix of the new data set can be computed. These determine the ellipsoid approximation of each region. A result is shown in Fig. 1(d). We can notice that the ellipse describes mainly the body of each region.

Once the trimmed ellipsoidal approximation is known, its orientation and its bounding rectangle can easily be computed [22, p. 393]. The rectangle defines the area where the watermark is to be embedded or detected. Knowledge of the center coordinates, dimensions and orientation of the bounding rectangle compensates for cropping, scaling and rotation of the watermarked image, respectively. These parameters are the output of the segmentation stage and are going to be used in both watermark embedding and detection stages.

IV. Watermark Construction and Embedding

In Section III, we developed a method of locating robust regions, so that they can be used as reference areas for watermark embedding. The robustness is addressed by the fact that the metrics which are used in the segmentation stage result in a geometric representation by ellipsoids that cannot be heavily altered after commonly referenced image processing operations. We choose to construct a watermark based on a chaotic trajectory [23], because of its controlled lowpass properties. This cannot be accomplished using a usual pseudo-random sequence, because this type of sequence produces noisy-like binary watermarks that would very easily be distorted by lowpass filtering or JPEG compression. The chaotic sequence we use contains a parameter that controls how fine the details of the produced watermark are. It is also cryptographically more secure than a pseudo-random one, because it is not invertible and the original watermark cannot be reconstructed without knowing the appropriate key. The first step to construct such a watermark, is to produce a sequence of real numbers by using a mapping function $\mathbf{F}$: $U \rightarrow U$, $U \subset \mathbb{R}$ of the form

$$z(n+1) = \mathbf{F}(z(n), \lambda), \qquad z(n) \in U, \quad \lambda \in \mathbb{R} \tag{2}$$

where $\mathbf{F}$ is the Renyi map [14], $n = 0, 1, 2, \ldots$ is the current iteration, and λ is the parameter that controls the chaotic behavior of the system.

The number of iterations is arbitrary and can be adapted to our needs. The system theoretically produces trajectories of an infinite period. The decision on whether the trajectory presents regular or chaotic behavior depends on the seed value $z(0)$. The values of the produced trajectory oscillate inside an interval $[z_{\min}, z_{\max}] \subset U$ that is related to λ. Thus, we can define a threshold level $z_{th} \in [z_{\min}, z_{\max}]$ in a way that, after thresholding the sequence numbers, a bipolar sequence $s(n) \in \{-1, 1\}$ is produced with approximately equal number of -1 s and 1 s. Parameter λ controls the frequency characteristics of the chaotic sequence, i.e., the frequency of the transitions $-1 \rightarrow 1$ and $1 \rightarrow -1$. For $\lambda > 1$ and values close to 1, we get a chaotic watermark with low number of transitions and, thus, lowpass properties, whereas when $\lambda \simeq 2$ the transitions are very frequent, the lowpass properties degrade and the sequence reduces to a pseudorandom one.

However, the sequence we produced so far is 1-D. To embed it in a digital image, we need to scan the image across the sequence in such a way that the lowpass properties are preserved. The classic raster scan is not proper for this task because the number of transitions is not under control in the vertical dimension. To avoid this, we use Peano scan order which has the property that every point along the scan is topologically closer to the previous and subsequent pixels than in the case of the raster scan [24]. In addition, it is possible to use cellular smoothing to eliminate spontaneous transitions that emerge after the Peano scan [23]. By using this technique, the output watermark has local neighborhoods of 1 s (or -1 s) that are more compact. The main disadvantage of the Peano scan is that it only produces square $2^n \times 2^n$ watermarks. Such a watermark can be anisotropically scaled to nonsquare bounding rectangles.

In order to construct different watermarks we use a key K that produces the seed value $z(0)$ for the generation of a chaotic trajectory. Keys of slightly different values provide trajectories that have small cross-correlation due to the strongly chaotic system under consideration, resulting in watermarks that provide both better FAR (false alarm ratio) and FRR (false rejection ratio) in the detection stage. This implies that the set $\mathbf{K}$ of possible keys that can produce distinct watermarks is quite large. This reduces the possibility of the watermark being tampered and also ensures noninvertibility of the watermark, which is one of the demands addressed in Section I. Thus, the corresponding key cannot be extracted from the two-dimensional (2-D) watermark.

For the watermark embedding process we use the extracted region characteristics defined in the previous section to embed the produced watermark in a specific image area that will be easy to detect even after certain intentional or unintentional attacks. A prototype watermark serves as a reference pattern which can be adapted according to the dimensions, center, and orientation of the bounding rectangle of each selected region before embedding. When the new region parameters are computed in the detection stage, each potential prototype watermark that is tested for presence in the watermarked and possibly manipulated image, is again adapted to these parameters before applying the detector. The prototype watermark is of size $2^n \times 2^n$ when Peano scan is used.

Before superimposing the watermark on the original image, a visual masking stage is introduced. For this purpose, the variance is computed for every point of the original image $f(x, y)$ over a proper neighborhood of size $N \times N$

$$\mathrm{Var}(x, y) = \frac{1}{N^2 - 1} \sum_{i=-(N-1)/2}^{(N-1)/2} \sum_{j=-(N-1)/2}^{(N-1)/2} \cdot (f(x+i, y+j) - \mu(x, y))^2 \tag{3}$$

where $\mu(x, y)$ is the mean value over the same neighborhood. The local variance is then normalized according to its maximum value, and is compared against a threshold T_{var}, which is a function of the watermark power h. If the variance exceeds this threshold, this means that the local neighborhood contains a large amount of texture or edge information, and the embedded watermark can be invisible. Otherwise, the region is considered to be homogeneous, having almost constant luminance, and is not suitable for watermark embedding. The dependency of the variance threshold on the watermark power is such that, if the watermark power is increased, the threshold will also increase nonlinearly, so that the watermark remains imperceptible. This means that a trade off should be made, so that the watermark is still both perceptually insignificant and recoverable to a reasonable degree. Several sophisticated techniques of perceptual watermarking that could alternatively be used are presented in [25], [26].

