Thesis for the Degree of Master of Engineering

A Watermarking Scheme Using Polyline and Polygon Characteristics of Shapefile in the GIS Digital Map

by

Xiao Jiao Huo

Department of Information Engineering

The Graduate School

Pukyong National University

February 2011

A Watermarking Scheme Using Polyline and Polygon Characteristics of Shapefile in the GIS Digital Map

GIS 디지털 지도상에서 Shapefile의 폴리곤 및 폴리라인 특성을 이용한 워터마킹 기법

Advisor: Prof. Ki-Ryong Kwon

by

Xiao Jiao Huo

A thesis submitted in partial fulfillment of the requirements
for the degree of Master of Engineering

in Department of Information Engineering, The Graduate School,
Pukyong National University

February 2011

A Watermarking Scheme Using Polyline and Polygon Characteristics of Shapefile in the GIS Digital Map

A dissertation
by
Xiao Jiao Huo

Approved by:

(Chairman) Bong-Kee Sin

(Member) Sang Uk Shin

(Member) Ki-Ryong Kwon

February 25, 2011

Contents

Table Contents

Figure Contents

GIS 디지털 지도상에서 Shapefile의 폴리곤 및 폴리라인 특성을 이용한 워터마킹 기법

Xiao Jiao Huo

부 경 대 학 교 대 학 원 정 보 공 학 과

요 약

오늘날 우리는 컴퓨팅 기술과 네트워킹 기술의 급속한 성장이 이루어지고 있는 21세기의 초입에 접어들고 있다. 이러한 기술들은 시간이 지날수록 더욱 더 급격히 진보하고 있으며 해당 기술들의 빠른 진보는 과거와 비교하여 엄청난 속도로 디지털 데이터들을 생성, 조작, 저장 및 사용을 가능하게 한다. 그 중, 지리공간 데이터는 기술 발달로 인한 사용의 빈도 및 용도가 다양해진 매우 대표적인 콘텐츠로서 오늘날 모바일 및 향상된 인터넷 서비스 기술과 연동하여 다양한 형태로 생성, 가공, 저장 및 이용되고 있다. 지리 정보 및 위치 정보의 필요성이 갈수록 증가함에 따라, GIS(geographic information system)은 도시계획, 광역단위 기반 각종 편의시설 관리, 자연자원 및 환경 감시, 지표지역 특성 요약 등 다양한 곳에서 널리 활용되고 있다.

대다수 GIS 기반 지도들은 지리 정보를 보다 상세히, 쉽게 표현하기 위하여 벡터 데이터 형태로 나타낸다. 그러나 GIS 벡터 지도 역시 디지타이징 된 데이터로서 일반적인 디지털 콘텐츠들과 같이 불법적으로 쉽게 복사, 편집, 변형되는 단점을 가지고 있다. 하지만 이미지나 비디오, 음원과 같은 콘텐츠들에 비해 GIS 벡터 데이터에 대한 정보보호는 많이 취약한 편이다. 워터마킹 기법은 GIS 데이터의 불법적 도용 및 배포를 막기 위한 좋은 정보보호 기법으로 활용할 수 있다.

따라서 본 논문에서는 벡터 데이터의 기본성분인 폴리라인/폴리곤 성분 특성을 이용하여 GIS 벡터 지도에 적합한 강인한 워터마킹 기법을 제안한다. 제안 기법은 워터마크 검출을 위하여 원본을 사용하지 않으며 워터마크 삽입시 삽입 벡터 맵의 비가시성도 고려하였다. 제안 기법에서는 그룹으로 묶인 폴리라인/폴리곤의 길이 정보/지름 정보를 이용하여 워터마킹을 수행한다. 그리고 워터마크 비트의 값에 따라 폴리라인/폴리곤들을 적절한 그룹으로 묶어 각 성분들의 길이/지름 값들의 국부적 평균값을 변경한다. 또한, 워터마크의 보안성 및 강인성 확보를 위하여 PRNS(pseudo-random number sequence)를 이용하여 워터마크를 암호화 한 후 비트스트림의 길이를 증가하여 대상 맵에 삽입한다.

