Blind Watermark Approach for Map Authentication Using Support Vector Machine

Mourad Raafat Mouhamed[1,6], Hossam M. Zawbaa[2,6], Eiman Tamah Al-Shammari[3], Aboul Ella Hassanien[4,6], and Vaclav Snasel[5]

[1] Helwan University, Faculty of Science, Cairo, Egypt
mouradraafat@yahoo.com
[2] BeniSuef University, Faculty of Computers and Information, BeniSuef, Egypt
hossam.zawbaa@gmail.com
[3] Faculty of Computing Science and Engineering, Kuwait University
dr.eiman@ku.edu.kw
[4] Faculty of Computers and Information, Cairo University, Egypt
[5] VSB-Technical University of Ostrava, Czech Republic
[6] Scientific Research Group in Egypt (SRGE)
http://www.egyptscience.net

Abstract. This paper presents a blind and robust watermark approach for authentication 2D Map based on polar coordinates mapping and support vector machine is presented. The proposed system is composed of three phases. Firstly, in the preprocessing phase, the proposed algorithm mapped all vertices into polar coordinate system. Then, in the support vector machine phase, the watermark portable points will be chosen using support vector machine to reduce the number of these points which increases the imperceptibility without any effect on the robustness of the watermark. Afterwards, in the watermarking algorithm phase, the watermark is embedded into the map vertices using the random table of the decimal valued of the polar coordinates through the digit substitution of the decimal part. Finally, in the theoretical analysis and experimental results shows that the presented approach is robust against a various attacks including rotation, scaling, and translation. The proposed approach attained high imperceptibility.

Keywords: Support vector machine, vector watermarking, authentication, and geographic information system.

1 Introduction

2D Vector data has a very useful application like computer aided design (CAD) and geographic information system (GIS) where it cost a huge amount of money and time to collect data as in GIS or to design as in CAD. vector maps based on GIS data are rigorous representations of a geographical region and are used for many purposes such as military and civil cartography, urban planning, forestry etc. In those maps, each geographical structure, as a roads , a river or mountain, is defined by a definite number of vertices set in a specific arrangement.

A.I. Awad, A.E. Hassanien, and K. Baba (Eds.): SecNet 2013, CCIS 381, pp. 84–97, 2013.

Production of GIS data is a heavy work. Thus, GIS data constitute a valuable asset which should be protected from digital piracy. The rapid growth of digital technology makes the modification, illegal copy and attack the digital data is simply moreover of intense image processing tools have also made digital image manipulations much easier. In such application of 2D vector data the demands of integrity and authenticity are very tough, and no deformation is permitted [1]. Recently, research on watermarking is concentrated on raster images [2], [3], [4], [5], [6], and [7].

When watermarking GIS data, we have to keep the data distortion low, i.e. the value of the coordinates of the vertices which define Map entities must to be very closely to the value of the coordinates in original map. There are two watermarking techniques when the researcher deal with vector data first one is transform domain techniques and the other is the spatial domain. the transform domain techniques first translate the spatial data into a transform domain and then apply the watermark in that domain The spatial domain techniques deal directly with the coordinates of vertices [9] [10]. Various transforms, like the Fourier descriptors [11] [12], wavelet transform [13] or mesh-spectral domain [14] have been used in the literature.

The watermarking of vector graphics have been developed in and several researches such as changing line features, insertion new vertices, and replacing existing stroke segments by new lines in a stylistic way are described in detail. They can achieve high capacity and robustness, but the watermark can be easily removed by attacks designed specifically for each method. A method for hiding data in curves has been proposed in [15]. It parameterizes a curve using the B-spline model and adds a spread spectrum sequence to the coordinates of the B-spline control points. It is robust against various attacks, such as collusion, cropping, geometric transformations, vector/raster (raster/vector) conversions, printing-and-scanning and some of their combinations. But it requires the original image for integrity verification, i.e., it is non-blind [16].

Jungyeop Kim in [17] used polygon vector feature to embed his watermark in interior angles where it has a weak point that he cannot extract watermarks if the interior angles are changed in our proposed method we use point vector feature to embed our watermark and it was robust for versus attack as translation,rotation and scaling. There is a big challenge to protect the vector map from illegal copyrights and from attacks that can destroy the benefit of the map by changing places and coordinates on it. This will lead to a great loss in confidential data and cost to be reconstructed. This paper presents a blind watermark approach that achieves the authentication in 2D vector map.

