An Optimal Detector Structure for the Fourier Descriptors Domain Watermarking of 2D Vector Graphics

Victor Rodriguez Doncel, Nikos Nikolaidis, *Member, IEEE* and Ioannis Pitas, *Fellow, IEEE*

***Abstract*— Polygonal lines constitute a key graphical primitive in 2D vector graphics data. Thus, the ability to apply a digital watermark to such an entity would enable the watermarking of cartoons, drawings and Geographical Information Systems (GIS) data in vector graphics format. This paper builds on and extends an existing algorithm which achieves polygonal line watermarking by modifying the Fourier descriptors magnitude in an imperceptible way. Watermarks embedded by this technique can be detected in rotated, translated, scaled or reflected polygonal lines. The detection of such watermarks had been previously carried out through a correlator detector. In this paper, analysis of the statistics of the Fourier descriptors is exploited to devise an optimal blind detector. Furthermore, the problem of watermarking multiple lines as well as other implementation issues are being addressed. Experimental results verify the imperceptibility and robustness of the proposed method.**

***Index Terms*— watermarking, digital rights management, 2D vector graphics, Fourier descriptors**

I. Introduction

DIGITAL watermarking is a relatively new research topic that attracted the interest of numerous researchers both in the academia and the industry.

Watermarking is the practice of imperceptibly altering a piece of data in order to embed information about the data [1]. A watermarking system consists of two modules: The embedding module that inserts the information in the host data and the detection/decoding module that checks whether a given piece of data hosts a watermark and subsequently retrieves the conveyed information. Depending on whether the original data should be available or not during watermark detection, methods are characterized as non-blind or blind. Obviously, the availability of the original host signal makes non-blind detection much easier than blind detection but at the same time limits significantly the applicability of non-blind methods in real world setups. Thus, due to their wider scope of application, blind techniques received much more attention among researchers. With respect to the amount of information conveyed by the watermark, watermarking systems can be categorized into zero bit and multiple bit systems. Zero bit systems can only check whether the data under investigation host a watermark generated by a specific watermark key K, i.e., verify whether the data are watermarked or not. On the other hand, multiple bit systems are capable of encoding in the host data a message consisting of multiple bits.

According to the type, the amount and the properties of the embedded information, watermarking can serve a wide range of applications. Copyright protection (or more generally digital rights management) is perhaps the most important of these applications. In this case, the embedded data can carry information about the legal owner of a digital item and be used for warning a user that the item is copyrighted, for tracking down unauthorized copies of the item and for proving the ownership of the item when a legal dispute arises. Watermarks used for digital rights management should be of the robust type. Robustness of a watermarking method describes the degree of resistance of the method to modifications of the host signal due to either common signal processing operations or manipulations that are devised specifically in order to make the watermark undetectable. The latter are usually called attacks. A general framework for digital watermarking can be found in [2], [3], while [1] provides an extensive overview of the watermarking principles and techniques.

Digital watermarking has been mainly applied to still image, audio and video data, while other types of data have received far less attention. Indeed, little work has been done in watermarking of 2D vector graphics data, that are typically used in Geographical Information Systems (GIS), in Computer Aided Design (CAD) or cartoons and drawings in vector format. This paper presents a method for the copyright protection of 2D vector graphics data through blind, zero bit watermarking of 2D polygonal lines which constitute a common graphics primitive in such data. Essentially, the paper extends the method proposed in [4], [5] by devising an optimal blind detector for this scheme, taking into account the Fourier descriptors statistics. The embedding procedure is kept the same, i.e., watermarking is achieved by imperceptibly modifying the coordinates of the vertices that define a polygonal line through the modification of the magnitude of the Fourier descriptors of the line. However, the proposed optimal detector substantially increases the efficiency of the method in [4]. Furthermore, the problem of watermarking multiple lines in the same vector graphics file and reaching a global decision on whether this set of lines is watermarked or not, as well as other important implementation issues are being addressed. An earlier version of this work that was considerably shorter has been reported in [6]. Since the algorithm operates on vertices, it can be also utilized for the watermarking of 2D parametric curves (e.g. splines) by applying it on the control points of such curves. Thus, the algorithm can be used for watermarking 2D vector graphics data (i.e., data represented in one of the well-known SVG, EMF, CGM, DXF formats) containing either 2D polygonal lines or parametric curves. In all cases, a minimum number of vertices is needed for the polygonal line to be watermarked efficiently.

The structure of this paper is as follows. An review of techniques proposed so far for the watermarking of 2D vector graphics data as well as a short overview of the related field of 3D

Department of Informatics, Aristotle University of Thessaloniki, 54124 Thessaloniki, Greece Tel / Fax: +30 2310 996 304 (email: victor. rodriguezd @upf.edu, nikolaid@aiia.csd.auth.gr, pitas@aiia.csd.auth.gr)

mesh watermarking are presented in the next Section. A detailed description of the method developed in [4] is provided in Section III in order to make the paper self-contained. The new detection paradigm is introduced, with both theoretical and experimental arguments in Section IV. Several important implementation issues like multiple lines watermarking and watermarking of overlapping or adjacent shapes that share a certain number of points are discussed in Section V. Finally, experimental performance evaluation results are presented in Section VI, followed by conclusions.

II. Related Work

As already mentioned, only a few attempts towards watermarking of 2D vector graphics data, mainly 2D GIS data, can be found in the corresponding literature. [7] presents an overview of the problem of watermarking of GIS data and describes its technical and legal implications but provides no review of applicable techniques. In [4], [5] zero bit watermarking of polygonal lines in vector graphics files is achieved by modifying in a multiplicative way the magnitude of their middle frequency Fourier descriptors. Blind detection is performed by evaluating the correlation of the watermark with the sequence containing the magnitudes of the Fourier descriptors of the polygonal line under examination. Due to the properties of the Fourier descriptors the method is robust to translation, rotation, scaling, mirroring, change of the traversal starting point and direction, addition of noise on the vertex coordinates and curve smoothing. A Java implementation of this algorithm capable of watermarking SVG files and taking into account all geometric primitives in such files has been presented in [8]. No details on the way used to watermark all primitives and subsequently combine the detection results derived from each primitive are provided. The authors state that the method is robust to rotation translation scaling and mirroring. The authors of [9] embed a zero bit watermark in GIS data by modifying the s least significant decimal digits of the vertex coordinates. The modification is achieved by adding to these digits a pseudo-random noise sequence while concurrently taking care not to cause alterations that are above the tolerance level (i.e. the maximum allowable error) specified for the data. Moreover, the modifications are performed in a way that ensures that even after an attack that would change the coordinate values up to a predefined extent no overflow or underflow would occur on the selected digits, namely no other digits would be altered. Blind watermark detection is performed by a correlation detector. The authors claim that the method is robust to attacks that result in changes of the coordinates within the tolerance levels of the data but provide no experimental results. The proposed method seems to have limited robustness to rotation or scaling and thus might not be applicable to generic vector graphics data but only to GIS data where no such attacks are usually encountered. In [10] the authors propose separating the area of a 2D vector map in non-overlapping blocks and moving the polygonal lines vertices in each block towards the upper or lower triangular region of the block, depending on the watermark bit that is to be embedded in this block. Thus, multiple-bit watermarking is achieved. Embedded bits can be blindly extracted by checking whether the vertices in a block are in the lower/upper triangular region. The authors claim that the method is robust to noise addition without providing any information about the performance of the method to other attacks. The magnitude of Discrete Wavelet Transform (DWT) coefficients of vertices extracted from a vector graphics file is used for embedding the watermark in the zero-bit, blind technique proposed in [11]. More specifically, a complex signal is constructed by considering the x and y coordinates of each vertex as the real and imaginary part of a complex number and a 3-level DWT is applied on this signal. The watermark is embedded by changing the magnitude of the DWT coefficients in the HH_2 and HH_3 sub-bands. The properties of DWT ensure that the watermark can withstand, up to a certain degree, rotation, translation, uniform scaling and noise addition. In [12] four blind algorithms that embed multiple bits of information in 2D vector graphics images by introducing visible or invisible distortions on the lines of such images are proposed. The two algorithms that cause visible distortions in a stylistic way perform bit embedding by changing the line attributes (color or width) and by replacing the lines with small and very closely spaced line segments that give the impression of a continuous line. The two algorithms that embed information in an invisible way do so by introducing new redundant vertices within line segments or by subdividing line segments so that the lengths of the segments produced by the subdivision encode the bits to be embedded. The authors state that the algorithms are suitable for applications where no security is required, i.e., applications where no intentional attacks on the data in order to remove the watermark are expected. Enhancing a graphics file with additional information that can withstand format conversions is such an application. Indeed, the proposed methods seem to be robust to format conversion, translation, scaling and rotation and can decode the information embedded on the part of a vector graphics image that remains intact after cropping. However, the watermarks can be easily removed by attacks designed specifically for each method. For example, an attacker can easily destroy a watermark embedded by the point insertion algorithm by inserting new points or destroy a watermark inserted by the algorithm that changes the line attributes by changing again these attributes.