If w_0 is the prototype watermark, then the scaled and rotated watermark w_n of size $K_1 \times K_2$ is embedded to the region A_{rot}. The watermarked image $f_w(x, y)$ is defined as

$$f_w(x, y) = f(x, y) \qquad (x, y) \notin A_{emb} \vee ((x, y) \in A_{emb} \wedge \mathrm{Var}(x, y) \le T_{\mathrm{var}}) \tag{4}$$

$$f_w(x, y) = f(x, y) + h \cdot w_n(x, y) \qquad (x, y) \in A_{emb} \wedge \mathrm{Var}(x, y) > T_{\mathrm{var}} \tag{5}$$

where A_{emb} is the embedding image area. Alternatively, h can become a function of the local variance:

$$h(x, y) = h_{\max} \cdot s(\mathrm{Var}(x, y)) \tag{6}$$

where $s(\,)$ takes values in the range $[0, 1]$. s is chosen to increase monotonically with the variance. In our case, the watermark is embedded in the spatial domain and, thus, the watermark power is quantized to two integer values, 0 and $h_{\max}$, depending on the variance value. The masking principle is, in fact, useful when the watermark power is $h_{\max} \ge 3$. Otherwise the watermark is hardly visible even when embedded on the entire image region.

V. Watermark Detection

When a prototype watermark is to be detected inside a watermarked and possibly manipulated image, the image has to be first segmented, so that the salient features of the approximated regions are derived, as was explained in Section III. These features include the center coordinates, dimensions and orientation of the bounding rectangle of each approximated region. A prototype watermark of standard dimensions is constructed. Afterwards, this watermark is adapted to each embedding region by scaling, centering, and rotating it according to the bounding rectangle features. For each detection region $A_{\mathrm{det}_i}, i = 1, \ldots, M$, where M is the number of selected regions, the response of a hypothesis testing detector is computed

$$R(\hat{f}_w, \hat{w}_i) = \overline{\mathbf{a}}_i - \overline{\mathbf{b}}_i \tag{7}$$

where

$$\overline{\mathbf{a}}_i = \frac{1}{N_{\mathbf{A_i}}} \sum_{(x, y) \in \mathbf{A}_i} \hat{f}_w(x, y)$$

$$\overline{\mathbf{b}}_i = \frac{1}{N_{\mathbf{B}_i}} \sum_{(x, y) \in \mathbf{B}_i} \hat{f}_w(x, y) \tag{8}$$

with $\mathbf{A}_i = \{(x, y) \in A_{\mathrm{det}_i} | \hat{w}_i(x, y) = 1\}$ and $\mathbf{B}_i = \{(x, y) \in A_{\mathrm{det}_i} | \hat{w}_i(x, y) = -1\}$. $N_{\mathbf{A_i}}$ and $N_{\mathbf{B_i}}$ are the number of pixels of the sets $\mathbf{A}_i$ and $\mathbf{B}_i$, respectively. Thus, the

TABLE I
DETECTOR RATES FOR SEVERAL ATTACKS ON VARIOUS IMAGES

Attack	Image					
	Lena		Peppers		Southwell	
	FAR	FRR	FAR	FRR	FAR	FRR
no attack	$5.9352{\cdot}10^{-5}$	$1.2803{\cdot}10^{-4}$	$1.1506{\cdot}10^{-7}$	$1.4214{\cdot}10^{-13}$	$4.2305{\cdot}10^{-6}$	$7.0858{\cdot}10^{-4}$
median filter	0.0028	0.2412	$7.2713{\cdot}10^{-4}$	0.0736	0.0574	0.0793
moving average filter	0.0019	0.4556	$6.12{\cdot}10^{-4}$	0.1155	0.0528	0.0972
multiplicative Gaussian noise	0.004	$4.514{\cdot}10^{-4}$	$9.6846{\cdot}10^{-4}$	$8.5925{\cdot}10^{-8}$	0.0623	0.0016
JPEG compression	0.0035	0.0428	$9.0503{\cdot}10^{-4}$	0.0028	0.0581	0.0352
scaling	0.0469	0.0166	0.0635	$8.2553{\cdot}10^{-5}$	0.2054	0.0049
rotation	0.0029	0.0284	$9.2495{\cdot}10^{-4}$	$7.9701{\cdot}10^{-4}$	0.0465	0.017
cropping	0.0037	$4.9688{\cdot}10^{-4}$	$5.907{\cdot}10^{-4}$	$3.132{\cdot}10^{-8}$	0.0592	0.0015

detector expresses the difference $\overline{\mathbf{r}}_i$ of two sample means. The mean value and variance of the detector output are

$$\eta_{\overline{r}_i} = \eta_{\overline{a}_i} - \eta_{\overline{b}_i} \qquad \sigma^2_{\overline{r}_i} = \left(\frac{1}{N_{\mathbf{A}_i}} + \frac{1}{N_{\mathbf{B}_i}}\right)\sigma^2_{\hat{f}_w}. \quad (9)$$

In the case that the watermark is embedded on the entire embedding region, the detector output is assumed to follow a normal distribution. If the correct watermark is embedded on the image, then the mean value is $\eta_{\overline{r}_i} = 2h$ and the variance is $\sigma^2_{\overline{r}_i} = ((1/N_{\mathbf{A}_i}) + (1/N_{\mathbf{B}_i}))(\sigma^2_f + \sigma^2_{w_i})$, where σ^2_f is the variance of the initial image and $\sigma^2_{w_i}$ is the variance of the watermark, as is adapted for the certain region. Otherwise, if there is no watermark present, the mean value of the detector is $\eta_{\overline{r}_i} = 0$ and the variance is $\sigma^2_{\overline{r}_i} = ((1/N_{\mathbf{A}_i}) + (1/N_{\mathbf{B}_i}))\sigma^2_f$, which is not significantly different than in the case the watermark is present, because the factor $(1/N_{\mathbf{A}}) + (1/N_{\mathbf{B}})$ is very small and $\sigma^2_{w_i} \ll \sigma^2_f$. The detection is done over all regions where the watermark was embedded, and the overall detector output is defined as the maximal detector output for all watermarked image regions. This is expressed by