The rest of the paper is organized as follows. Section 2 gives an overview of the features of 2D spatial data. Section 3 describes the proposed watermarking map authentication approach including insert and extract watermark processes. Also, the calculation of mapping cartesian to polar coordinates and using the support vector machine. Section 4 discuss different attacks including rotation, translation and scaling. Section 5 presents the experimental result. Section 6 addresses conclusions and future work.

2 Features of Spatial Data: Background

Maps based on GIS data may be represented in spatial or transform domain in our work we will operate with spatial data where it be expressed as vector data. The vector data model represents each surface as a series of isolines; for example, elevation would be represented as a series of contours. However useful for displaying information, it does not easily support the calculation of surface characteristics such as the slope of the surface at a particular point, or the direction that the slope is facing. Both of these characteristics are important for analysis involving surfaces [9].

The road map is a real example of spatial data. A road map is a 2-D representation of object that contains points, polygons, and lines that can represent cities, roads, and political boundaries such as states or provinces. A road map is a imagination of geographic information. The location of cities, roads, and political boundaries that exist on the surface of the Earth are projected onto a 2-D display or piece of paper, preserving the relative positions and relative distances of the rendered objects [25].

The data that indicates the earth location such as (latitude and longitude, height and depth) of these rendered objects is the spatial data. For the rendered map, spatial data is used to project the locations of the objects on a two-dimensional. A GIS is often used to store, retrieve, and render this earth-relative spatial data. The feature of spatial data has its entity representative of using geometry which is built of one or more connected vertices, for more details refer to [8].

3 The Proposed Blind Watermark Approach for Map Authentication

The blind 2D vector watermark approach composed of three fundamental phases as follows:

- *Preprocessing:* The original map data is read as cartesian coordinates (x, y) and it will be converted into polar coordinates (r, Θ). Then, the data will divide into two groups one of them will be the training set and the other will be the testing set.
- *Support vector machine:* classifying data and define a little number of points to add the watermark. Support vector machine (SVM) success to find all watermark points at the extracting process.
- *Watermarking algorithm:* the watermark will be embed within the map image and the proposed method is blind to verify the watermark existing.

Fig. 1 and Fig. 2 depicts the building phases of the proposed system. These phases are described in detail in this section along with the steps involved and the characteristics feature for each phase.

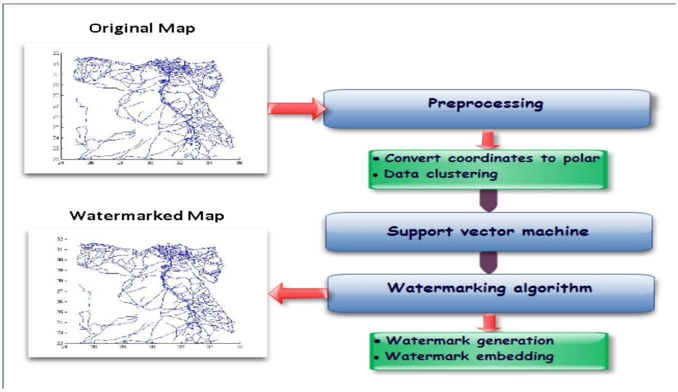


Fig. 1. The watermark embedding process

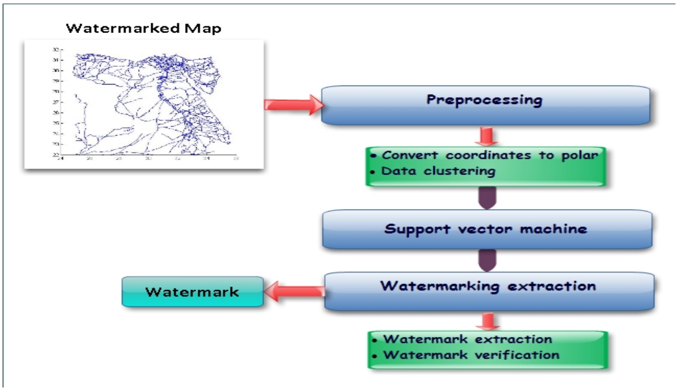


Fig. 2. The watermark extraction process

3.1 Preprocessing

Convert to Polar Coordinates. Mapping the cartesian coordinate system which has an origin point (x_0,y_0) then any point (x, y) in this system can be changed into polar coordinate system to be (r, θ) by the following equations 1 and 2.