In [13], a non-blind watermarking method which is capable of embedding multiple bits of information in geographical maps in vector format is proposed. The method splits the map into a number of non-overlapping rectangles using a modified quadtree subdivision method and embeds each bit of information by displacing the vertices lying in each rectangle. Each bit might be embedded multiple times for increased robustness. In order to detect the watermark and extract the embedded bits, the watermarked map is aligned with the original map through an iterative method involving landmark points thus inverting any affine transformations that might have been applied on this map. Afterwards, vertices that have been inserted or deleted from the watermarked map are identified and excluded from further processing. Finally, detection is performed by comparing, in each rectangle, the averaged vertex coordinates of the watermarked map with those of the original map. Experimental results show that the method is robust to translation, uniform scaling, change of the order of objects within the file, addition or removal of vertices, moderate additive noise and partially robust to cropping. However, the fact that the method is non-blind limits its applicability to cases where the original data of a map under examination can be identified and used during detection. An improved variant of the above technique is proposed in [14]. This variant treats vertices in a map as a point set and creates an associated 2D mesh by applying Delaunay triangulation. The mesh is subsequently split into a number of mesh patches using

TABLE I
CATEGORIZATION OF EXISTING 2D VECTOR GRAPHICS WATERMARKING TECHNIQUES.

Method	Conveyed Information	Detection
proposed method	zero bit	blind
[4], [5], [8]	zero bit	blind
[9]	zero bit	blind
[11]	zero bit	blind
[12]	multiple bits	blind
[10]	multiple bits	blind
[13]	multiple bits	non-blind
[14]	multiple bits	non-blind
[16]	multiple bits	non-blind
[17]	zero bit	non-blind

a set of non-overlapping rectangles and the patch within each rectangle is transformed to the "frequency" domain using the mesh spectral analysis technique proposed in [15]. Embedding is performed by modifying the value of the low frequency spectral coefficients. Similar to [13], detection is performed by comparing within each rectangle the values of the spectral coefficients of the watermarked and the original map. Detection is preceded by an alignment step that registers the watermarked map with the original one. The method achieves superior robustness when compared to [13], but, like its predecessor is a non-blind one. Non-blind, multiple-bit watermarking of 2D vector maps in the DFT domain was proposed in [16]. Similar to [5], each bit of information is embedded by modifying in an additive way the magnitude of a different Fourier descriptor of the curve. Only middle frequency coefficients are modified. Prior to detection, correspondences between the vertices of the watermarked map and those of the original map are found. Decoding of the embedded bits is performed by evaluating the Fourier descriptors and comparing for each descriptor the value of its magnitude in the watermarked and the original data. The method is robust to noise addition on the vertex coordinates, insertion and removal of vertices and vertex reordering but is fairly sensitive to cropping. Finally, in [17] a non-blind method for watermarking of vector maps and drawings that is based on B-spline modelling of the curves is proposed. The method fits B-spline curves on the polygonal lines to be watermarked and embeds information by altering in an additive way the coordinates of the control points of the resulting splines. Subsequently, the watermarked B-splines are sampled in order to generate watermarked polygonal lines. Before detection, the polygonal lines are approximated by B-splines and an iterative alignment algorithm is used to register the potentially watermarked spline with the original curve. Detection is performed by correlation between the original watermark signal and an estimate of the embedded watermark signal. The properties of the B-splines and fact that the method is a non-blind one grant it robustness against a number of attacks like cropping, geometric transformation, removal or addition of points, vector to raster and raster to vector conversion etc. Table I provides a categorization of the reviewed 2D vector data watermarking algorithms with respect to the information conveyed by the watermark (zero bit versus multiple bits) and the way detection is performed (blind versus non blind).

Watermarking of 2D vector graphics data is somewhat related to watermarking of 3D mesh models, although important differences, like the fact that no natural ordering (and thus no globally accepted traversal scheme) can be devised for points in the $3D$ space. The majority of watermarking methods for such models embed data by modifying the position of the vertices that form the mesh, although techniques that modify the connectivity of the mesh (without changing the vertex positions) [18], the texture images [19] or the texture coordinates [18] have been also proposed. A number of techniques that rely on vertex position modification for watermark embedding operate directly on the spatial domain and do so either by taking into account the vertex connectivity or by ignoring connectivity information. In [20], [21] embedding is performed by displacing the vertices so as to alter the distribution of triangle normals. Detection is performed in semi-blind way, since information regarding the original orientation of the model is required. In [22] the watermark key along with a traversal rule are used to establish a triangle strip over the mesh. Subsequently, each triangle of the strip is used to encode one bit of the message. To do so, one of the edges of each triangle is partitioned into a number of intervals, each encoding either zero or one, and the vertex lying opposite to this edge is moved so that its orthogonal projection on the edge resides in the appropriate interval. In [23], [24] embedding is performed by representing vertices in spherical coordinates r, θ, ϕ and modifying only the r coordinate whereas blind detection is achieved by employing a statistical approach. In [25] a number of vertices where distortions induced by the watermark are likely to be less visible are selected and ordered. Multiple bit watermark embedding is performed by displacing each of these vertices so that local constraints involving first and second order moment descriptions of their neighborhoods are enforced. Watermark detection is performed in a blind manner.

In addition, a number of 3D mesh watermarking algorithms that operate on a suitable transform domain have been proposed. The method proposed in [26] operates on semi-regular meshes and performs watermark embedding by applying 3D wavelet decomposition [27] and modifying the wavelet coefficients at a suitable resolution level. Watermark detection is performed in a blind fashion through a statistical detection approach. The authors of [28] partition the mesh into patches and apply to each of them a spectral decomposition approach [15], [29] that is based on the eigenvalue decomposition of a Laplacian matrix derived from the connectivity of the mesh. Multiple bit watermark embedding is performed by modifying the mid and high frequency spectral coefficients. In [30] the authors ignore connectivity information and treat the mesh as a series of vertices. The spectrum of this series is computed by applying Singular Spectrum Analysis (SSA) [31] and the resulting singular values are modified in order to encode the watermark. Non-blind watermark detection is achieved by comparing the singular values of the watermarked and the original data.

III. A WATERMARKING SYSTEM FOR POLYGONAL LINES

A. Watermark embedding

In the system proposed in [4] the polygonal line $\mathbf{v}$ is described as a series of N vertices, $v[n] = (v_x[n], v_y[n])$, that can be considered as a complex signal:

$$\mathbf{x} : x[n] = x_R[n] + ix_I[n] \ , \ n = 0, 1, ..., N-1 \qquad (1)$$

whose real and imaginary parts are the 2D vertex x and y coordinates:

$$x_R[n] = v_x[n], \quad x_I[n] = v_y[n] \qquad (2)$$

A complex DFT is applied on this signal, resulting in its *Fourier descriptors* $\mathbf{X} : X[k] = X_R[k] + iX_I[k],\ k = 0, 1, ..., N-1$:

$$X[k] = \sum_{n=0}^{N-1} x[n] \exp\left(\frac{-i2kn\pi}{N}\right), 0 \leq k \leq N-1 \qquad (3)$$

The Fourier descriptors representation of a polygonal line bears some important geometric invariance properties [32] that can be exploited in order to devise a robust watermarking method. In particular, the Fourier descriptors magnitude remains unaltered after some geometrical transformations of the polygonal line (cf. Section III-B).