실험결과를 통하여 다양한 공격들에 대하여 제안 기법의 비가시성, 보안성 및 강인성이 보장됨을 확인하였고 원본없이 워터마크 검출이 가능함을 보였다.

A Watermarking Scheme Using Polyline and Polygon Characteristics of Shapefile in the GIS Digital Map

Xiao Jiao Huo

Department of Information Engineering, The Graduate School,
Pukyong National University

Abstract

Today, information systems are undergoing much improvement and they are able to create, manipulate, store and use spatial data much faster and at a rapid rate as compared to conventional methods. As the necessity of the geographic information and the position information has been augmented, the geographic information system (GIS) has been widely used in the city planning, utilities management, natural resource environment, land surveying and so on. Most GIS map use the vector data for representing geographic information more detail and easier. But GIS vector map can be copied easily, edited and diffused illegally like as most of digital data. Therefore the watermarking scheme has become necessary to prevent plagiarizing cases of GIS data.

This paper propose an invisible blind, safety and robust watermarking scheme for the copyright protection of GIS vector digital map by using the polyline/polygon geometric characteristics. In our scheme, we calculate the length/perimeter of all polylines/polygons in a map and cluster polylines/polygons to some groups. And then we embed a watermark bit by changing the local mean length/perimeter of polylines/polygons of a suitable group. For getting greater safety and robustness, we use PRNS (pseudo-random number sequence) processing the watermark and embed the watermark multiple times in all map. Our scheme has a good invisible, safety and robustness against various geometric attacks as show in experimental result, and need not the original map in the extracting process of watermark.

I . Introduction

With the digital technology and internet developing, the application of digital technology can see everywhere in our daily life, with extending human's scope of activity, the demand of geographic information is more and more. Geographic information systems (GIS) has become the necessary technique. GIS is a set of tools that captures, stores, analyzes, manages, and presents data that are linked to location(s). It is widely used in various fields; cartography, remote sensing, land surveying, public utility management, natural resource management, photogrammetry, geography, urban planning, emergency management, navigation, aerial video, and localized search engines [1].

GIS is a computer based information system used to digitally represent and analyze the geographic features present on the Earth' surface and the events (non-spatial attributes linked to the geography under study) that taking place on it. The meaning to represent digitally is to convert analog (smooth line) into a digital form. GIS digital maps can be classified into either raster-digital or vector-digital maps. A raster digital map represents a map as raster image data, i.e., an image represented by 2D array of pixels. Vector digital maps employ geometrical primitives such as points, lines, polylines, and polygons to represent objects in the maps, such as building outlines, roads, rivers, and contour lines. But since a GIS digital map is easy to be updated, duplicated, and distributed by any user, it must be protected by any scheme. Digital watermarking is a core technology and an effective method to counter such abuses.

Digital watermarking is the process of embedding information into digital multimedia content such that the information (which we call the watermark) can later be extracted or detected for a variety of purposes including copy prevention and control. Digital watermarking adds a structure called watermark to achieve that the target data object imperceptibly and inseparably. The information that is encoded to the watermark can be used to identify the copyright owner or to detect tampering [2]. Digital watermarking has become an active and important area of research, and development and commercialization of watermarking techniques is being deemed essential to help address some of the challenges faced by the rapid proliferation of digital content.

As the raster digital map represents a map as raster image data, most of the watermarking algorithm developed for digital images can be applied to the raster digital maps. Let's begin with a review of the existing representatives of the image watermarking techniques. In image processing there is two broad categories: spatial domain methods and frequency domain methods. The term spatial domain refers to the image plane itself, and approaches in this category are based on direct manipulation of pixels in an image. Frequency domain processing techniques are based on modifying the Fourier transform of an image. Corresponding, there is also two categories which are spatial domain method and frequency domain method in image watermarking techniques [3]-[4].