$$r = \sqrt{(x - x_0)^2 + (y - y_0)^2} \tag{1}$$

$$\Theta = \frac{y - y_0}{x - x_0} \tag{2}$$

In Figure 3, the origin of this system is $(0, 0)$ so that r expressed the distance between the point $(2, 3)$ and the origin and theta the angle between this radius

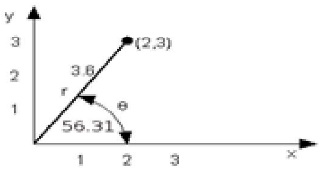


Fig. 3. Mapping from Cartesian to Polar Coordinates

and the x-axis, so that this point in polar coordinate expressed as $(3.6, 56.31)$. But to find this point in Cartesian coordinates again it has to use the following equations 3 and 4:

$$x = r * cos(\Theta) \tag{3}$$

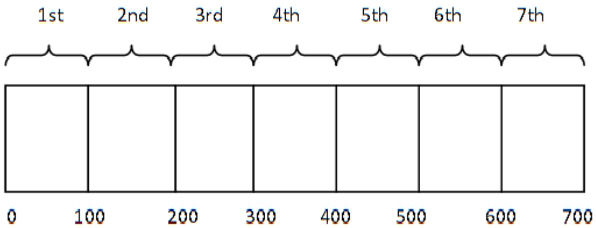
$$y = r * sin(\Theta) \tag{4}$$

Data Grouping and Clustering. The SVM algorithm need to group of data to deal with them and make the matching or classifying, so that the all data r and Θ will sorting descending or ascending separately then all of them will be divided into to group A and B. Group A will be divided to (n) intervals and the border of each interval will be the training data set (i.e the number of border in each point will be (n+1)). This process will happened with group B and the border of the intervals will act the test data set. Finally, the training set and testing set will be the input of the support vector machine.

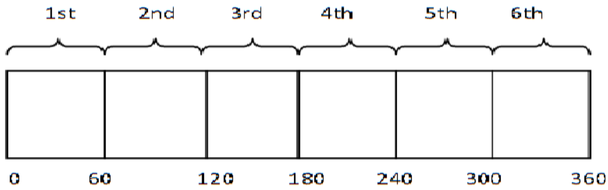
Fig. 4.(a) illustrate that if the the range radius (r) is [0, 700] the range will divided into seven intervals such as [0-100], [100, 200], .., [600-700]. The interval borders (0, 100, 200, 300, ... , 700) will be used to be the training data set or test data set. Fig. 4.(b) show the seme operation, but this operation for finding the borders of Theta (Θ) intervals.

3.2 Support Vector Machine

The efficiency of any watermark approaches measure by the imperceptibility and robustness. The imperceptibility measure by the random mean square error (RMSE) where it expressed the difference between the original and watermarked map and in the spatial data RMSE depend on the number of vertices that will be change and the value of this change. So, if we decrease the number of vertices, the RMSE will be decrease and then the imperceptibility will increase. Machine learning techniques like (Fuzzy c-Mean, K nearest classifier, support vector machine,.. etc) are help in classifying data and define a little number of points to add the watermark. But, when we work with Fuzzy c-Mean (FCM) and K-Mean machine learning techniques, we were found there is randomize and this will lead to instability of the robustness in watermark extraction. Support vector machine (SVM) success to find all watermark points at the extracting process.



(a) Radius interval borders



(b) Theta interval borders

Fig. 4. Interval Borders

SVM has emerged in recent years as a popular approach to the classification of data. SVM is margin-based classifier with good generalization capabilities [18]. It is the method of creating functions from a set of labeled training data. Given a training data set with n samples $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$, where x_i is a feature vector in a v-dimensional feature space and with labels $y_i \in -1, 1$ belonging to either of two linearly separable classes C_1 and C_2. The function can be either a classification function or a general regression function. SVM finds an optimal separating hyper-plane between data points of different classes in a high dimensional space. SVM decoding models are based on the structural risk minimization (SRM) principle from statistical learning theory. In kernel model selection, mostly iterative search is applied in order to optimize the parameters within a specified range [20]. Geometrically, the SVM modeling algorithm finds an optimal hyperplane with the maximal margin to separate two classes, which requires to solve the optimization problem, as shown in equations 5 and 6.