Watermark embedding takes place by modifying the magnitude of the Fourier descriptors, according to the following multiplicative embedding rule:

$$|X'[k]| = |X[k]|(1 + sW[k]), k = 0, 1, ..., N-1 \qquad (4)$$

where $X'[k]$ represents the k-th Fourier descriptor of the watermarked polygon, the scalar $s, \quad 0 < s < 1$ controls the power of the watermark signal (which is related with the severity of modifications that will be induced by the watermark in the host signal), and $W[k]$ is a sample of the watermark signal. The watermark does not affect the phase of the Fourier descriptors. The watermark is a signal generated by a pseudorandom number generator using an integer as a seed. This seed is the watermark secret key. More specifically, samples $W[k]$ obtain equiprobably the values $+1$, -1, whereas samples with $W[k] = 0$ are used for low and high frequency Fourier descriptors. More specifically, the watermark is not embedded in the low frequencies to avoid causing severe distortions in the polygonal line. It is not embedded in the high frequencies either, so that it is robust to low-pass attacks, e.g. contour smoothing. Thus, the watermark $\mathbf{W} : W[k]\ k = 0, 1, ..., N-1$ has the form:

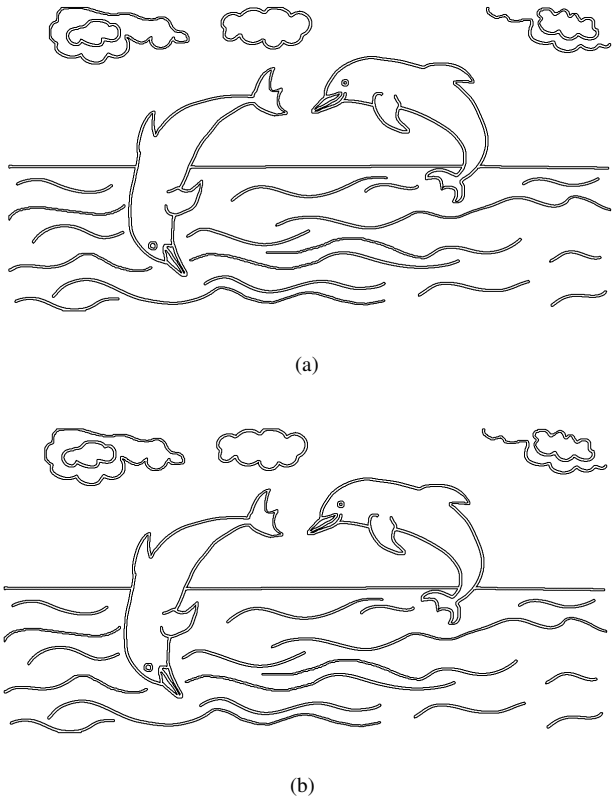
$$W[k] = \begin{cases} 0 & if\ k < aN\ or\ k > (1-a)N\ , \\ & or\ bN < k < (1-b)N \\ \pm 1 & else \end{cases} \qquad (5)$$

where $0 < a < b < 0.5$. The parameters a, b control the range of frequencies that will be affected by the watermark.

After watermark embedding through (4), the inverse DFT is applied in order to derive the watermarked polygonal line:

$$x'[n] = \frac{1}{N} \sum_{k=0}^{N-1} X'[k] \exp\left(\frac{i2kn\pi}{N}\right), 0 \leq n \leq N-1 \qquad (6)$$

An original SVG image and its watermarked counterpart can be seen in Fig. 1. Only the contours delineating the two dolphins have been watermarked. No differences between the two images can be perceived at this magnification. As a matter of fact, one can magnify this image several times before start seeing any difference.



(a)

(b)

Fig. 1. Comparison between an original SVG graphics file and its watermarked counterpart.

B. Robustness against host signal manipulations

As a direct consequence of the Fourier descriptors properties, the watermark is robust to a number of geometrical transformations [4], [32], as well as to their combinations. These transformations are:

- **Rotation.** Rotation of the polygonal line affects only the phase of the Fourier descriptors. Therefore, the watermark, which is applied on the magnitude of the Fourier descriptors, is not affected by rotation.
- **Scaling, translation.** In the proposed algorithm, a normalization is applied in the polygonal line before the detection step. This normalization scales and translates the polygonal line in a way that makes the mean and the variance of both x and y vertex coordinates zero and one respectively. By doing so, the watermark is not affected by uniform scaling and translations. The watermark is not affected by translation for an additional reason; translation affects only the DC term of the Fourier transform. As the watermark is not embedded in the DC term, but in the middle frequencies, translations of the polygonal line do not affect the watermark.
- **Change of traversal starting vertex.** This is the case when a different vertex is chosen as a starting point for the polygonal line. Such a change has no visual impact on the polygonal line, but could lead to 'de-synchronization' during detection and cause failure of the watermark detector. However, in this case, the magnitude of the Fourier descriptors remains the same [33] and the watermark withstands the attack.
- **Inversion of traversal direction.** If the vertices of the polygonal line are presented in the reverse order, the synchronization will be lost and the algorithm will not work. However, solutions to this problem exist: one can choose always the same traversal direction (clockwise or counterclockwise) during embedding and detection, i.e. traverse the polygonal lines always in in the same direction even if the its vertices are listed the other way, or choose watermark signals that are symmetric with respect to their middle point, so that

traversal direction has no effect.
- **Mirroring.** Mirroring the polygonal line causes mirroring of the Fourier descriptors. Performing detection on both the polygonal line and its mirrored form can solve this problem.

C. Correlator detector

Given a possibly watermarked polygonal line, the watermark detector aims at verifying whether it hosts a certain watermark $\mathbf{W}$ or not. Two hypotheses should be checked:

- H_0: The polygonal line does not host watermark $\mathbf{W}$
- H_1: The polygonal line hosts watermark $\mathbf{W}$

Hypothesis H_0 can occur either in the case that the polygon is not watermarked (sub-hypothesis H_{0a}) or when it is watermarked by another watermark $\mathbf{W}'$, where $\mathbf{W} \neq \mathbf{W}'$, (sub-hypothesis H_{0b}).

In [4] detection using a correlator detector was proposed. The test statistic used in this detector is the correlation between the watermark and the Fourier descriptors magnitude of the polygonal line under examination. Let $\mathbf{M} : M[k], k = 0, 1, ..., N-1$ be the vector of the Fourier descriptors magnitude for the polygonal line under consideration. The above mentioned correlation can be expressed as:

$$C = \frac{1}{N} \sum_{k=0}^{N-1} M[k]W[k] \tag{7}$$

In order to decide on the valid hypothesis, C is compared against a suitably selected threshold T. Watermark is declared present and hypothesis H_1 is chosen if $C > T$, H_0 is decided if $C < T$.

IV. Optimal watermark detection

The statistical detection theory states that the correlator is the optimal signal (watermark) detector if the noise (which in the case of watermarking is the host signal, i.e. the Fourier descriptors magnitude) is additive and the noise samples are Gaussian independent random variables. In the watermarking scheme described above, the watermark is not additive (see 4). Furthermore, the Fourier descriptors magnitude does not follow a Gaussian distribution.

Thus, a better watermark detector can be designed if the statistics of the Fourier descriptors magnitude are modelled more accurately and taken into consideration. To this end, in this section we follow an approach similar to that of [34] and [35] that was originally proposed in the context of 2D raster images, to obtain expressions for an optimal detector for polygonal lines watermarked using the methodology presented in Section III-A.

A. Likelihood Ratio Test

The proposed approach is based on the Bayes decision theory, and the likelihood ratio test (LRT). In the rest of the paper we will consider only those samples $M[k]$ that might have been affected by watermarking, (i.e. the samples for which the watermark $W[k]$ in (5) obtains non-zero values) and denote by A the set of k values that are indexing those samples. Each sample $M[k], k \in A$ of $\mathbf{M}$ is a random variable, with conditional probability density functions for the two events H_0 and H_1, $p(M[k] \mid H_0)$ and $p(M[k] \mid H_1)$ respectively. The LRT can be defined as:

$$\Lambda = \frac{p(\mathbf{M} \mid H_1)}{p(\mathbf{M} \mid H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} T \tag{8}$$

If $M[k]$ are assumed to be independent then:

$$p(\mathbf{M}|H_j) = \prod_{k \in A} p(M[k]|H_j) \; j = 0, 1 \tag{9}$$

and the LRT has the following form:

$$\Lambda = \frac{\prod_{k \in A} p(M[k]|H_1)}{\prod_{k \in A} p(M[k]|H_0)} \tag{10}$$

For hypothesis H_{0a}, and assuming that no distortions occurred in the signal, $M[k] = |X[k]|$ and thus:

$$p(M[k]|H_0) = f(|X[k]|) \tag{11}$$

where $f(|X[k]|)$ denotes the probability density function (pdf) of the Fourier descriptors magnitude $|X[k]|$ of the original signal. By taking into account the fact that the transformation (4) is applied to the random variable $|X[k]|$, the pdf of the watermarked Fourier descriptors magnitude $p(M[k]|H_1)$ can be expressed as a function of the original signal Fourier descriptor magnitude pdf $f(|X[k]|)$ [33]:

$$p(M[k]|H_1) = \frac{1}{|1 + sW[k]|} f(|X[k]|/(1 + sW[k])) \tag{12}$$

Given that a bi-valued $[-1, 1]$ watermark is used and $0 < s < 1$, (12) can be rewritten as:

$$p(M[k]|H_1) = \frac{1}{1 + sW[k]} f(|X[k]|/(1 + sW[k])) \tag{13}$$

By substituting (13) and (11) in (10), the LRT Λ can be calculated in terms of $f(|X[k]|)$:

$$\Lambda = \frac{\prod_{k \in A} f(|X[k]|/(1 + sW[k]))}{\prod_{k \in A} (1 + sW[k]) f(|X[k]|)} \underset{H_0}{\overset{H_1}{\gtrless}} T \tag{14}$$

Thus Λ can be expressed analytically if an expression for $f(|X[k]|)$ can be obtained. Such an expression will be derived in Section IV-B. In practice the quantity $\Lambda' = \log(\Lambda)$ which is equal to:

$$\begin{aligned} \Lambda' = &\sum_{k \in A} \log(f(|X[k]|/(1 + sW[k]))) \\ &- \sum_{k \in A} \log((1 + sW[k]) f(|X[k]|)) \end{aligned} \tag{15}$$

was used instead of Λ and all experimental results refer to this quantity.