In a vector digital map, connectivity among objects, e.g., vertices, is specified explicitly, and the distance between vertices are irregular. Consequently, standard techniques such as Fourier or wavelet transformation used in image watermarking algorithms can not be applied readily. Probably due to these difficulties, we know of only a few published works on watermarking vector digital map such as Ohbuchi's [5], Kim's [6]-[7] and Chang's [8]-[9] watermarking algorithm.

This paper propose an invisible blind and robust watermarking scheme for the copyright protection of GIS vector digital map by using the polyline/polygon geometric characteristics. The proposed scheme is based on the polyline/polygon type data of Environmental Systems Research Institute (ESRI) shapefile, which is a popular geospatial vector data format for GIS. In our scheme, we calculate the length/perimeter of all polylines/polygons in a map and cluster polylines/polygons to some groups with the uniform step of length/perimeter dynamic range. After that we input watermarking message and use PRNS (pseudo-random number sequence) processing the watermark bit to improved security. And then we embed a processed watermark bit by changing the local mean length/perimeter of polylines/polygons of a suitable group. For getting greater robustness, we also embed the watermark bits multiple times in all the map. In embedding process of watermark, we produce some keys that are needed in the extracting process of watermark such as PRNS. Finally, we use saved keys to extract watermark bits in the subgroups and use a checkout method to get the optimized watermark message. Our scheme need not original map in the extracting process of watermark while satisfy invisibility [5]-[9] and [14]-[27].

From experimental results, we confirmed that the artifacts by the embedded watermark cannot be observed by users, and the watermark was not damaged or detected by other designers. And the watermark has the robustness against geometric attacks like as translation, rotation, random noise, insert and delete vertex, scrambling of order of geometric primitives in a data file, and cropping.

The rest of the paper is structured as follows. In the next part, we introduce GIS data system and structure. In part Ⅲ, we describe the conventional watermarking algorithm of vector digital map. Then we present proposed watermarking algorithm in part Ⅳ. After that we show the experimental results in part Ⅴ. In part Ⅵ, it is a summary and conclusion.

Ⅱ. GIS Data System and Structure

Geographic Information System (GIS) is a system of hardware, software and procedures to facilitate the management, manipulation, analysis, modelling, representation and display of georeferenced data to solve complex problems regarding planning and management of resources. GIS have emerged in the last decade as an essential tool for urban and resource planning and management. Their capacity to store, retrieve, analyse, model and map large areas with huge volumes of spatial data has led to an extraordinary proliferation of applications. Geographic information systems are now used for land use planning, utilities management, ecosystems modeling, landscape assessment and planning, transportation and infrastructure planning, marker analysis, visual impact analysis, facilities management, tax assessment, real estate analysis and many other applications.

2.1 GIS Data Components

GIS data consists of spatial data and attribute components. In spatial component, the observations have two aspects in its localisation: absolute localisation based in a coordinates system and topological relationship referred to other observations. Example: The Department of Geomatics is located at the particular coordinate X, Y, or, The Department is located between Grattan Street and Old Engineering Building. A GIS is able to manage both while computer assisted cartography packages only manage the absolute one [10]. In attribute component, the variables or attributes can be studied considering the thematic aspect (statistics), the locational aspect (spatial analysis) or both. For example, a polygon represent a lake, the attribute of the lake may contain lake's perimeter, lake's area, lake's depth, water quality or pollution level.

2.2 Raster Data and Vector Data

Spatial data include two data types: Raster data and Vector data. In raster type data, a set of cells located by coordinate is used; each cell is independently addressed with the value of an attribute. Each cell contains a single value and every location corresponds to a cell. One set of cell and associated value is a LAYER. Raster models are simple with which spatial analysis is easier and faster. Raster data models require a huge volume of data to be stored, fitness of data is limited by cell size and output is less beautiful [11]. Vector data is defined by the vectorial representation of its geographic data. It is represented in the form of coordinates. In vector data, the basic units of spatial information are points, lines (arcs) and polygons. Each of these units is composed simply as a series of one or more coordinate points, for example, a line is a collection of related points, and a polygon is a collection of related lines.