$$maximize \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j . K(x_i, x_j) \tag{5}$$

$$Subject - to : \sum_{i=1}^{n} \alpha_i y_i, 0 \leq \alpha_i \leq C \tag{6}$$

Where, α_i is the weight assigned to the training sample x_i. If $\alpha_i > 0$, x_i is called a support vector. C is a regulation parameter used to trade-off the training accuracy and the model complexity so that a superior generalization capability can be achieved. K is a kernel function, which is used to measure the similarity between two samples. Different choices of kernel functions have been proposed and extensively used in the past and the most popular are the gaussian radial basis function (RBF), polynomial of a given degree, and multi-layer perceptron. These kernels are in general used, independently of the problem, for both discrete and continuous data. In the recent work the support vector machine is used to carry the watermark points.

3.3 Watermarking Algorithm: Generation of Watermark

The polar coordinates of each point are (r, Θ) have decimal and integer part, the integer part will be used to generate watermarking by using the random table and own-value as a following equation 7 [17]:

$$w = own\_value - \frac{key}{1000} \tag{7}$$

Where own_value will be produced by using ASCII code and key will generate by changing the integer part using random table 1.

Table 1. Real and changed values

Real value	0	1	2	3	4	5	6	7	8	9
Changed value	6	13	11	1	16	19	7	5	8	4

An example for converting own word (HKD) to get own_value by using ASCII code standard. H: 1001000(71), K: 1001011(74), and D: 1000100(67). Eventually, own_value becomes 212(71+74+67). This own_value becomes essential value when watermarks are embedded and extracted. Another example for finding the key, when the integer part of coordinate is 13 then the key will be 131 where the real value of 1 will be 13 and 3 will be 1 [17].

Embedding Watermark. The change in CAD and GIS data does not effect if it be after the sixth decimal digit so the watermark will be embed in the sixth digit or more. The proposed approach is described in the algorithm (1):

Watermarking Extraction. Where the proposed method is blind then it doesn't use the original data will follow the same steps of embedding process to verify the watermark existing. Algorithm (2) shows the details of the extracting watermarking.

Algorithm 1. The watermark embedding algorithm

1: Input: The original map vertices in Cartesian coordinates (x,y)
2: Input: change coordinates to the center point of map(x_0,y_0)
3: For each point cartesian coordinates (x,y) will be changed into polar (r, Θ)
4: sorting polar coordinates (r, Θ)
5: applying SVM on the polar coordinates r, Θ separately and find support vectors index for each coordinate
6: For each point every index of each coordinates we find cartesian coordinates(i.e watermark portable (x,y)) and then change into polar (r, Θ)
7: For each point polar coordinates (r, Θ) value will multiply by 10^s (strength of watermark $s >= 6$)
8: For each (r, Θ) decimal part will be changed by watermark and then marked polar ($\acute{r}$, $\acute{\Theta}$) coordinates will be produced.
9: For each ($\acute{r}$, $\acute{\Theta}$) will be changed into Cartesian another time to be ($\acute{x}$, $\acute{y}$);
10: Output: The marked map in Cartesian coordinates ($\acute{x}$, $\acute{y}$)

Algorithm 2. The watermark extracting algorithm

1: Input: The watermarked map map vertices in Cartesian coordinates ($\acute{x}$,$\acute{y}$)
2: Input: change coordinates to the center point of map(($\acute{x_0}$,$\acute{y_0}$))
3: For each point cartesian coordinates ($\acute{x}$,$\acute{y}$) will be changed into polar ($\acute{r}$,$\acute{\Theta}$)
4: sorting polar coordinates (r, Θ)
5: applying SVM on the polar coordinates r, Θ separately and find support vectors index for each coordinate
6: Input : The marked map vertices in Cartesian coordinates ($\acute{x}$,$\acute{y}$)
7: For each point Cartesian coordinates ($\acute{x}$,$\acute{y}$) will be changed into polar ($\acute{r}$,$\acute{\Theta}$)
8: For each point polar coordinates ($\acute{r}$,$\acute{\Theta}$) value will multiply by 10^s (strength of watermark $s >= 6$)
9: For each ($\acute{r}$,$\acute{\Theta}$) decimal part will verify that the watermark is existing.