The detection performance of a watermarking system can be measured by using two error probabilities, namely the probability of false alarm P_{fa} (defined as the probability to decide that a watermark exists in a signal that is not watermarked or, is watermarked with a different watermark) and the probability of false rejection P_{fr} (i.e. the probability of failing to detect a watermark in a signal that is watermarked).

B. Evaluation of the probability density function of the Fourier descriptors magnitude

The forward DFT transform in (3) is essentially a summation with a large number of terms that can be assumed to be random variables with a small correlation. Thus by applying the Central Limit Theorem for random variables with small dependency [36], $X_R[k]$ and $X_I[k]$ can be modelled as normally distributed random variables having different mean and variance for each k. Furthermore, if the variances of the real and imaginary part for the same value of k are assumed to be the same, i.e., if $\sigma_k^2 = \sigma_{X_{Rk}}^2 = \sigma_{X_{Ik}}^2$, then the magnitude $|X[k]|$ follows a Rayleigh distribution [33]:

$$f(|X[k]|) = \frac{|X[k]|}{\sigma_k^2} \exp\left(-\frac{|X[k]|^2}{2\sigma_k^2}\right), \quad k > 0. \tag{16}$$

The variance σ_k^2 of this distribution has to be estimated for all different values of k using the signal that is available during detection, i.e. $M[k]$. Since this signal might be watermarked $M[k]$ might not be equal to $|X[k]|$. However one can assume that the watermark does not affect significantly the estimated variance, as its amplitude is very small compared to that of the host signal. Thus, it can be safely assumed that $M[k] \approx |X[k]|$.The approach we propose is to estimate the variance of $M[k]$ for a specific value of $k \in A$ by considering the samples $M[i]$, $(k-P) < i < (k+P)$ in a small window around $M[k]$, i.e. we assume similarly to [34] that for small P, the samples of this window, i.e. the samples close to $M[k]$ are identically distributed with $M[k]$. Thus, mean and variance are estimated as follows:

$$\widehat{\mu}_k = \frac{\sum_{i=k-P}^{k+P} M[i]}{2P+1}, \quad k \in A \tag{17}$$

$$\widehat{\sigma}_k^2 = \frac{\sum_{i=k-P}^{k+P} (M[i] - \widehat{\mu}_k)^2}{2P}, \quad k \in A \tag{18}$$

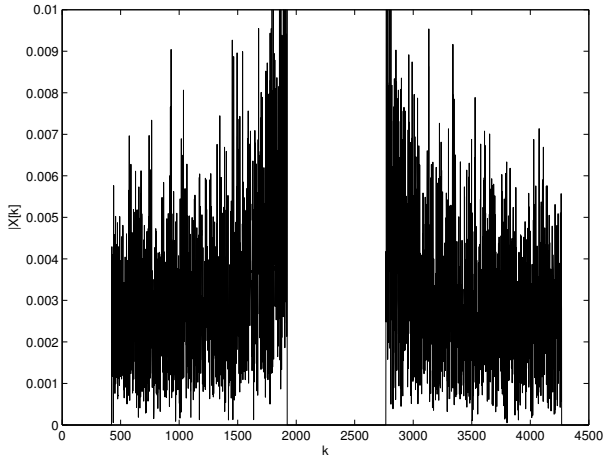
An alternative and computationally more efficient approach is to evaluate the above estimators in a window, and assign the estimated values to all the samples in this window:

$$\widehat{\mu}_k = \widehat{\mu}_l = \frac{\sum_{i=2Pl+1}^{2P(l+1)} M[i]}{2P}, \quad \forall k \in [2Pl+1, \ldots, 2P(l+1)], \quad l = 0 \ldots \tag{19}$$

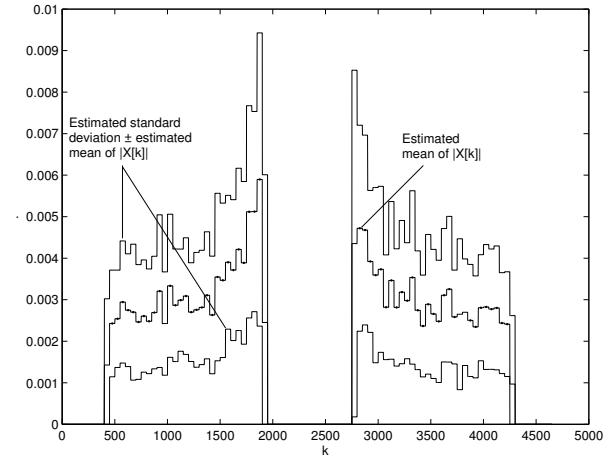
$$\widehat{\sigma}_k^2 = \widehat{\sigma}_l^2 = \frac{\sum_{i=2Pl+1}^{2P(l+1)} (M[i] - \widehat{\mu}_l)^2}{2P-1}, \quad \forall k \in [2Pl+1, \ldots, 2P(l+1)], \quad l = 0 \ldots \tag{20}$$

A value of $P = 25$ has been experimentally found to yield a good trade-off between accuracy in the variance estimation and fidelity of representation of the actual power spectral density in this case. The magnitude values $M[k], k \in A$ of the Fourier descriptors in the polygonal line shown in Fig. 1 are shown in Fig. 2a. The estimated mean and standard deviation are shown in Fig. 2b.

In order to test the validity of the assumption that $X_R[k]$ and $X_I[k]$ can be modelled as normally distributed random variables we have performed the Kolmogorov-Smirnov test for goodness of fit to a normal distribution [37]. This test evaluates the hypothesis that a set of samples is derived from a gaussian distribution, against the alternative hypothesis that these sample do not stem from a gaussian distribution. As already mentioned, $X_R[k]$ and



(a)



(b)

Fig. 2. (a) Fourier descriptors magnitude for the polygonal line in Fig. 1. (b) Mean and mean $\pm$ standard deviation estimated using the procedure in section IV-B.

$X_I[k]$ are assumed to be gaussian but not identically distributed, i.e., with different mean and variance for each k. Thus, ideally, the gaussianity of each coefficient should be checked separately. However the method for estimating the mean and variance of coefficients presented above assumes that coefficients within a window of size $2P = 50$ have approximately the same mean and variance. Thus the Kolmogorov-Smirnov test has been applied as follows: For each of the polygonal lines that have been tested, the real and imaginary parts of the Fourier descriptors coefficients for values of $k \in A$, were split to contiguous non-overlapping sets each consisting of $2P = 50$ contiguous coefficients and the 5% level test has been applied on these sets. Results for various polygonal lines showed that the gaussianity assumption is a safe one, i.e. the test declared that the null hypothesis of gaussian distribution of the samples cannot be rejected for none of the sets of real and imaginary parts of the coefficients.

It should be noted that certain assumptions that were adopted in the derivations presented above, like the assumption that the

magnitudes of the Fourier descriptors are independent random variables, and the assumption that $X_R[k]$ (and also $X_I[k]$) are identically distributed for all k within a small window, might not hold in practice, or be only partially valid, depending on the polygonal line under investigation. However, these assumptions were deemed necessary since without them the derivations would be intractable. Similar assumptions were adopted in [34]. Moreover, as will be shown in section VI, the derived detector achieves very good performance, a fact that justifies, up to a certain extent, the adoption of these assumptions.

V. Implementation issues

Vector graphics files and more particularly files in SVG format have certain characteristics that should be taken into account in order to achieve robust and invisible watermarking. Such issues will be examined in this Section.

A. Watermarking embedding and detection in multiple lines

In the previous sections a method for watermarking a single polygonal line has been described. However, vector graphics files or GIS data usually consist of *many* polygonal lines. For the watermarking of such files with the proposed system, the system should be able to deal with the presence of multiple polygonal lines.

Watermarking all lines whose length is above a certain threshold (e.g. 300 vertices) seems to be the most reasonable approach to uniformly distribute the watermark through the file in order to be sure that even after the removal of certain lines it would still be detectable. Furthermore, by doing so, one can get advantage of the availability of more vertices to watermark in order to enhance the detection performance, since, as will be exemplified in Section VI, performance improves with the number of vertices. Thus, a watermark sequence is created for each of the K lines whose length is above a threshold T_L, and embedded as described in Section III. All watermarks are generated using the same watermark key, since choosing a different watermark for each line would make detection difficult in case lines were rearranged within the file. During detection, the system should reach a global decision on whether the set of polygonal lines is watermarked or not. One way to do so would be by using decentralized fusion, i.e., by obtaining for each line a binary (watermarked / not watermarked) decision and then reaching a global decision by e.g. the majority rule. However, we have chosen to use centralized fusion, i.e. reaching a global decision by combining in an appropriate way the information obtained from each polygonal line, as centralized fusion methods have usually better performance than decentralized ones, since decentralized (local) decision-making results in loss of information [38].