$$R = \max_{1 \le i \le M} R(\hat{f}_w, \hat{w}_i). \quad (10)$$

The detector output (10) must be compared against a proper threshold R_{thr} that will inform us with a satisfying certainty about the presence or the absence of the watermark. The distribution of the resulting output is not anymore normal, both in the case that no watermark is detected and in the case the correct watermark is detected. The expected mean values are now greater than 0 and $2h$, respectively. However, when searching for an efficient detection threshold, we will consider the approximating distributions as normal, for simplicity reasons. Alternatively, the median or mean value of the regional detection outputs can be used. It should be noticed that after watermark embedding and possible manipulation, the regions parameters may have slightly changed. The error in rectangle height, width and center coordinates estimation has been observed to be 3 pixels in the average in the tested images of size either 800×800 or 512×512. The error in rectangle orientation estimation is 0.02 radians in the average. Because the watermark is very sensitive to geometric operations, a local search for each parameter is necessary to lead us to the exact parameter values that were defined on the original image prior to embedding. The total CPU time for detecting a certain watermark at a certain position is about 1.4 s for a 512×512 image on a Silicon Graphics O_2 workstation with a R10000 processor and 256 Mbytes of main memory. Thus, a typical time for detection including local search is about 4–5 min. This is longer than methods that do not handle geometric distortions, but certainly faster than resorting to exhaustive search. The detection output in the result diagrams, is always calculated for the correct dimensions, center and orientation of the watermark, and for the region that provides the maximum output, for each key, among all selected regions. The results are also normalized by dividing the detector output by $2h$. As expected due to masking, as well as due to the fact that the distribution is not normal anymore, after normalization the output is different than 1.

In order to decide for an efficient detection threshold indicating watermark existence for any considered attack case, we follow an experimental approach. One-hundred watermarks are embedded and detected after several attacks on both the original and the respective watermarked images. Both of the experimental distributions, for every image and every attack, are approximated by normal distributions. The average value of the

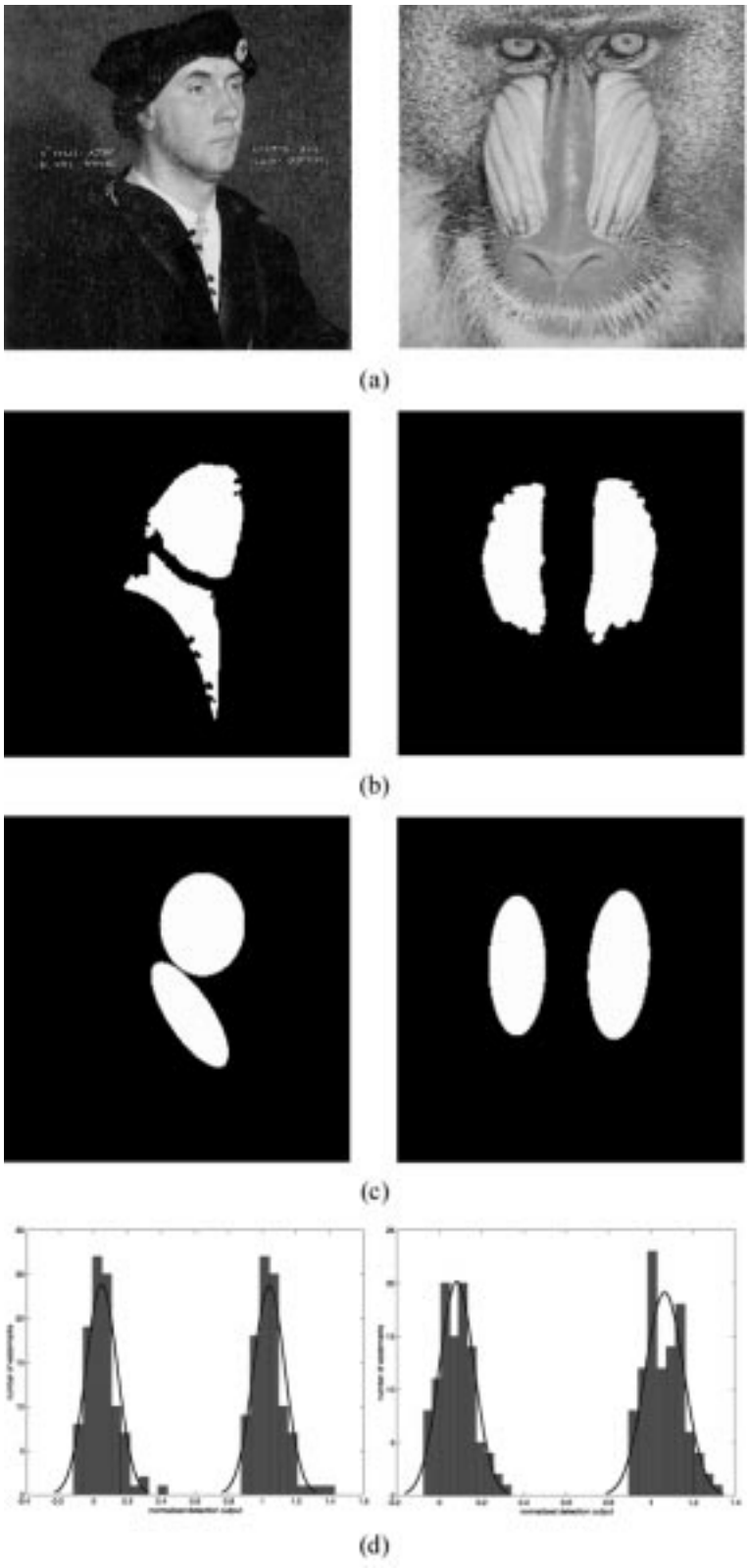

Fig. 2. (a) Watermarked images. (b) Largest regions of the segmented images. (c) Ellipsoid approximation of the regions in (b). (d) Experimental distributions of the normalized detector output.

means of each pair of distributions is considered a good detection threshold for the certain attack on the certain image. However, in order for the threshold to be applicable for many attacks and images, the mean value of the acceptable thresholds is considered as the common threshold for watermark detection. This threshold can be computed after performing the attacks for a training set of images. This was the approach that was followed in our experiments.