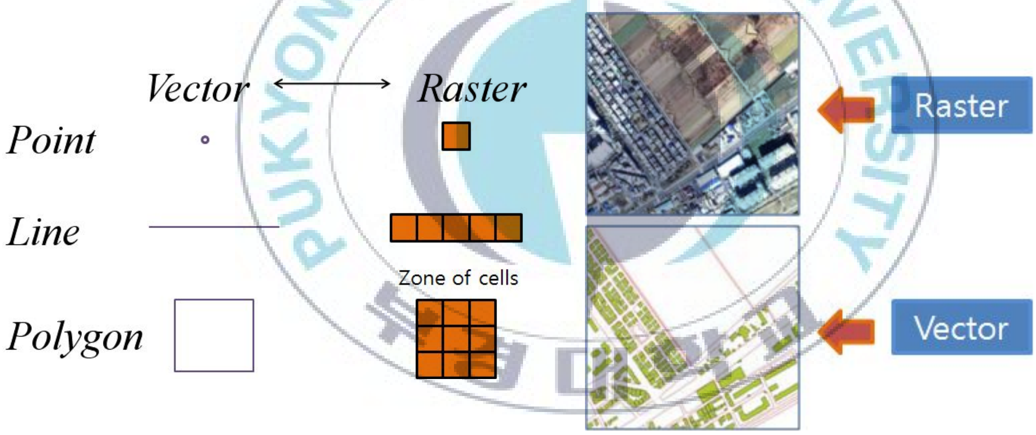


Figure 1. Vector and Raster data represent Real World Map.

Figure 1 shows vector and raster data representation of the real world phenomena. Table 1 shows some main GIS data formats about vector data and raster data.

Table 1. GIS data formats about vector data and raster data.

Format	Format Content
Shapefile	· A geospatial vector data format for geographic information systems software. · Stores nontopological geometry and attribute information for the spatial features in a data set. · Require less disk space and are easier to read and write. · Support point, line, and area features.
TIGER	· Topologically Integrated Geographic Encoding and Referencing. · Used by the United States Census Bureau. · The most well-integrated and automated geographic database.
DXF	· Drawing Exchange Format or Drawing Interchange Format. · Created for the purpose of exchanging data in AutoCAD. · Save as binary data format and ASCII data types.
TIFF	· Tagged Image File Formats. · Saves the scanned images and reads them. · Use run length and other image compression schemes.
GIF	· Graphic Interchange Format. · A file format for image files, commonly used on the internet. · It is well-suited for images with sharp edges and relatively few gradations of color.

As Table 2 shows that the vector data has more advantage less disadvantage than raster data [12], especially, vector data allows much more analysis capability such as roads, power, rail, telecommunication, etc (Examples: Best route, largest port, airfields connected to two-lane highways) while raster data can not display all the characteristics of the features, and based on coordinate digit's vector data has more suitable by computer processing than raster data based on image. So, in the digital maps, it is usually use vector data to storage geographic information. Also the GIS watermarking scheme proposed in this paper is based on vector data.

Table 2. The advantages and disadvantages of vector and raster.

	Vector	Raster
Advantage	1. High resolution 2. Works well with boundaries 3. Explicit representation of linear features 4. Efficient storage of sparse data	1."Easy" creation from image data 2. Easy to overlay 3. Efficient storage for dense, heterogeneous data
Disadvantage	1. Manipulations require sophisticated algorithms 2. Processing can require lots of computer time 3. Inefficient storage of dense data	1. Must pre-define spatial resolution 2. Requires large amounts of storage space 3. Inefficient when data is sparse or homogeneous 4. Deals poorly with linear features

2.3 Shapefile Format

The shapefile [13] is a popular geospatial vector data format for geographic information systems software. It is developed and regulated by Environmental Systems Research Institute (ESRI) as a open specification for data interoperability among ESRI and other software products. A shapefile stores non topological geometry and attribute information for the spatial features in a data set. The geometry for a feature is stored as a shape comprising a set of vector coordinates. Shapefiles can support point, line, and area feature. Area features are represented as closed loop, double-digitized polygons. Attributes are held in a dBASE format file. Each attribute record has a one-to-one relationship with the associated shape record.