4 Watermark Verification

4.1 Translation Attack Verification

Given a vertex in a map with Cartesian coordinates (x,y), a translation by Δx and Δy in the x and y axes, respectively, leads to new coordinates ($\acute{x}$, $\acute{y}$) = (x+ Δ x, y+ Δ y). The corresponding new polar coordinates ($\acute{r}$, $\acute{\Theta}$) become:

$$\acute{r} = \sqrt{((x+\Delta x)-(x_0+\Delta x))^2 + ((y+\Delta y)-(y_0+\Delta y))^2} = r \tag{8}$$

$$\acute{\Theta} = \arctan \frac{((y+\Delta y)-(y_0+\Delta y))}{((x+\Delta x)-(x_0+\Delta x))} = \Theta \tag{9}$$

We can get $\acute{r}$ = r and $\acute{\Theta} = \Theta$, that means the angular and radial coordinates did not change. Due to the fixation of these coordinates' values we will found no change in watermark.

4.2 Rotation Attack Verification

Given a vertex with coordinates (x, y) in the Cartesian coordinate system, after rotating it by an angle α , the new Cartesian coordinates $(\acute{x}, \acute{y})$ as in the following equations 10 and 11:

$$\acute{x} = x * \cos(\alpha) + y * \sin(\alpha) \tag{10}$$

$$\acute{y} = y * \cos(\alpha) - x * \sin(\alpha) \tag{11}$$

After applying a polar transformation to $(\acute{x}, \acute{y})$, the radial coordinate becomes as in the equation 12:

$$\acute{r} = \sqrt{(x - x_0)^2 + (y - y_0)^2} \tag{12}$$

Where (x_0, y_0) is the map center. We can get $\acute{r} = r$, which means the radial coordinate, is not changed. So the proposed method is invariant to rotation.

4.3 Scaling Attack Verification

Any vertex with coordinates (x, y), after a scaling attack by a same factor of $\mathbf{S}$ Cartesian coordinates will be $(\acute{x}, \acute{y})$ as in equations 13 and 14.

$$\acute{x} = x * \mathbf{S} \tag{13}$$

$$\acute{y} = y * \mathbf{S} \tag{14}$$

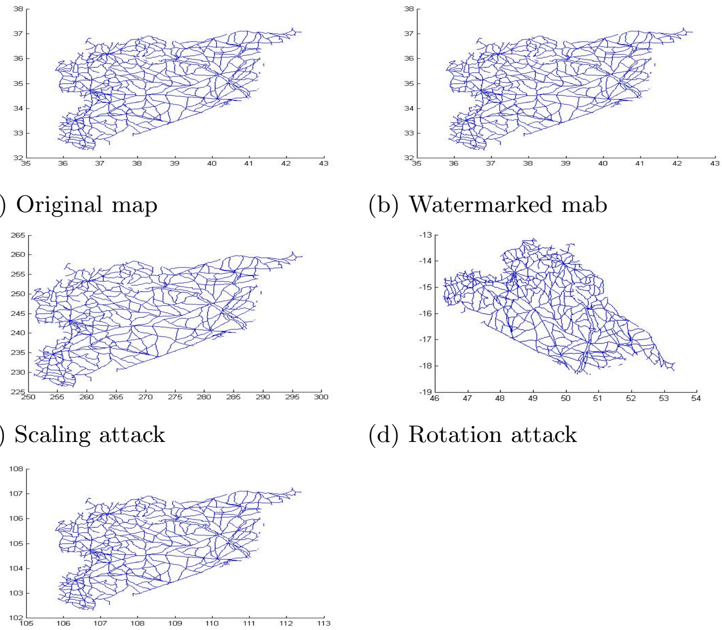
By changing into polar coordinates, we get the equation 15:

$$\acute{\Theta} = \arctan \frac{((y * \mathbf{S}) - (y_0 * \mathbf{S}))}{((x * \mathbf{S} - (x_0 * \mathbf{S}))} = \Theta. \tag{15}$$

Since $\acute{\Theta} = \Theta$, the angles before and after this attack are the same. As indicated in previous equation, it shows that the proposed method is invariant to scaling. Now it's proved that the proposed method is robust against this attacks.