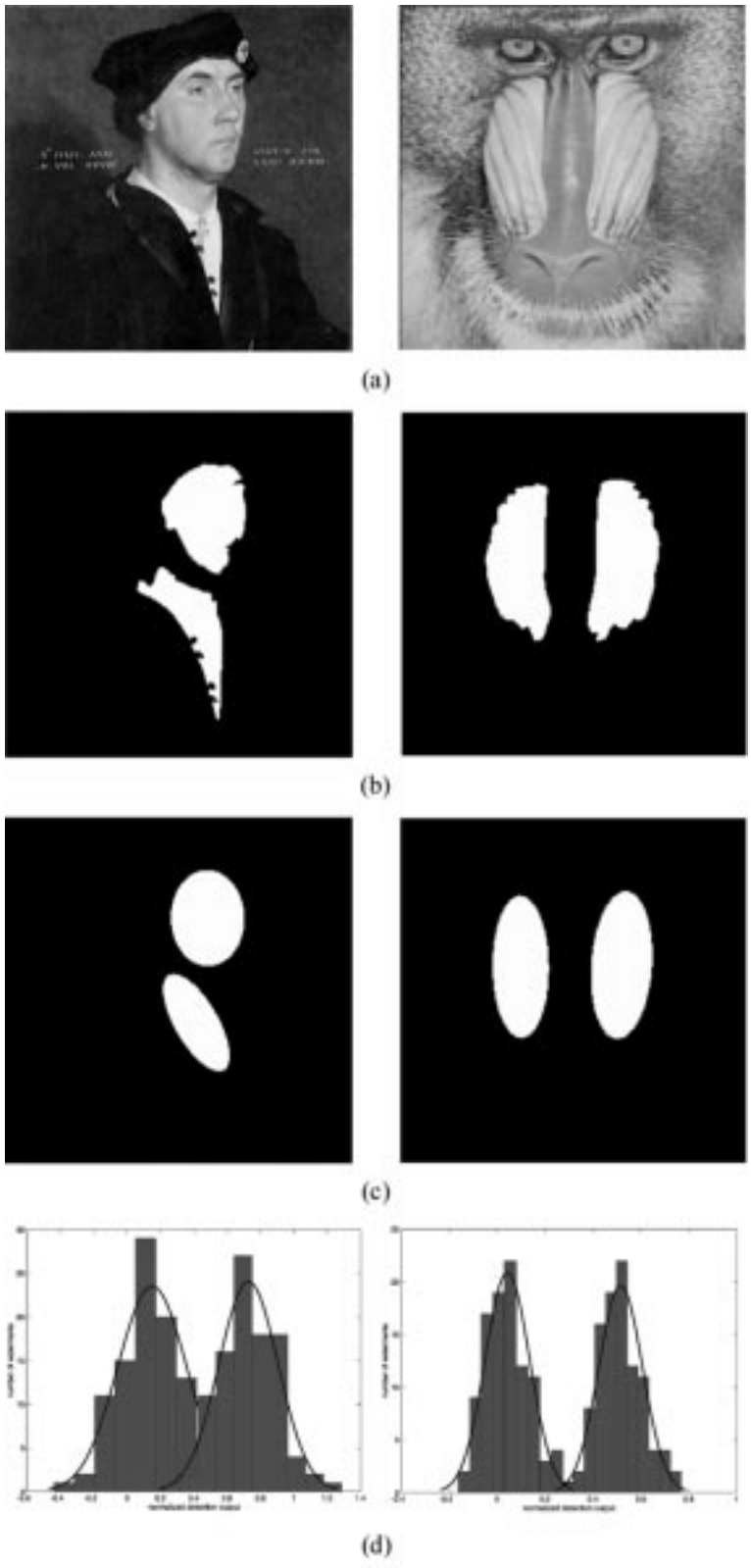

Fig. 3. (a) Median 3×3 filtered watermarked images. (b) Largest regions of the segmented images. (c) Ellipsoid approximation of the regions in (b). (d) Experimental distributions of the normalized detector output.

From a different viewpoint, a multimodal fusion technique for detection could be employed, where the image regions would correspond to the various detection "experts" that can be employed in order to decide about the presence of a watermark. Different weighting factors can be assigned to the several experts according to their significance in the detection process, which is decided after the appropriate training. Properties such as region compactness, size and distance from the image bound-