An ESRI shapefile consists of a main file, an index file, and a dBASE table. The main file is a direct access, variable-record-length file in which each record describes a shape with a list of its vectors. In the index file, each record contains the offset of the corresponding main file record from the beginning of the main file. The dBASE table contains feature attributes with on record per feature. The one-to-one relationship between geometry and attributes is based on record number. Attribute records in the dBASE file must be in the same order as records in the main file. The suffix for the main file is .shp. The suffix for the index file is .shx. And the suffix for the dBASE table is .dbf. Note that the main file, the index file, and the dBASE file have the same prefix. For example, xiaojiao.shp (main file), xiaojiao.shx (index file) and xiaojiao.dbf (dBASE file).

The main file (shp) contains a fixed-length file header followed by variable-length records. Each variable-length record is made up of a fixed-length record header followed by variable-length record contents. Figure 2 illustrates the main file organization.

File Header	
Record Header	Record Contents
Record Header	Record Contents
Record Header	Record Contents
Record Header	Record Contents
...	
...	
Record Header	Record Contents

Figure 2. Organization of the Main file.

The main file header is 100 bytes long. Figure 3 shows the field in the file header with their byte position, value, type, and byte order. In the figure, position is with respect to the start of the file.

Position	Field	Value	Type	Byte Order
Byte 0	File Code	9994	Integer	Big
Byte 4	Unused	0	Integer	Big
Byte 8	Unused	0	Integer	Big
Byte 12	Unused	0	Integer	Big
Byte 16	Unused	0	Integer	Big
Byte 20	Unused	0	Integer	Big
Byte 24	File Length	File Length	Integer	Big
Byte 28	Version	1000	Integer	Little
Byte 32	Shape Type	Shape Type	Integer	Little
Byte 36	Bounding Box	Xmin	Double	Little
Byte 44	Bounding Box	Ymin	Double	Little
Byte 52	Bounding Box	Xmax	Double	Little
Byte 60	Bounding Box	Ymax	Double	Little
Byte 68*	Bounding Box	Zmin	Double	Little
Byte 76*	Bounding Box	Zmax	Double	Little
Byte 84*	Bounding Box	Mmin	Double	Little
Byte 92*	Bounding Box	Mmax	Double	Little

* Unused, with value 0.0, if not Measured or Z type

Figure 3. Description of the Main File Header.

The record header for each record stores the record number and content length for the record. It has a fixed length of 8 bytes. Figure 4 shows the field in the record header with their byte position, value, type, and byte order. In the figure, position is with respect to the start of the record.

Position	Field	Value	Type	Byte Order
Byte 0	Record Number	Record Number	Integer	Big
Byte 4	Content Length	Content Length	Integer	Big

Figure 4. Description of Main File Record Header.

The record contents of a shape type followed by the geometric data for the shape. The length of the record contents depends on the number of parts and vertices in a shape. Figure 5 shows the possible shape types.

Value	Shape Type
0	Null Shape
1	Point
3	PolyLine
5	Polygon
8	MultiPoint
11	PointZ
13	PolyLineZ
15	PolygonZ
18	MultiPointZ
21	PointM
23	PolyLineM
25	PolygonM
28	MultiPointM
31	MultiPatch

Figure 5. The shape type and the corresponding value.

The index file (.shx) contains a 100-bytes header followed by 8-bytes, fixed-length records. Figure 6 shows the index file organization.

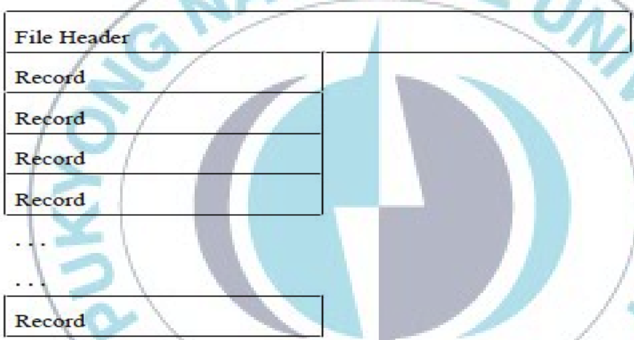


Figure 6. Organization of the index file.