5 Experimental Results

Our experimental result had been implemented with windows 7 professional (64 bit) and matlab 7.9.0 and the real data from DIVA-GIS site with different shape files. Now the original Syria and watermarked map showed at figure 5(a), 5(b), respectively followed by scaling, rotation, and translation attacked map in the figure 5(c), figure 5(d) and figure 5(e) respectively.



(a) Original map (b) Watermarked mab

(c) Scaling attack (d) Rotation attack

(e) Translation attack

Fig. 5. Syria map watermarking and attack

5.1 Analysis of Visibility

In the experiments, the parameters were chosen as follows: embedding strength $s = 6$, own_value is the corresponding ASCII standard code of word (Mourad), and the table 1. Here, the root mean square error (RMSE) as formulated in equations 16 and 17, are used to represent the distortion inflicted on the graphic by our watermarking approach.

$$RMSE = \frac{\sum \|V - V_w\|}{N - 1} \tag{16}$$

where

$$V = \sqrt{x^2 + y^2} \tag{17}$$

Where is N is the number of vertices, $\mathbf{V}$ and V_w are the original and marked vertex respectively. Tables 2, 3 expresses the relation between RMSE and the strength of watermarking of four maps Roma, Egypt,China and Syria without(i.e watermarked all vertices [23]) and with using SVM respectively and if we get Egypt roads as example we found that at strength s=8 when we didn't use SVM the RSME was 3.78E-09 from table (II) but when we improve this approach by using SVM we found that the RMSE was 1.48E-12 and that prove the superior of using SVM in the imperceptibility issue.

Table 2. Relation between watermark strength s and $RMSE$ without using SVM

Map	No. vertices	s=5	s=6	s=7	s=8	Execution time
$Roma\_roads$	31986	4.024E-06	4.5358E-07	3.3736E-08	3.3922E-09	35.704692
$Egypt\_roads$	30022	3.61E-06	3.57E-07	4.11E-08	3.78E-09	97.259982
$China\_roads$	389442	6.67E-06	6.61E-07	6.64E-08	6.61E-09	906.244648
$Syria\_roads$	11828	4.15E-06	3.27E-07	4.85E-08	4.61E-09	88.073596
$Tunesia\_roads$	6533	3.9560e-006	3.9884e-007	4.0086e-008	4.0791e-009	15.977018
$Turky\_roads$	70184	4.6485e-006	4.6450e-007	4.6771e-008	4.6767e-009	203.980169

Table 3. Relation between watermark strength s and $RMSE$ by using SVM

Map	No. vertices	s=5	s=6	s=7	s=8	Execution time
$Roma\_roads$	31986	3.0158E-09	1.2012E-10	8.3363E-12	7.3786E-13	7.709183
$Egypt\_roads$	30022	1.03E-09	1.44E-10	1.44E-11	1.48E-12	6.126416
$China\_roads$	389442	6.27E-10	6.24E-11	7.47E-12	6.87E-13	461.154683
$Syria\_roads$	11828	4.63E-09	3.91E-10	3.522E-11	3.70E-12	7.181789
$Tunisia\_roads$	6533	1.6549e-009	3.5108e-010	1.6398e-011	1.6719e-012	7.093668
$Turky\_roads$	70184	5.2360e-010	3.7549e-011	3.8138e-012	5.4038e-013	16.556669

The experimental result show there is a negative relation between the number of watermarked vertices and the imperceptibility measurement (RMSE), figure 6, Table 4 illustrate the relation between the number of vertices and the RMSE at watermark strength $s = 8$, ex: China-roads map when the number of watermarked vertices 26, 44, 58, and 75. The RMSE is $4.5695E - 13$, $7.2842E - 13$, $9.429E - 13$ and $1.53E - 12$ respectively that show that there is direct relation between the RMSE when the number of vertices increase that make an increasing in the RMSE. Also the experimental gave better execution time with SVM than without it for example, the time of embedding watermark in $China\_roads$ map at strength $s = 8$ without using SVM gave 906.244648 seconds, where when uses SVM it was 461.154683 and this verify that the using SVM in the technique reduce the execution time.