If one assumes that the magnitude of the Fourier descriptors $M_i[k], i = 0, ..., K-1, k \in A_i$, of the K lines is not only independent within a certain polygonal line but also independent across different polygonal lines, then the contributions from the K polygonal lines can be combined in a single likelihood ratio test as follows:

$$\Lambda = \frac{\prod_{i=0}^{K-1} \prod_{k \in A_i} p(M_i[k]|H_1)}{\prod_{i=0}^{K-1} \prod_{k \in A_i} p(M_i[k]|H_0)} \tag{21}$$

This value, compared against a suitable threshold can be used to decide whether the set of lines is watermarked or not. Similar to the single line case, $\Lambda' = log(\Lambda)$ was used instead of Λ. As will be shown in Section VI, the proposed fusion rule outperforms other rules in terms of detection performance.

Fig. 3. GIS data (Scotland and England) depicting polygonal lines that share a number of points.

B. Watermarking of adjacent or overlapping polygonal lines

In certain cases, a vector graphics file might be composed of several polygonal lines that are adjacent and share a number of points. This is the case of GIS data, where regions (e.g. countries, counties etc.) defined on a map naturally share borders and thus certain points belong to both polygonal lines defining the adjacent regions. As an example, Fig. 3 shows two polygonal lines, corresponding to England (1095 points) and Scotland (1132 points). Furthermore, vector graphics files might consist of multiple overlapping shapes arranged in a layered way. The fill and outline of shapes that lay behind other shapes might be partially visible or not visible at all and the overlayed shapes can share a border.

In both cases described above, modification of common points is a delicate issue, as the slightest change in the part of the polygonal line that is shared with another adjacent line would cause the two lines to separate and changes would become apparent. In a similar way, in two overlapping shapes that share a border a change on one shape due to watermarking might reveal the other shape and cause visible distortions.

The only way not to cause visible distortions in this case is to keep the two lines coincident on their shared part after watermark insertion. This is relatively easy to handle when only one of the involved lines is watermarked since one can apply the distortions that occur in the watermarked line to the second line and retain the common border. However, if both polygonal lines have to be watermarked, other solutions should be sought. Options include:

- Method 1. Watermark one polygonal line while also applying the corresponding distortion to the second line and then repeat the same procedure on the second line. In this case, the shared part of the two lines will be modified by both watermarks.
- Method 2. Watermarking independently both polygonal lines, and subsequently move each shared point to the average of the two positions generated by the watermarking embedding.

Obviously, the points shared by the two lines that are affected by the two successive watermarking operations will provide

distorted information when individually detecting each of the watermarks. The influence of double watermarking the shared points can be seen as noise, and as such it should be quantified. In Section VI a set of experiments that aimed at pinpointing the best among the methods mentioned above are presented. It should be noted however that the approaches presented above are applicable only in cases where two or more shapes share the same vertices. If for example two overlapping shapes are close to one another but do not share the same vertices, the approaches above will not work and watermarking can cause the border areas of the background layer to be revealed.

C. Other issues

In many cases, especially in complex and visually "rich" graphics, the outline of a shape in an SVG file is described through the *path* entity which defines this outline using a mixture of adjacent polygonal lines represented as a sequence of line segments and Bezier curves that are specified through their control points. These entities pose no problem for watermark embedding since one can consider all vertices within a path, i.e. the endpoints of the linear segments and the control points of the Bezier curves as a single line, an approach that was followed in this paper. However, since the Bezier curves can define a large section of a smooth outline with only a few control points, modifying these control points in order to embed watermark information might result in the modification of a large part of the outline, a fact that in some cases can cause visible distortions. A possible solution to this problem is to sample the Bezier curve and convert it to a polygonal line consisting of multiple small segments.

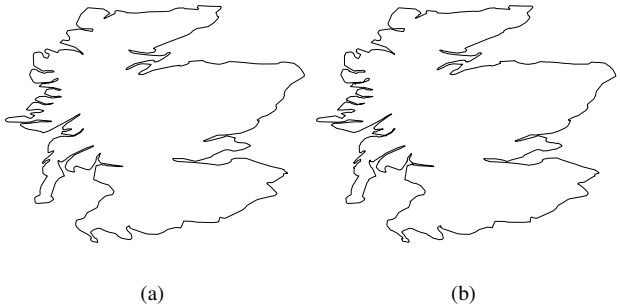
Finally, it is important to note that the fact that not all shapes and lines are watermarked results in a very effective visual masking, especially in complex graphics. In other words, even if the modifications induced by the watermark on a certain shape are visible when this shape is viewed separately, these changes are hardly noticeable when viewing the whole graphic image. Furthermore, shapes whose outline is not visible due to the presence of other shapes in the foreground can be used to achieve a totally invisible watermarking.

VI. Experimental results

Experiments over several polygonal lines of different characteristics were performed in order to verify that the proposed optimal detector is superior to the correlator and to test the behavior of the system to different attacks. Fig. 4a presents such a polygonal line, consisting of 1132 vertices depicting the mainland of Scotland. The watermarked line can be seen in Fig. 4b

As mentioned in Section IV, the performance of a watermarking system is determined by P_{fa} and P_{fr}. Both probabilities depend on the threshold T used during detection. The plot of $P_{fa}(T)$ versus $P_{fr}(T)$ for different thresholds is the *Receiver Operating Characteristic* (ROC) curve which can be used to judge the performance of a system under various operating conditions. The point of the ROC where $P_{fa} = P_{fr}$ is called the Equal Error Rate (EER) and can be used as a simple scalar metric of the performance of an algorithm.

Unfortunately $P_{fa}(T)$ and $P_{fr}(T)$ are not very easy to estimate. Such an estimation involves counting, for different values of the threshold T, the number of errors (erroneously detected watermarks or missed watermarks) on a large set of experiments,



(a) (b)

Fig. 4. Original (a) and watermarked (b) test polygonal line.

This method is obviously not practical, since in our case we are dealing with very low error probabilities and thus the number of trials that should be performed should be very large. In order to proceed with the estimation, it was assumed that the output Λ' of the optimal detector is a Gaussian random variable since it is the sum of a large number of random variables that can be considered to be weakly correlated. Thus, the pdf of the detector output under the two hypotheses can be fully determined in terms of its conditional mean $\mu_{\Lambda'|H_0}$, $\mu_{\Lambda'|H_1}$ and variance $\sigma^2_{\Lambda'|H_0}$, $\sigma^2_{\Lambda'|H_1}$.

Having accepted the gaussianity of the optimal detector output, error probabilities can be conveniently expressed as:

$$P_{fr}(T) = \int_{-\infty}^{T} \frac{1}{\sqrt{2\pi}\sigma_{\Lambda'|H_1}} \exp\left(\frac{(x-\mu_{\Lambda'|H_1})^2}{2\sigma^2_{\Lambda'|H_1}}\right) dx \quad (22)$$

$$P_{fa}(T) = \int_{T}^{\infty} \frac{1}{\sqrt{2\pi}\sigma_{\Lambda'|H_0}} \exp\left(\frac{(x-\mu_{\Lambda'|H_0})^2}{2\sigma^2_{\Lambda'|H_0}}\right) dx \quad (23)$$

In order to verify the assumption of the gaussianity of Λ' under both H_0 and H_1, various polygonal lines were watermarked 1000 times, each time with a different watermark, and subsequently the proposed detection algorithm was used to detect the watermark that was indeed embedded in the line as well as a watermark produced with a different key. Subsequently the Kolmogorov-Smirnov 5% level test for goodness of fit to a normal distribution was applied on the two sets of 1000 Λ' values that were generated. In all cases, the test showed that the gaussianity hypothesis cannot be rejected.

In addition, the gaussianity assumption was verified by comparing the ROC curve obtained by counting the erroneously detected and missed watermarks after a large number of experiments with the ROC curve obtained under the gaussianity assumption. In the latter case, the mean and variance $\mu_{\Lambda'|H_0}$, $\mu_{\Lambda'|H_1}$ $\sigma^2_{\Lambda'|H_0}$, $\sigma^2_{\Lambda'|H_1}$ of the optimal detector output for watermarked and not watermarked polygonal lines were estimated through two sets of 10000 experiments each, using different keys and involving detection on watermarked data with the correct watermark keys and detection on non watermarked data. The good agreement of the two curves (Figure 5) justifies our ROC evaluation methodology. The same procedure was used in all experiments involving ROC curves or EER evaluation.