The index file header is identical in organization to the main file header described above. The file length stored in the index file headers is the total length of the index file in 16-bits words.

The i'th record in the index file stores the offset and content length for the i'th record in the main file. Figure 7 shows the fields in the file header with their byte position, value, type and byte order. In the table, position is with respect to the start of the index file record.

Position	Field	Value	Type	Byte Order
Byte 0	Offset	Offset	Integer	Big
Byte 4	Content Length	Content Length	Integer	Big

Figure 7. Description of Index Records.

The offset of a record in the main file is the number of 16-bit words from the start of the main file to the first byte of the record header for the record. Thus, the offset for the first record in the main file is 50, given the 100-byte header.

The dBASE file (.dbf) contains any desired feature attributes or attribute keys to which other tables can be joined. Its format is a standard DBF file used by many table-based application in Windows. Any set of fields can be present in the table. The table contain one record per shape feature and the record order is as same as the order of shape in the main file (.shp).

Because shapfiles do not have the processing overhead of a topological data structure, they have advantages over other data sources such as faster drawing speed and edit ability. Shapfiles handle single features that overlap or that are noncontiguous. They also typically require less disk space and are easier to read and write. And there is so many softwares support shapefile format such as ARC/INFO, PC ARC/INFO, Spatial Database Engine (SDE), ArcViewGIS and BusinessMap. Also, we can directly create or modify shapefile by creating a program according to the shapefile specifications. So that, we choose shapefile format to do our research about embedding watermarking in digital vector maps.

Ⅲ. Conventional Watermarking Algorithm of Vector Digital Map

In this section, in order to facilitate understanding the proposed scheme, we will introduce the method of Ohbuchi, Kim and Chang respectively.

3.1 Ohbuchi's Watermarking Algorithm

Ohbuchi. ect [5] proposed an earliest scheme that embedding watermarks for vector digital maps. This scheme divide the vector map to many rectangles in order to improve robustness in geometric variation like as translation, cropping. A watermark bit is embedded by displacing an average of coordinates of a set of vertices that lies in a rectangular area created on a map by adaptively subdividing the map. The watermark embedding using PRNS (pseudo-random number sequence) improved security. The watermark extracting need original map and watermarked map.

Ohbuchi scheme describe three area subdivision methods; (1) *Uniform* (UNIF), (2) *Quadtree* (QUAD), and (3) *Modified quadtree* (MQUAD). Each method subdivides a given map as described below.

· *Uniform* (UNIF): Subdivide the map into $k \times l$ uniform size rectangular sub-areas.

· *Quadtree* (QUAD): Subdivide the map adaptively according to the area-quadtree algorithm so that every rectangle contain more than d vertices.

· *Modified quadtree* (MQUAD): Subdivide the map adaptively as with the QUAD method. If a subdivision created a rectangle containing less than d vertices, the rectangle is merged with an adjacent rectangle at the same subdivision level. If there were more than one candidate for the pairing, the one with the smallest number of vertices is chosen.