5.2 Robustness Analysis

The result of applying geometrical attacks (i.e translation, rotation, and scaling) on roads maps of Egypt, Syria, Roma and China, illustrate the power of robustness of this proposed approach and that proved by calculating the **NC** between the watermarked and original maps after the attacks. The normal correlation **NC** calculate by the following equation 18:

$$NC = \frac{\sum W \times \acute{W}}{\sqrt{\sum W^2 + \sum \acute{W}^2}} \tag{18}$$

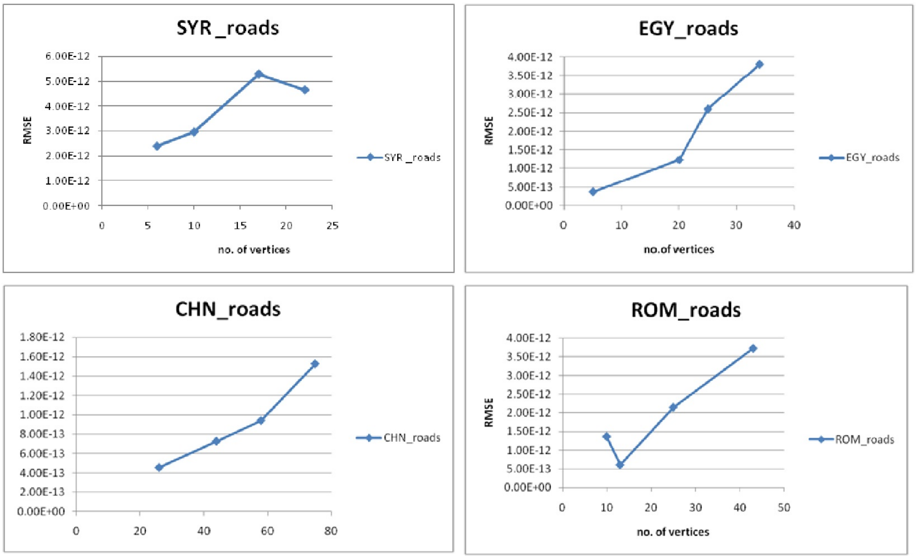


Fig. 6. Relation between Number of watermarked vertices and $RMSE$

Table 4. Relation between number of watermarked vertices and $RMSE$ without using SVM

	$no.of vertices$	$RMSE$
$Roma\_roads$	10	1.3644E-12
	13	6.0411E-13
	25	2.1503E-12
	43	3.7301E-12
$Egypt\_roads$	5	3.7064E-13
	20	1.2282E-12
	25	2.5982E-12
	34	3.7999E-12
$China\_roads$	26	4.5695E-13
	44	7.2842E-13
	58	9.429E-13
	75	1.53E-12
$Syria\_roads$	6	2.3896E-12
	10	2.9595E-12
	17	5.2718E-12
	22	4.6366E-12

Table 5. Relation between NC and versus attacks where S, R and T are Scaling, Rotation and Translation attack

Map	No.of vertices	S attack	R attack	T attack
$Roma_roads$	31986	1.00E+00	1.00E+00	1.00E+00
$Egypt\_roads$	30022	1.00E+00	1.00E+00	1.00E+00
$China\_roads$	389442	1.00E+00	1.00E+00	1.00E+00
$Syria\_roads$	11828	1.00E+00	1.00E+00	$1.00E + 00$
$Turky\_roads$	70184	1.00E+00	1.00E+00	$1.00E + 00$
$Tunisia\_roads$	6533	1.00E+00	1.00E+00	$1.00E + 00$

Table 5 illustrates the power of robustness of the proposed approach with different maps with versus attack.

6 Conclusions

This paper presents blind and robust watermark approach for map authentication. It uses support vector machine and polar coordinates to embed watermark in its decimal value by using random table as a key. The experimental results show that the machine learning (SVM) help to increase the imperceptibility without change the robustness of watermark. this proposed method is robust against geometrical attack such equal scaling, rotation and translation and that can verify the authentication of this geospatial data where the good robustness guarantee that the watermark survive also after attack that can safe the ownership of data.