The correlator output (7) is the sum of a number of random variables of the form $M[k]W[k]$. Even though samples $M[k]$ are not independent, correlation is rather weak and thus the terms $M[k]W[k]$ can be also considered weakly correlated. Thus by

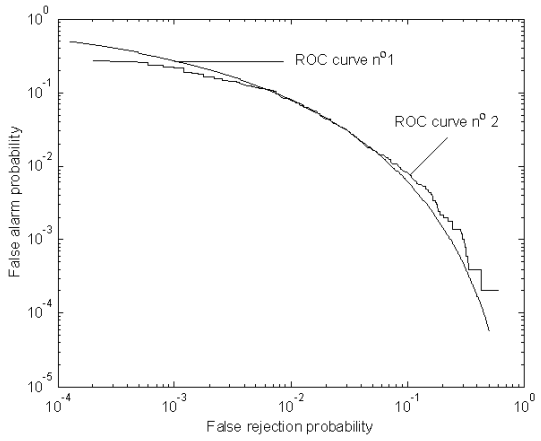


Fig. 5. Gaussianity of optimal detector output. ROC curve n.1 was created under the assumption of gaussianity, ROC curve n.2 was created from actual errors after a set of experiments.

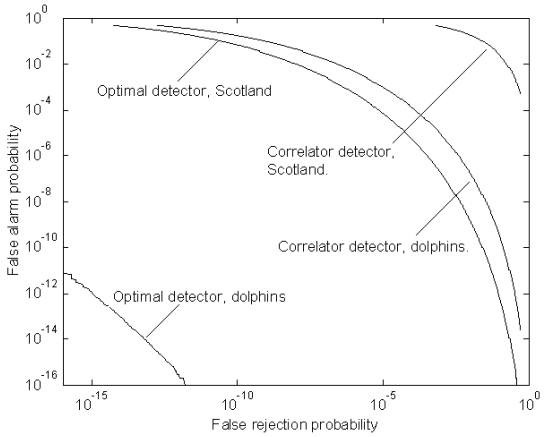


Fig. 6. ROC curves for the correlator and the optimal detector for two different polygonal lines (Fig. 4, 1132 vertices and Fig. 1, 4689 vertices). No attacks were inflicted on the polygonal lines.

using the central limit theorem the output of the correlator can be also assumed to be normally distributed [36].

The ROC curves for the two test polygonal lines shown in Fig. 1 and Fig. 4 (calculated under the detector output gaussianity assumption), for both the correlator and the optimal detector, can be seen in Fig. 6. An embedding power of $s = 0.140$ has been used in both cases. Results show that the quality of the detection improves with the number of vertices in the polygonal line. The optimal detector performs in both cases considerably better than the correlator. Similar results were obtained for all the polygonal lines that were tested.

Furthermore, the proposed method was compared against the method reported in [11]. This method was selected among the 2D vector graphics watermarking methods reviewed in Section II because it shares the same characteristic with the proposed method i.e., it is a zero-bit, blind method. The ROC curves for the two test polygonal lines depicting Scotland and England shown in Fig. 3 for both the proposed method and the method in [11] are depicted in Fig. 7. The parameters of both methods were selected so that the distortions induced by the watermark were invisible and of the same magnitude. In other words, the parameters were

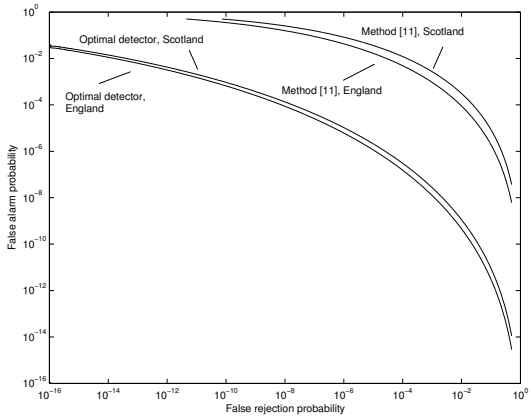


Fig. 7. ROC curves for the proposed method and the method in [11] for the two polygonal lines depicted in Fig. 3. No attacks were inflicted on the polygonal lines.

TABLE II

EER OBTAINED FOR SUBSAMPLED VERSIONS OF THE POLYGONAL LINES THAT HAVE BEEN SELECTED FOR WATERMARKING IN THE GRAPHICS FILE DEPICTED IN FIG. 1. THE ORIGINAL LINES CONSISTED OF 4689 VERTICES.

Vertices	4689	2345	1563	1173
EER	$5.42 \cdot 10^{-40}$	$9.11 \cdot 10^{-9}$	$4.99 \cdot 10^{-5}$	$1.67 \cdot 10^{-3}$

chosen so that the SNR between the original and the watermarked curves for both methods was equal to 74dB. It is obvious that the proposed method achieves better detection results that the method in [11].

The effect of the number of available vertices on the performance of the optimal detector has been exemplified with a suitable experiment. More specifically, the lines that have been selected for watermarking in the graphics file depicted in Fig. 1 have been subsampled with factors $1/2$, $1/3$ and $1/4$ and tests with multiple watermarking trials (embedding power $s = 0.20$) were conducted in order to calculate the EER in each case. Results are presented in Table II. One can easily notice that the detection performance decreases with the number of available vertices.

In addition, several attacks were tried out in order to assess the robustness of the watermark against attacks on the host signal. The ROC curves for these attacks are shown in Fig. 8. Translations, rotations (Fig. 9(a)) and isotropic scaling were applied on the polygonal line in Fig. 4, and did not affect at all the watermark, i.e., as expected, the ROC curve did not change at all. Therefore, ROC curves for these attacks are not presented here. Furthermore, addition of gaussian noise independently in both x, y coordinates of vertices (Fig. 9(b)), independent low pass filtering of the two coordinates using a moving average filter of window length 5 (Fig. 9(c)) and anisotropic scaling with scaling factors 2 and 0.5 in the X and Y dimension respectively (Fig. 9(d)) were applied on the polygonal line. The results prove that the method is sufficiently robust to these attacks.

Another set of experiments aimed at judging the performance of the method proposed in Section V-A for watermarking a set of polygonal lines and reaching a global decision on whether the set is watermarked or not, using (21). The proposed fusion method was compared against fusion methods tested in a similar context in [39]. In that paper, the authors used the correlator detector to

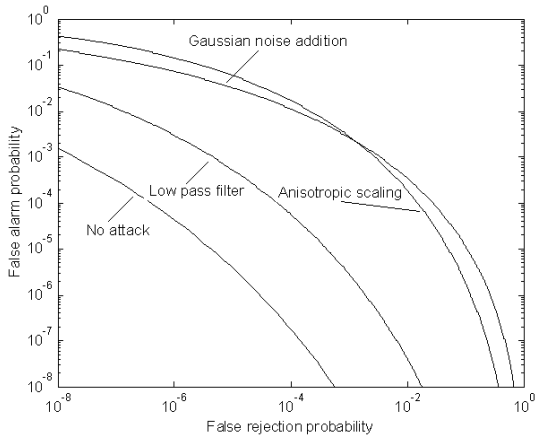


Fig. 8. ROC curves of the optimal detector for the polygonal line in Fig. 4 being subject to three different attacks.

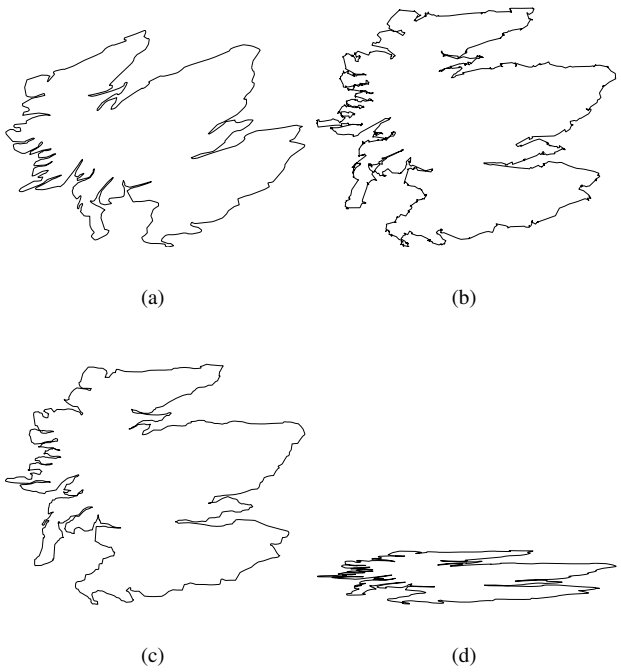


Fig. 9. Manipulated versions of the polygonal line of Fig. 4: (a) rotated (30 degrees), (b) distorted by additive gaussian noise, (c) low-pass filtered, (d) scaled with different factors in the two dimensions.

detect watermarks in individual polygonal lines and subsequently used a number of simple fusion rules in order to combine the detector outputs C_i (7). In a similar manner, the proposed fusion rule (21) was tested against the following rules for combining the individual optimal detector outputs $\Lambda'_i, i = 0, ..., K-1$ from K different polygonal lines:

- Median value of Λ'_i
- Minimum value of Λ_i
- Maximum value of Λ'_i
- Weighted mean of Λ'_i, where the weight assigned to output Λ'_i is equal to the number of vertices N_i of the corresponding polygonal line:

TABLE III
EER OBTAINED FOR THE DIFFERENT FUSION RULES WHEN APPLIED ON A SET OF 12 POLYGONAL LINES.