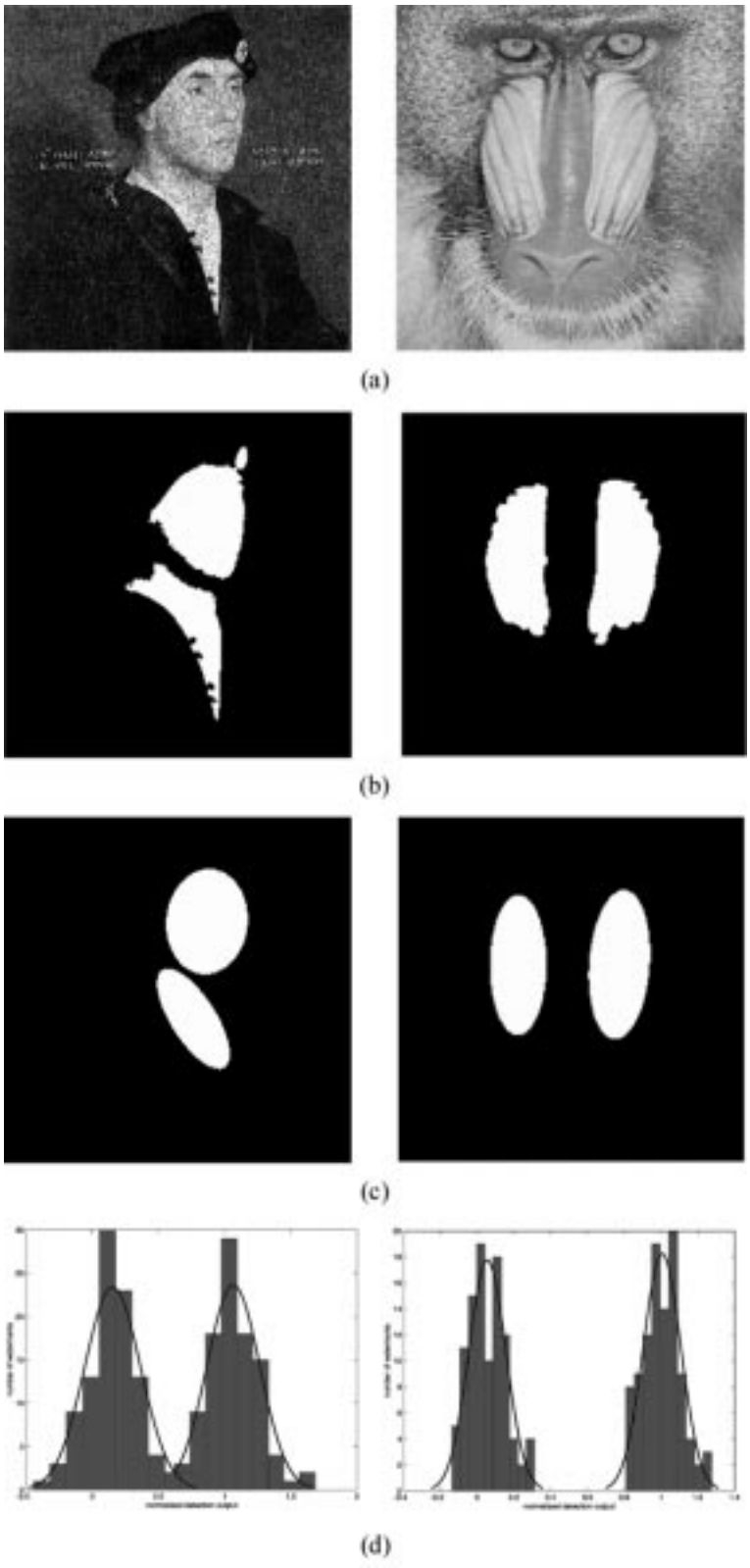

Fig. 4. (a) Multiplicative Gaussian noise on watermarked images. (b) Largest regions of the segmented images. (c) Ellipsoid approximation of the regions in (b). (d) Experimental distributions of the normalized detector output.

aries can be used to decide on the significance factors of the several regions. This helps to compensate for cases when some regions have been seriously distorted while others are not significantly affected.

We choose not to use masking in the detection stage, because the local variance may have changed significantly due to manipulations. The response is thus computed over the entire expected area of the embedded watermark.

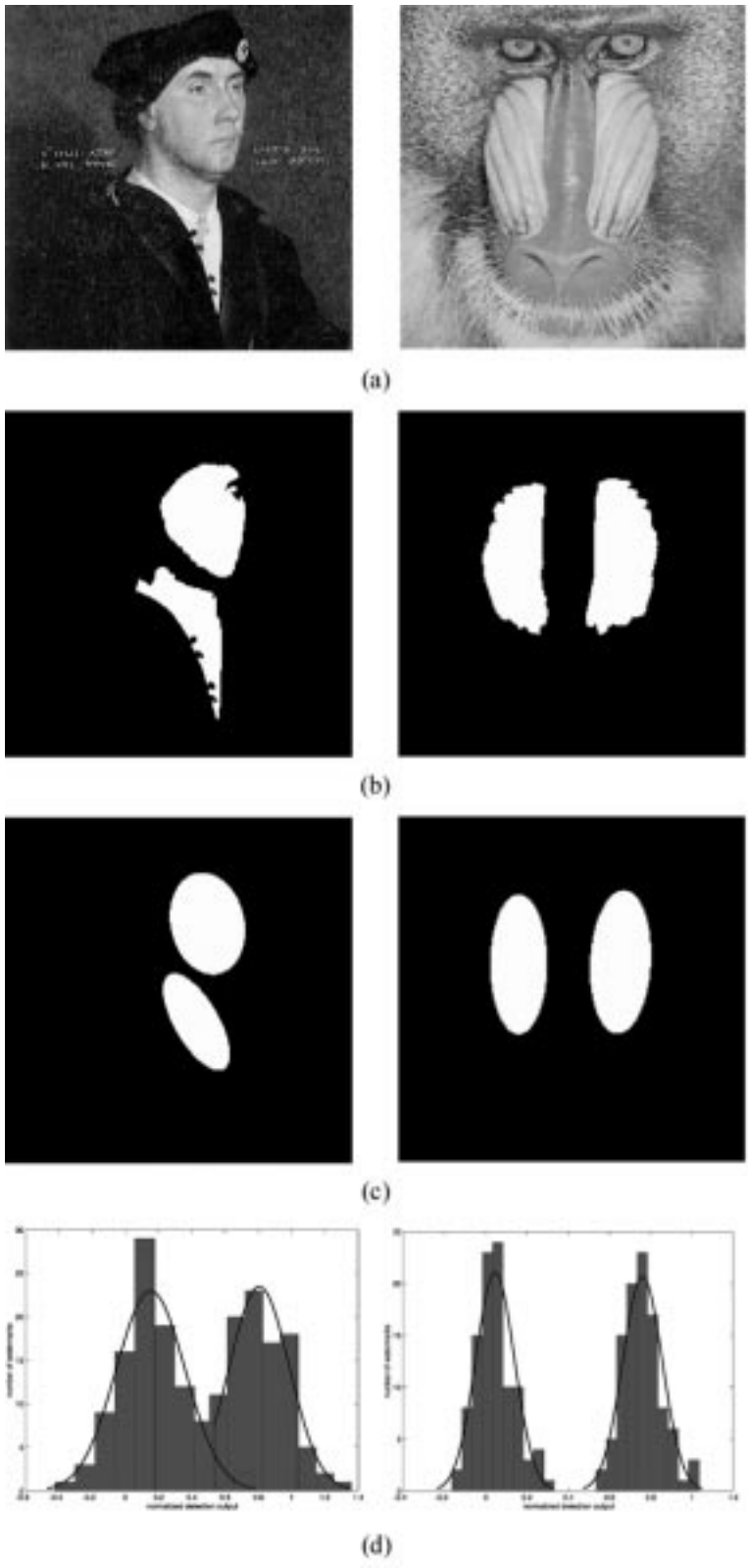

Fig. 5. (a) JPEG compressed watermarked images (quality 60%). (b) Largest regions of the segmented images. (c) Ellipsoid approximation of the regions in (b). (d) Experimental distributions of the normalized detector output.

VI. Experimental Results

We tested the robustness of our approach by applying several processing attacks on numerous images. We present some results on four of them, namely, "Lena," "Baboon," "Peppers" (all of size 512×512), and "Southwell," which is a 800×800 part of a painting by Hans Holbein the Younger.