References

1. Shao, C., Wang, X., Xu, X., Niu, X.: Study on lossless data hiding algorithm for digital vector maps. Journal of Image and Graphics 12(2), 206–211 (2007)
2. Peng, F., Lei, Y., Long, M., Sun, X.: A reversible watermarking scheme for two-dimensional CAD engineering graphics based on improved difference expansion. Computer-Aided Design 43(8), 1018–1024 (2011)
3. Anusudha, K., Sangeetha, A.: Digital Watermarking of Satellite Images Using Secured Spread Spectrum Technique. International Journal of Recent Trends in Engineering 1(1) (May 2009)
4. Wolthusen, S.D.: Proceedings of the IEEE Workshop on Information Assurance (2006)
5. Celik, M., Sharma, G., Tekalp, M.: Lossless generalized-LSB data embedding. IEEE Transactions on Image Processing 14(2), 253–266 (2005)
6. Gaurav, B., Jonathan Wu, Q.M., Balasubramanian, R.: New robust adjustable logo watermarking scheme. Computers and Security 3(1), 40–58 (2012)

7. Nana, W., Chaoguang, M.: Reversible fragile watermarking for 2-D vector map authentication with localization. Computer-Aided Design 44, 320–330 (2012)
8. Suk-Hwan, L., Ki-Ryong, K.: A watermarking for 3D mesh using the patch CEGIs. Digital Signal Processing 17, 396–413 (2007)
9. Fei, P., Re-Si, G., Chang-Tsun, L., Min, L.: A semi-fragile watermarking algorithm for authenticating 2D CAD engineering graphics based on log-polar transformation. Computer-Aided Design 42, 1207–1216 (2010)
10. Kitamura, I., Kanai, S., Kishinami, T.: Copyright protection of vector map using digital watermarking method based on discrete Fourier transform. In: IEEE Geoscience and Remote Sensing Symposium (IGARSS 2001), vol. 3, pp. 1191–1193 (2001)
11. Kang, H., Kim Kab, I., Choi, J.: A map data watermarking using the generalized square mask. In: Proceedings of IEEE International Symposium on Industrial Electronics (ISIE 2001), vol 3, pp. 1956-1958 (2001)
12. Solachidis, N., Nikolaidis, I.P.: Fourier Descriptors Watermarking of Vector Graphics Images. In: Proceedings of IEEE International Conference on Image Processing, vol. 3, pp. 9–12 (2001)
13. Solachidis, V., Pitas, I.: Watermarking polygonal lines using Fourier descriptors. IEEE Computer Graphics and Applications 24(3), 44–51 (2004)
14. Ohbuchi, R., Ueda, H., Endoh, S.: Watermarking 2D vector maps in the mesh-spectral domain. In: Proceeding of Shape Modeling International, pp. 216–225 (2003)
15. Yuanyuan, L., Luping, X.: A blind watermarking of vector graphics images. In: Proceedings of Fifth International Conference on Computational Intelligence and Multimedia Applications (ICCIMA 2003), pp. 424–429 (2003)
16. Zhang, W., Ren, X., Zhang, G.L.: An robust object locating method based on log polar transform and affine transform. Journal of Image and Graphics 11(9), 1255–1259 (2006)
17. Jungyeop, K.: Robust Vector Digital Watermarking Using Angles and a Random Table. Advances in Information Sciences and Service Sciences 2(4) (December 2010)
18. Syed, F.T., Asifullah, K.A., Anwar, M.M.: Support Vector Machine based Intelligent Watermark Decoding for Anticipated Attack. World Academy of Science, Engineering and Technology 21, 1001–1006 (2008)
19. Chapelle, O., Vapnik, V., Bousquet, O., Mukherjee, S.: Choosing Multiple Parameters for Support Vector Machines. Machine Learning 46(1-3), 131–159 (2002)
20. Hsu, C.W., Chang, C.C., Lin, C.J.: A practical guide to support vector machines. Technical report, Department of Computer Science and Information Engineering, National Taiwan University (2003)
21. Staelin, C.: Parameter selection for support vector machines. Technical report, HP Labs, Israel (2002)
22. Moghaddam, B., Yang, M.H.: Learning Gender with support faces. IEEE Transaction on Pattern Analysis and Machine Learning 24 (2002)
23. Mourad, R., Mouhamed, A.M., Hassanien, A.: Blind 2D Vector Data Watermarking Approach Using Random Table and Polar Coordinates. In: International Conference on Uncertainty Reasoning and Knowledge Engineering, pp. 67–70 (2012)
24. Wu, Q., Zhou, D.-X.: Analysis of support vector machine classification. Journal of Computer Analysis and Applications 8, 99–119 (2006)
25. Oracle® Spatial Developer's Guide, 11g Release 2 (11.2) E11830-11