Fusion rule	EER
LRT, eq.(21)	3.12E-14
Weighted mean (N_i)	1.26E-12
Weighted mean (N_i^2)	1.80E-10
Minimum	4.36E-10
Median	1.21E-5
Maximum	1.86E-3

$$\Lambda = \frac{\sum_{i=0}^{K-1} N_i \Lambda_i}{\sum_{i=0}^{K-1} N_i} \tag{24}$$

- Weighted mean of Λ'_i, where the weight assigned to output Λ'_i is equal to the square of the number of vertices N_i^2 of the corresponding polygonal line:

$$\Lambda = \frac{\sum_{i=0}^{K-1} N_i^2 \Lambda_i}{\sum_{i=0}^{K-1} N_i^2} \tag{25}$$

Table III presents the EER obtained when applying the various fusion rules on a set of 12 polygonal lines that represent borders of countries from GIS data, each having less than 1000 points. The results verify that the fusion method proposed in Section V-A provides the best performance. The weighted mean of the detector outputs, with weights that correspond to the length of each line (and thus favor lengthier lines) was the rule that obtained the second best performance. Experiments involving other sets of lines offered similar results.

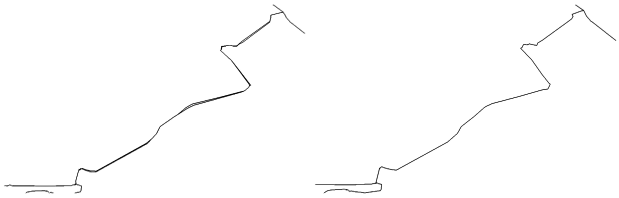
Moreover, a set of experiments was carried out in order to asses the performance of the two methods proposed in Section V-B for handling polygonal lines that share a number of points. Within this experiment, two pairs of polygonal lines, namely the pair shown in Fig. 3 and an additional one, were watermarked using the two procedures. Table IV presents the number of points shared between the two lines at each pair, as well as the EER obtained for each of the two methods, and the EER obtained when watermarking each line independently, without providing special handling for the shared points. Fusion of the detector results obtained from each individual line was performed using (21). As expected, the best results were obtained when no special handling of the shared points is performed (last column of the table). However, as can be seen in Fig. 10a this approach results in visible distortions since the two lines do not coincide at their common segment after the watermark embedding. Among the two other methods, method 1 provided the best detection results, which were also very close to the ones obtained when no correction was performed. Thus this method can handle efficiently the case of lines that share a number of points, not only in terms of detector efficiency, but also in terms of invisibility of the distortions as can be seen in Fig. 10b.

Additional examples of watermarked SVG files along with the original data can be seen in Fig. 11 (embedding power $s = 0.20$, EER: $2.69 \cdot 10^{-11}$, one watermarked line) and Fig. 12 (embedding power $s = 0.15$, EER: $5.51 \cdot 10^{-9}$, three watermarked lines). Fig. 11 exemplifies the visual masking effect introduced by the existence of shapes that have not been watermarked, that has been described in Section V. The SVG file in Fig. 12 contained multiple overlapping shapes that share borders in certain instances. The

TABLE IV

EER FOR THE TECHNIQUES PROPOSED FOR HANDLING POLYGONAL LINES THAT SHARE A NUMBER OF POINTS. RESULTS FOR TWO PAIRS OF LINES ARE PRESENTED.

	shared points (percentage)	EER method 1	EER method 2	EER no correction
pair 1	11%	4.10 E-7	7.31 E-7	2.99 E-7
pair 2	2%	3.90 E-13	9.87 E-13	3.30 E-13



(a) (b)

Fig. 10. (a) Detail of two polygonal lines that share a number of points obtained when watermarking these lines without taking care of the shared points. (b) Results obtained by using Method 1 described in Section V-B.

approach presented in Section V to handle such instances has been successfully used in this case. It has to be noted that the EER values when applying the correlator detector on the same data were $1.56 \cdot 10^{-6}$ for the SVG file in Fig. 11 and $1.11 \cdot 10^{-4}$ for the SVG file in Fig. 12.

VII. CONCLUSIONS AND FUTURE WORK

An optimal, blind detector structure for watermarked polygonal lines in 2D vector graphics data (e.g. in SVG format) has been presented in this paper. The performance of the proposed detector was assessed in terms of detection error probabilities and ROC curves. The detector was compared against the correlator detector proposed in [4], verifying its superiority.

As expected, the algorithm proved to be totally immune to a number of geometric distortions (rotation, isotropic scaling, change of traversal starting point, etc). Sufficient robustness to other attacks like noise addition and anisotropic scaling was also showcased. Procedures for watermarking sets of polygonal lines and reaching a global decision on whether these sets are watermarked or not, as well as techniques for handling polygonal lines that share a number of points were proposed and experimentally evaluated. Other important issues related to the application of the algorithm on vector graphics images with emphasis on SVG files were also discussed.

In general, the proposed algorithm provides a very efficient approach to robust watermarking of vector graphics images, achieving in most cases very good invisibility and good detection performance. Its major advantage with respect to existing approaches is that it achieves very good performance in a blind detection framework whereas most of the proposed algorithms require the original graphics image to be available during detection. Currently, the algorithm is not sufficiently robust to polygonal line simplification (vertex removal). Future work will try to deal with this issue.

ACKNOWLEDGEMENT

This work has been supported in part by the European Commission through the IST Programme under Contract IST-2002-507932 ECRYPT.

REFERENCES

[1] I. Cox, J. Bloom, and M. Miller, *Digital Watermarking: Principles & Practice*. Morgan Kaufmann, 2001.

[2] G. Voyatzis and I. Pitas, "Protecting digital-image copyrights: A framework." *IEEE Computer Graphics and Applications*, vol. 19, no. 1, pp. 18–24, January/February 1999.

[3] A. Tefas, N. Nikolaidis, and I. Pitas, "Watermarking techniques for image authentication and copyright protection," in *Handbook of Image and Video Processing*. Al Bovik (editor), Elsevier, 2005.

[4] V. Solachidis and I. Pitas, "Watermarking polygonal lines using fourier descriptors," *IEEE Computer Graphics and Applications*, vol. 24, no. 3, pp. 44–51, May/June 2004.

[5] V. Solachidis, N. Nikolaidis, and I. Pitas, "Watermarking polygonal lines using fourier descriptors," in *Proceedings of IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'2000)*, vol. 4, 2000, pp. 1955–1958.

[6] V. Doncel, N. Nikolaidis, and I. Pitas, "Watermarking polygonal lines using an optimal detector on the fourier descriptors domain," in *Proc. of 2005 EURASIP European Signal Processing Conference (EUSIPCO 2005)*, September 2005.

[7] C. Lopez, "Watermarking of digital geospatial datasets: a review of technical, legal and copyright issues," *International Journal of Geographical Information Science*, vol. 16, no. 6, pp. 589–607, September 2002.

[8] S. Battiato, G. D. Blasi, G. Gallo, and E. Messina, "Firemark: a java tool for SVG watermarking," in *Proc. of 4-th Annual Conference on Scalable Vector Graphics (SVGOpen 2005)*, 2005.

[9] M. Voigt and C. Busch, "Watermarking 2d-vector data for geographical information systems," in *Proc. SPIE, Security and Watermarking of Multimedia Content*, January 2002, pp. 621–628.

[10] K. Hwan, K. Kab, and C. Jong, "A map data watermarking using the generalized square mask," in *Proceedings of IEEE International Symposium on Industrial Electronics*, vol. 3, 2001, pp. 1956–1958.

[11] Y. Li and L. Xu, "A blind watermarking of vector graphics images," in *Proceedings of 5th International Conference on Computational Intelligence and Multimedia Applications, (ICCIMA 2003)*, 2003, pp. 424–429.

[12] H. Sonnet, T. Isenberg, J. Dittmann, and T. Strothotte, "Illustration watermarks for vector graphics," in *Proceedings of 11th Pacific Conference onComputer Graphics and Applications*, 2003, pp. 73–82.