The detection threshold was decided by considering several attacks on two images acting as a training set, namely, "Lena" and "Peppers." The mean of the thresholds found for each attack

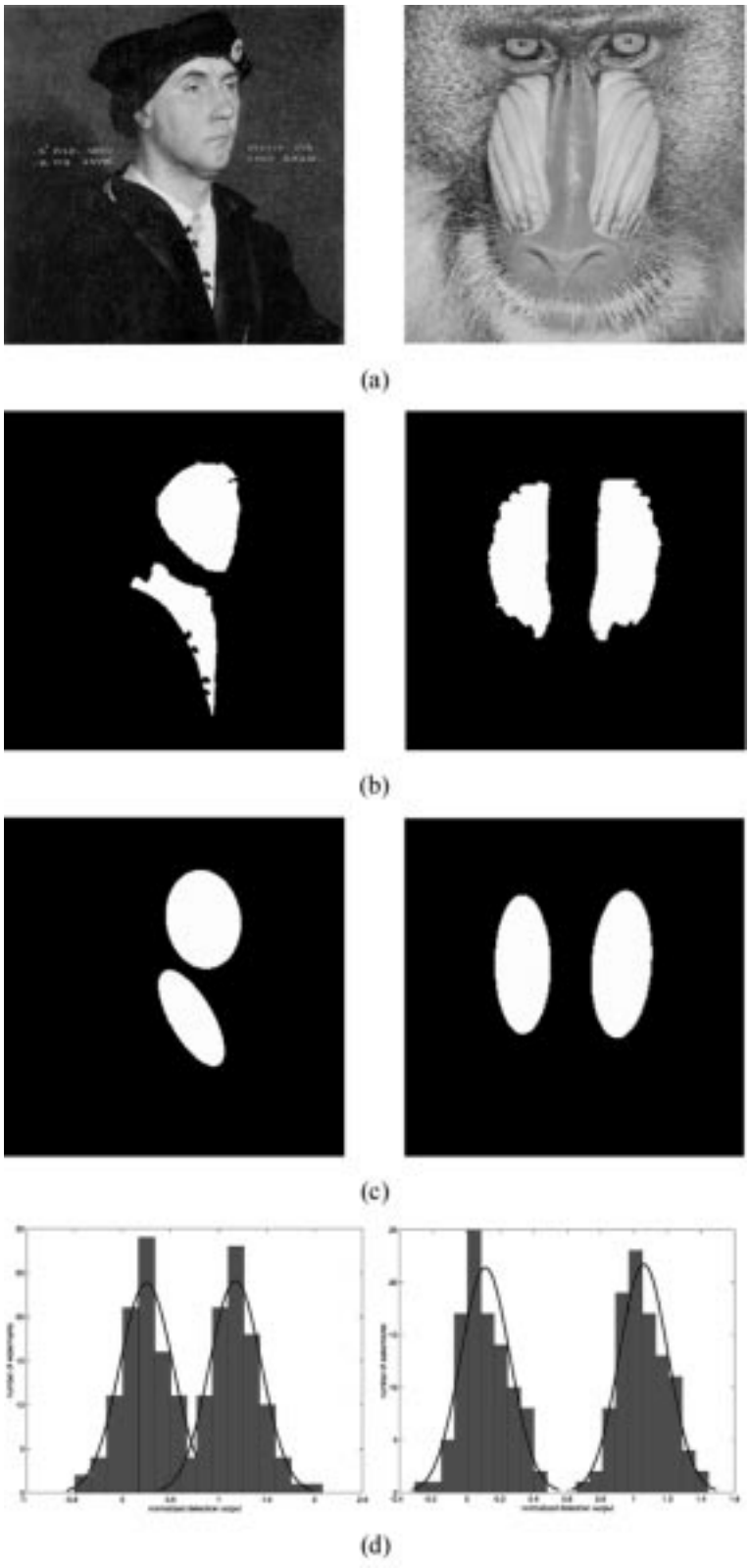

Fig. 6. (a) Scaled watermarked images. (b) Largest regions of the segmented images. (c) Ellipsoid approximation of the regions in (b). (d) Experimental distributions of the normalized detector output.

on each of these images, was defined as the common threshold. Detection was performed over two or three regions for each image. Table I shows the false acceptance ratio (FAR) and the false rejection ratio (FRR) for several attacks on the training set of images, as well as on the "Southwell" image, considering a common threshold of 0.4871. The size of the prototype watermark is 128×128, the watermark power is $h = 3$ and the chaotic parameter is $\lambda = 1.8$. The normalized threshold vari-

Fig. 7. (a) Rotated watermarked images. (b) Largest regions of the segmented images. (c) Ellipsoid approximation of the regions in (b). (d) Experimental distributions of the normalized detector output.

ance for masking was chosen $T_{\text{var}} = 0.002$. Peano scan as well as subsequent cellular smoothing were employed before embedding the watermark in order to retain its lowpass properties. The fact that some of the embedding regions are rather small, results in somehow increased values for FAR or FRR after some manipulation. An attempt to assess the minimum size of input data set that ensures a certain probability of false alarm during detection can be found in [27].