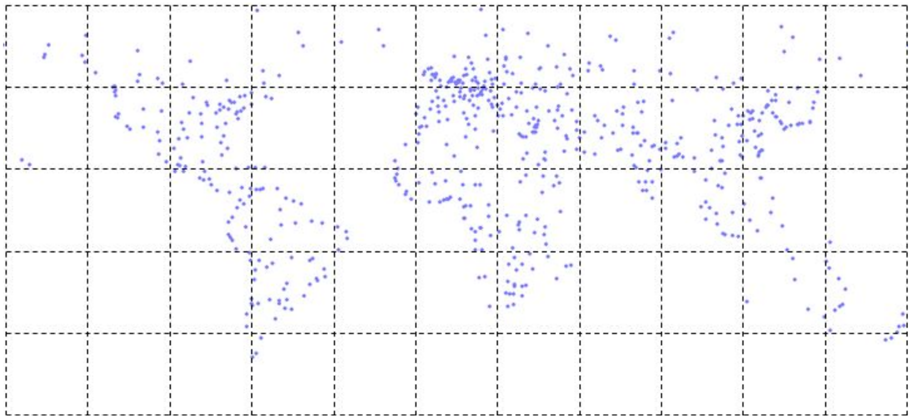


Figure 8. An example of Uniform (UNIF) decomposition.

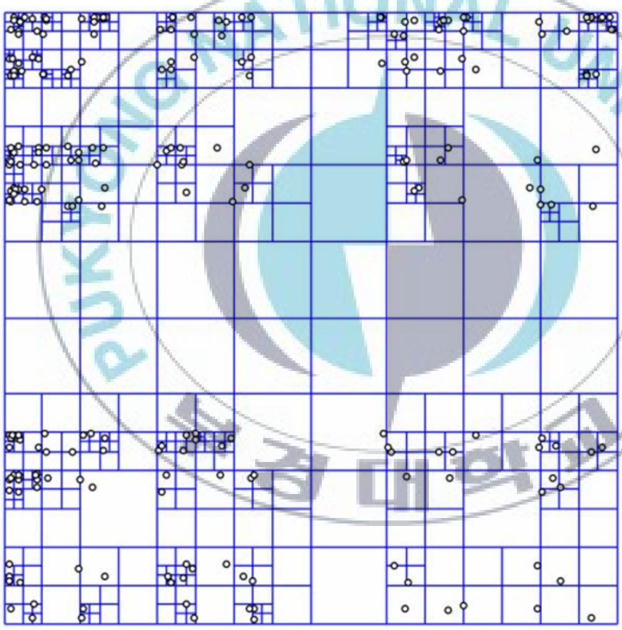


Figure 9. An example of Quadtree(QUAD) decomposition.

As mentioned, the original map is divided to $I(1<i<I)$ rectangles by using UNIF, QUAD or MQUAD. After the subdivision, the data to be

embedded is an m-dimensional bit vector $\mathbf{a}=(a_1, a_2, \ldots, a_m)$ in which each bit $a_j \in \{0,1\}$, each bit a_j is spread spatially over the map by duplicating each symbol by *chip rate* c, producing a watermark symbol vector $\mathbf{b}=(b_1, b_2, \ldots, b_{mc})$, $b_i \in \{0,1\}$ of length $m \cdot c$. Finally, the bit vector b_i is converted to an embedding symbol vector $\mathbf{b}' = (b_1', b_2', \ldots, b_{mc}')$ $b_i' \in \{-1,1\}$.

$$b_i' = \begin{cases} -1, & \text{if } b_i = 0 \\ 1, & \text{if } b_i = 1 \end{cases} \tag{1}$$

The coordinate of the vertex after watermarking is computed by the following formula:

$$\hat{v}_{i,m} = v_{i,m} + b_i' \cdot p_i \cdot \alpha, \ (1 < m < M) \tag{2}$$

Here, i indicates ith rectangle that contains M vertices. $v_{i,m}$ and $\hat{v}_{i,m}$ are the coordinate of mth vertices $(1 < m < M)$ in the ith rectangle prior to and after watermarking, respectively. $P_i \in \{-1,1\}$ be the *pseudo-random number sequence* (PRNS) generated from a known key k_w, the extraction requires the key k_w used for the embedding. α $(\alpha > 0)$ be the modulation amplitude.