[13] R. Ohbuchi, H. Ueda, and S.Endoh, "Robust watermarking of vector digital maps," in *Proceedings of IEEE International Conference on Multimedia and Expo 2002 (ICME2002)*, August 2002.

[14] R. Ohbuchi, H. Ueda, and S. Endoh, "Watermarking 2d vector maps in the mesh-spectral domain," in *Proceedings of the Shape Modeling International*, 2003, pp. 216–225.

[15] Z. Karin and C. Cotsman, "Spectral compression of mesh geometry," in *Proceedings of SIGGRAPH 2000*, 2000, pp. 279–286.

[16] I. Kitamura, S. Kanai, and T. Kishinami, "Copyright protection of vector map using digital watermarking method based on discrete fourier transform," in *Proceedings of IEEE International Geoscience and Remote Sensing Symposium, 2001, (IGARSS '01)*, vol. 3, 2001, pp. 1191–1193.

[17] H. Gou and M. Wu, "Data hiding in curves with application to fingerprinting maps," *IEEE Transactions on Signal Processing*, vol. 53, no. 10, pp. 3988–4005, October 2005.

[18] R. Ohbuchi, H. Masuda, and M. Aono, "Data embedding algorithms for geometrical and non-geometrical targets in three-dimensional polygonal models," *Computer Communications*, no. 21, pp. 1344–1354, October 1998.

[19] E. Garcia and J.-L. Dugelay, "Texture-based watermarking of 3D video objects," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 8, pp. 853–866, August 2003.

[20] O. Benedens, "Geometry-based watermarking of 3D models," *IEEE Computer Graphics and Applications*, vol. 19, no. 1, pp. 46–55, January/February 1999.

[21] K.-R. Kwon, S.-G. Kwon, S.-H. Lee, T.-S. Kim, and K.-I. Lee, "Watermarking for 3d polygonal meshes using normal vector distributions of each patch," in *Proceedings of IEEE International Conference on Image Processing*, vol. 2, 2003, pp. 499–502.

[22] F. Cayre and B. Macq, "Data hiding on 3D triangle meshes," *IEEE Transactions on Signal Processing*, vol. 51, no. 4, pp. 939–949, 2003.

[23] J.-W. Cho, R. Prost, and H.-Y. Jung, "An oblivious watermarking for 3-D polygonal meshes using distribution of vertex norms," *IEEE Transactions on Signal Processing*, vol. 55, no. 1, pp. 142–155, January 2007.
[24] S. Zafeiriou, A. Tefas, and I. Pitas, "Blind robust watermarking schemes for copyright protection of 3D mesh objects," *IEEE Transactions on Visualization and Computer Graphics*, vol. 11, no. 5, pp. 596–607, May 2005.
[25] A.-G. Bors, "Watermarking mesh-based representations of 3-d objects using local moments," *IEEE Transactions on Image Processing*, vol. 15, no. 3, pp. 687–701, March 2006.
[26] F. Uccheddu, M. Corsini, and M. Barni, "Wavelet based blind watermarking of 3D models," in *ACM Multimedia and Security Workshop*, 2004, pp. 143–154.
[27] M. Lounsbery, T. D. DeRose, and J. Warren, "Multiresolution analysis for surfaces of arbitrary topological type," *ACM Transactions on Graphics*, vol. 16, no. 1, pp. 34–73, 1997.
[28] F. Cayre, P. Rondao-Alface, F. Schmitt, B. Macq, and H. Matre, "Application of spectral decomposition to compression and watermarking of 3D triangle mesh geometry," *Signal Processing: Image Communication*, vol. 18, no. 4, pp. 309–319, April 2003.
[29] G. Taubin, T. Zhang, and G. Golub, "Optimal surface smoothing as filter design," IBM, Technical Report RC-20404, March 1996.
[30] K. Murotani and K. Sugihara, "Watermarking 3D polygonal meshes using the singular spectrum analysis," in *Proceedings of IMA Conference on the Mathematics of Surfaces*, 2003, pp. 85–98.
[31] N. Golyandina, V. Nekrutkin, and A. Zhigljavsky, *Analysis of Time Series Structure: SSA and Related Techniques*. Chapman & Hall/CRC, 2001.
[32] A. K. Jain, *Fundamentals of Digital Image Processing*. Prentice Hall, 1989.
[33] A. Papoulis, *Probability and Statistics*. Prentice Hall, 1991.
[34] M. Barni, F. Bartolini, A. D. Rosa, and A. Piva, "A new decoder for the optimum recovery of nonadditive watermarks," *IEEE Transactions on Image Processing*, vol. 10, no. 5, pp. 755–767, May 2001.
[35] Q. Cheng and T. S. Huang, "Robust optimum detection of transform domain multiplicative watermarks," *IEEE Transactions on Signal Processing*, vol. 51, no. 4, pp. 906–925, April 2003.
[36] P. Billingsley, *Probability and Measure*. Wiley, 1995.
[37] W. J. Conover, *Practical Nonparametric Statistics*. Wiley, 1980.
[38] B. V. Dasarathy, *Decision Fusion*. IEEE Computer Society Press, 1994.
[39] A.Giannoula, N. Nikolaidis, and I.Pitas, "Watermarking of sets of polygonal lines using fusion techniques," in *Proc. of IEEE International Conference on Multimedia and Expo (ICME 02)*, vol. 2, 2002, pp. 549–552.

Victor Rodriguez Doncel was born in Salamanca, Spain. He received his degree in Telecommunication Engineering from the University of Valladolid in 2001. After working in the industry on the development of operational research and artificial intelligence algorithms, he joined the Aristotle University of Thessaloniki, Greece, to conduct research on watermarking of vector images.

Currently he is working towards a PhD degree in the University Pompeu-Fabra, Barcelona, Spain, on topics related to digital rights management.

Nikos Nikolaidis received the Diploma of Electrical Engineering in 1991 and the PhD degree in Electrical Engineering in 1997, both from the Aristotle University of Thessaloniki, Greece. From 1992 to 1996 he served as teaching assistant in the Departments of Electrical Engineering and Informatics at the same University. From 1998 to 2002 he was postdoctoral researcher and teaching assistant at the Department of Informatics, Aristotle University of Thessaloniki. He is currently a Lecturer in the same Department. Dr. Nikolaidis is the co-author of the book "3-D Image Processing Algorithms" (J. Wiley, 2000). He has co-authored 6 book chapters, 22 journal papers and 82 conference papers. He currently serves as Associate Editor in the International Journal of Innovative Computing Information and Control and the EURASIP Journal on Image and Video Processing. His research interests include computer graphics, image and video processing and analysis, copyright protection of multimedia and 3-D image processing. Dr. Nikolaidis has been a scholar of the State Scholarship Foundation of Greece.

Ioannis Pitas received the Diploma of Electrical Engineering in 1980 and the PhD degree in Electrical Engineering in 1985 both from the Aristotle University of Thessaloniki, Greece. Since 1994, he has been a Professor at the Department of Informatics, Aristotle University of Thessaloniki. From 1980 to 1993 he served as 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 or Visiting Assistant Professor at several Universities. He has published 153 journal papers, 400 conference papers and contributed in 22 books in his areas of interest and edited or co-authored another 5. He has also been an invited speaker and/or member of the program committee of several scientific conferences and workshops. In the past he served as Associate Editor or co-Editor of four international journals and General or Technical Chair of three international conferences. His current interests are in the areas of digital image and video processing and analysis, multidimensional signal processing, watermarking and computer vision.

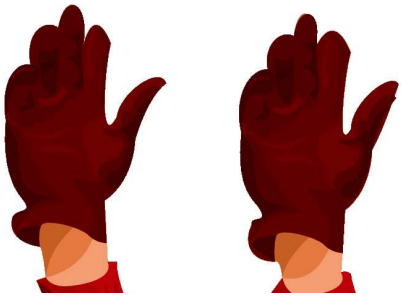
(a)

(b)

(c) (d)

Fig. 11. (a) Original "tiger" SVG graphics file, (b) watermarked graphics file (embedding power $s = 0.20$, EER: $2.69 \cdot 10^{-11}$, one watermarked polygonal line), (c) original polygonal line (496 vertices), (d) watermarked polygonal line.

(a) (b)



(c) (d)

(e)

Fig. 12. (a) Original "cowboy" SVG graphics file, (b) watermarked graphics file (embedding power $s = 0.15$, EER: $5.51 \cdot 10^{-9}$, three polygonal lines with a total of 1523 vertices have been watermarked) with corrections for overlapping shapes, (c) detail of original graphics file, (d) detail of watermarked graphics file with corrections for overlapping shapes, (e) detail of watermarked graphics file with no correction for overlapping shapes. The background shapes that have been watermarked are partially revealed in the last case.