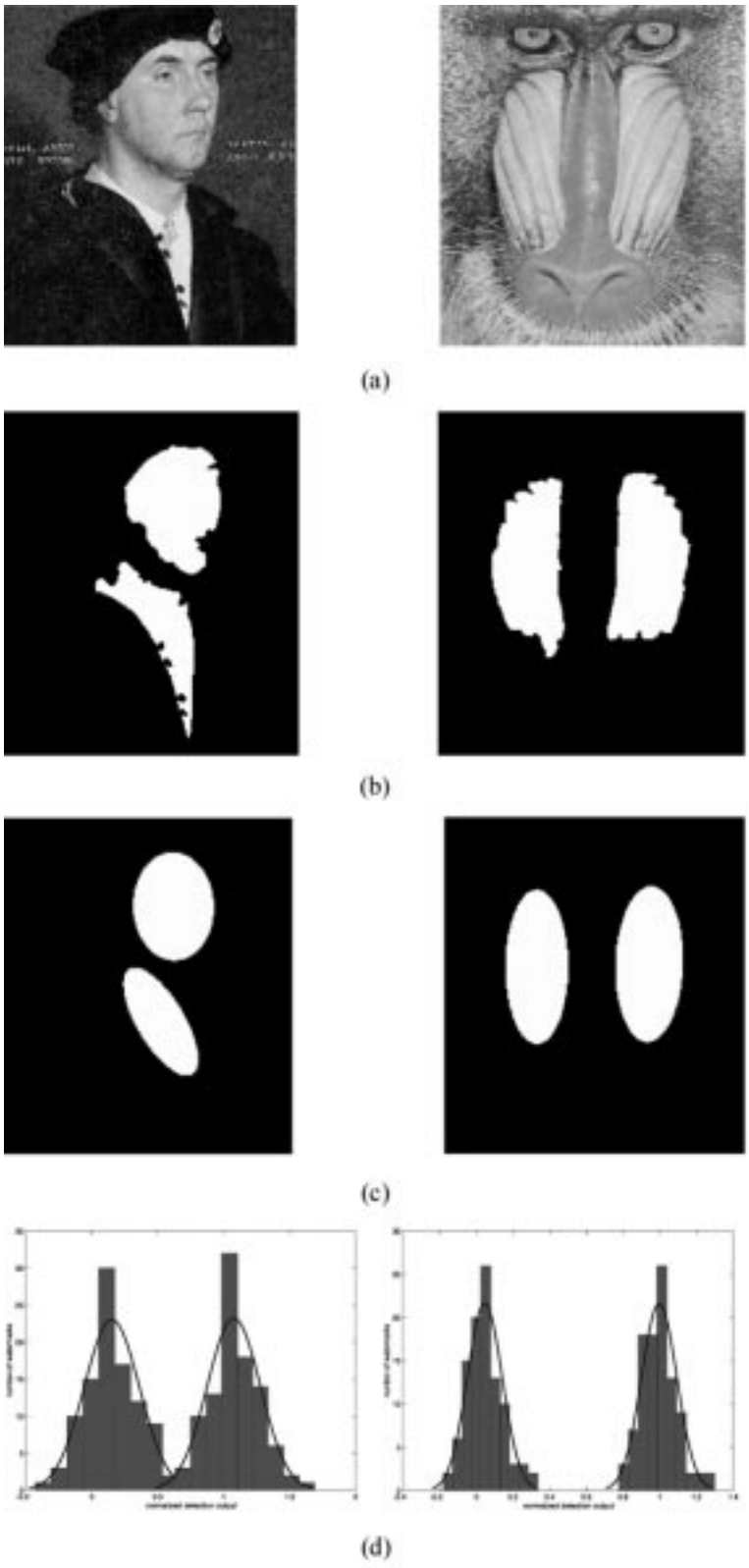

Fig. 8. (a) Cropped watermarked images. (b) Largest regions of the segmented images. (c) Ellipsoid approximation of the regions in (b). (d) Experimental distributions of the normalized detector output.

Figs. 2–8 illustrate sample results of the region approximation and watermark detection procedure on two of the images under concern, "Southwell" and "Baboon." The PSNR level considering the total watermarked area was 41.18 dB for the first image and 40.39 dB for the second one, stating that the distortion imposed by the watermark was hardly noticeable. The attacks studied were 3×3 median filter, multiplicative Gaussian noise having standard deviation $\sigma = 0.3$, JPEG compression

of quality 60%, scaling by a factor 1.25 for both dimensions, rotation by 12° and, finally, asymmetric cropping to a size of 600 × 700 for "Southwell" and to 400 × 450 for "Baboon." In Figs. 2(a)–8(a) the final images after being watermarked and attacked are shown. In Figs. 2(b)–8(b) the two largest regions for each image are shown. Figs. 2(c)–8(c) show their respective ellipsoid approximations. Finally, Figs. 2(d)–8(d) show the experimental distributions for detecting 100 different watermarks on the original image undergone the attack, and on the correctly watermarked image undergone the same attack. The vertical axis shows the number of watermarks that give a certain detector output, and the horizontal axis shows the detector output values. Similar results were obtained for the other images under test. The results denote the fact that though the geometric handling of the prototype watermark ensured the robustness of the method to certain geometric manipulations, notably rotation, scaling, cropping, and translation, the watermark proved to be robust to other attacks like compression and filtering. However, nonaffine geometric distortions, like the one introduced by Stirmark, could not be coped efficiently, since they distort the image regions and their parameters. Thus, the correct position of the watermark is impossible to recover after such an attack.

VII. Conclusions and Future Work

In this paper, we developed a method for embedding and detecting chaotic watermarks in large images. An adaptive clustering technique is employed in order to derive a robust region representation of the original image. The robust regions are approximated by ellipsoids, whose bounding rectangles are chosen as the embedding area for the watermark. The prototype watermark used for embedding is chosen to be a chaotic one, modified in such a way as to retain certain lowpass properties. The watermark is geometrically adapted before embedding, using the orientation, center coordinates and dimensions of the bounding rectangle. A hypothesis testing detector is employed in order to decide about the presence of a potential watermark. A visual masking technique is added in order to avoid annoying artifacts imposed by the embedded watermark. Experimental results display the robustness of the method for a variety of images. Future directions of the current work include development of more robust techniques for salient feature extraction, improvement of the watermark detection stage performance, as well as examination of alternative chaotic generators that may perform better than the one employed in this work.

References

[1] G. Voyatzis and I. Pitas, "Protecting digital-image copyrights: A framework," *IEEE Comput. Graph. Applicat.*, vol. 19, no. 1, pp. 18–24.

[2] S. Craver, N. Memon, B. L. Yeo, and M. M. Yeung, "On the invertibility of invisible watermarking techniques," in *Proc. IEEE Int. Conf. Image Processing (ICIP'97)*, vol. 1, Santa Barbara, CA, Oct. 1997, pp. 540–543.

[3] N. Nikolaidis and I. Pitas, "Copyright protection of images using robust digital signatures," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP '96)*, vol. 4, Atlanta, GA, May 1996, pp. 2168–2171.

[4] X.-G. Xia, C. G. Boncelet, and G. R. Arc, "A multiresolution watermark for digital images," in *Proc. ICIP '97*, vol. 1, Santa Barbara, CA, Oct. 1997, pp. 548–551.

[5] A. Piva, M. Barni, F. Bartolini, and V. Capellini, "DCT-based watermark recovering without resorting to the uncorrupted original image," in *Proc. IEEE Int. Conf. Image Processing (ICIP'97)*, vol. 1, Santa Barbara, CA, Oct. 1997, pp. 520–523.

[6] J. O'Ruanaidh and T. Pun, "Rotation, scale and translation invariant digital image watermarking," in *Proc. ICIP '97*, vol. 1, Santa Barbara, CA, Oct. 1997, pp. 536–539.

[7] I. J. Cox, J. Killian, T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Trans. Image Processing*, vol. 6, pp. 1673–1687, Dec. 1997.

[8] M. Kutter and F. A. P. Petitcolas, "A fair benchmark for image watermarking systems," in *Proc. Electronic Imaging '99 Security Watermarking Multimedia Contents*, vol. 3657, San Jose, CA, Jan. 1999, pp. 226–239.

[9] M. J. J. J. B. Maes and C. W. A. M. van Overveld, "Digital watermarking by geometric warping," in *Proc. ICIP '98*, vol. 2, Chicago, IL, Oct. 1998, pp. 424–426.

[10] S. Pereira, J. J. K. Ó Ruanaidh, F. Deguillaume, G. Csurka, and T. Pun, "Template based recovery of Fourier-based watermarks using log–polar and log–log maps," in *Proc. IEEE Int. Conf. Multimedia Computing and Systems (ICMCS'99)*, vol. 1, Florence, Italy, June 1999, pp. 870–874.

[11] F. Hartung, J. K. Su, and B. Girod, "Spread spectrum watermarking: Malicious attacks and counter-attacks," in *Proc. Electronic Imaging '99: Security and Watermarking of Multimedia Contents*, vol. 3657, San Jose, CA, Jan. 1999, pp. 147–158.

[12] M. Kutter, F. Jordan, and F. Bossen, "Digital watermarking of color images using amplitude modulation," *J. Electron. Imag.*, vol. 7, no. 2, pp. 326–332, 1998.

[13] T. N. Pappas, "An adaptive clustering algorithm for image segmentation," *IEEE Trans. Signal Processing*, vol. 40, pp. 901–914, Apr. 1992.

[14] R. L. Devaney, *An Introduction to Dynamical Systems*. New York: Penjamine/Cummings, 1986.

[15] L. Vincent and P. Soille, "Watersheds in digital spaces: An efficient algorithm based on immersion simulations," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 13, pp. 583–598, June 1991.

[16] L. Vincent, "Morphological grayscale reconstruction in image analysis: Applications and efficient algorithms," *IEEE Trans. Image Processing*, vol. 2, pp. 176–201, Feb. 1993.

[17] A. S. Wright and S. T. Acton, "Watershed pyramids for edge detection," in *Proc. IEEE ICIP '97*, Santa Barbara, CA, Oct. 1997, pp. 578–581.

[18] D. Cortez, P. Nunes, M. M. de Sequeira, and F. Pereira, "Image segmentation toward new image representation models," *Signal Process.: Image Commun.*, vol. 6, no. 6, pp. 485–498, 1995.

[19] J. Besag, "On the statistical analysis of dirty pictures," *J. R. Statist. Soc. B*, vol. 48, no. 3, pp. 259–302, 1986.

[20] J. Luo, R. T. Gray, and H.-C. Lee, "Incorporation of derivative priors in adaptive Bayesian color image segmentation," in *Proc. ICIP '98*, vol. 3, Chicago, IL, Oct. 1998, pp. 780–784.

[21] A. G. Bors and I. Pitas, "Object segmentation in 3-D images based on alpha-trimmed mean radial basis function network," in *Proc. EUSIPCO '98*, vol. 2, Rhodes, Greece, Sept. 1998, pp. 1093–1096.

[22] A. K. Jain, *Fundamentals of Digital Image Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[23] G. Voyatzis and I. Pitas, "Chaotic watermarks for embedding in the spatial digital image domain," in *Proc. ICIP '98*, vol. 2, Chicago, IL, Oct. 1998, pp. 432–436.

[24] C. Gotsman and M. Lindenbaum, "On the metric properties of discrete space filling curves," in *Proc. Int. Conf. Pattern Recognition*, vol. 3, Jerusalem, Israel, Oct. 1994, pp. 98–102.

[25] R. B. Wolfgang, C. I. Podilchuk, and E. J. Delp, "Perceptual watermarks for digital images and video," *Proc. IEEE*, vol. 87, pp. 1108–1126, July 1999.

[26] M. D. Swanson, B. Zhu, and A. H. Tewfik, "Transparent robust image watermarking," in *Proc. 1996 IEEE Int. Conf. Image Processing*, vol. 3, Lausanne, Switzerland, Sept. 1996, pp. 211–214.

[27] A. Tefas, A. Nikolaidis, N. Nikolaidis, V. Solachidis, S. Tsekeridou, and I. Pitas, "Performance analysis of watermarking schemes based on skew tent chaotic sequences," in *Proc. IEEE-EURASIP Workshop Nonlinear Signal Image Processing (NSIP 2001)*, Baltimore, MD, June 2001.

Athanasios Nikolaidis was born in Serres, Greece, in 1973. He received the Diploma degree in computer engineering from the University of Patras, Patras, Greece, in 1996. He is currently a Research and Teaching Assistant and is pursuing the Ph.D. degree at the Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki, Greece.

His research interests include nonlinear image and signal processing and analysis, face detection and recognition and copyright protection of multimedia.

Mr. Nikolaidis is a member of the Technical Chamber of Greece.

Ioannis Pitas (S'83–M'84–SM'94) received the Diploma of Electrical Engineering in 1980 and the Ph.D. degree in electrical engineering in 1985 both from the University of Thessaloniki, Thessaloniki, Greece.

Since 1994, he has been a Professor at the Department of Informatics, University of Thessaloniki. From 1980 to 1993, he was Scientific Assistant, Lecturer, Assistant Professor, and Associate Professor in the Department of Electrical and Computer Engineering at the same University. He served as a Visiting Research Associate at the University of Toronto, Toronto, ON, Canada, University of Erlangen-Nuernberg, Germany, Tampere University of Technology, Tampere, Finland, and Visiting Assistant Professor at the University of Toronto. He was Lecturer in short courses for continuing education. His current interests are in the areas of digital image processing, multidimensional signal processing, and computer vision. He has published over 300 papers and contributed to eight books in his area of interest. He is the coauthor of the book *Nonlinear Digital Filters: Principles and Applications* (Norwell, MA: Kluwer, 1990) and author of *Digital Image Processing Algorithms* (Englewood Cliffs, NJ: Prentice Hall, 1993). He is the editor of the book *Parallel Algorithms and Architectures for Digital Image Processing, Computer Vision, and Neural Networks* (New York: Wiley, 1993). He is coeditor of *Multidimensional Systems and Signal Processing*.

Dr. Pitas has been member of the European Community ESPRIT Parallel Action Committee. He has also been an invited speaker and/or member of the program committee of several scientific conferences and workshops. He was Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS and is currently an Associate Editor of the IEEE TRANSACTIONS ON NEURAL NETWORKS. He was Chair of the 1995 IEEE Workshop on Nonlinear Signal and Image Processing (NSIP'95). He was Technical Chair of the 1998 European Signal Processing Conference. He is General Chair of IEEE International Conference on Image Processing 2001.