The extraction requires original map and key k_w. The watermark a_j can be easy obtained through the formula (3)~(6).

$$\overline{\tilde{v}}_i = \sum_{k=1}^{h_i} \tilde{v}_i \ , \quad \bar{v}_i = \sum_{k=1}^{h_i} v_i \tag{3}$$

Here, $\bar{v}_i$ be the average of coordinates of vertices in the ith rectangle prior to the watermarking, $\overline{\tilde{v}}_i$ be the average of coordinates of vertices in the ith rectangle in watermarked map.

$$q_j = \sum_{i=jc}^{(j+1)c-1} \left(\overline{\tilde{v}}_i - \bar{v}_i \right) \cdot p_i = \sum_{i=jc}^{(j+1)c-1} b_i' \cdot \alpha \cdot p_i^2 \tag{4}$$

$$q_j = c \cdot \alpha \cdot b_i^{'} \tag{5}$$

where q_j takes one of the two values $\{-ac, ac\}$.

$$a_j = sign(q_j) \tag{6}$$

The string a_j can easily be converted to the original message bit sequence b_i by applying an inverse of the mapping as the embedding.

Ohbuchi. ect method use PRNS and repeatedly embed the watermarks to improve robustness against various attack. The method has a shortcoming that need original map when extracting the watermark. And the method has the weak robustness against attacks of noise addition, translation and so on.

3.2 Kim's Watermarking Algorithm

Kim [6]-[7] propose a copyright protection of GIS vector map based on polyline of DXF file format. This scheme select the embedding target layer of polyline and generate polyline data group using the length of connected lines. And then embed each watermark bit into the line length of polyline data by changing all coordinates of vertices according to the constraint of the robustness and the invisibility.

Firstly, Kim selects randomly any layer among layers that the number of polyline component is above M times the watermark bit number N_w and determines this layer I_i as the target layer.

Thus, the target layer I_i has the polyline data L_i.

$$L_i = \left\{ L_{i,k} | k \in [1, N_{L_i}] \right\},\ N_{L_i} > N_w \times M. \tag{7}$$

A polyline component $L_{i,k} = v_{ik,j} | j \in [1, N_{ik}]$ has N_{ik} vertices connected each other and $v_{ik,1} \neq v_{ik,N_{ik}}$.

Calculates a center point m of all vertices that included in polyline components $L_i = \{L_{i,k} | k \in [1, N_{L_i}]\}$ in the selected layer.

Set the reference length R_{ik} and the embedding length T_{ik}, for embedding a bit into a selected polyline component.

$$R_{ik} = (l_{ik,1} + l_{ik,N_{ik}})/2 = (\| \overrightarrow{mv_{ik,1}} \| + \| \overrightarrow{mv_{ik,N_{ik}}} \|)/2 \tag{8}$$

$$T_{ik} = \Sigma_{j=2}^{N_{ik}-1} \| \overrightarrow{mv_{ik,j}} \| / (N_{ik} - 2) \tag{9}$$

The watermark bit $w_{j \in [1, N_w]}$ is embedded into the difference between the reference length and the embedding length; if $w_j = 1$, then $R_{ik} < T_{ik}$. Otherwise, $R_{ik} > T_{ik}$.

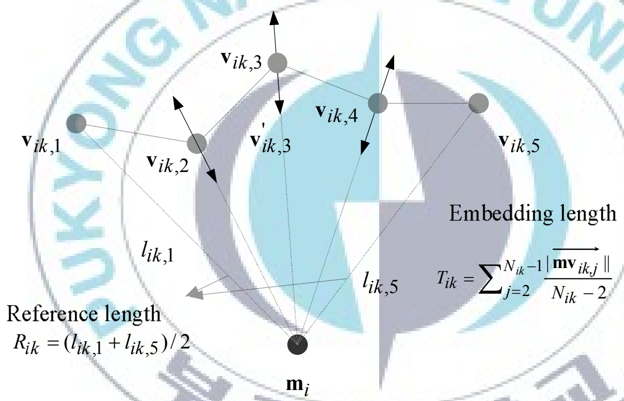


Figure 10. Embedding a watermark bit by the reference length and the embedding length in a polyline component with 5 vertices.

In order to satisfy this condition, Kim change the coordinates of vertices that included in the embedding length.

$$v'_{ik,j} = v_{ik,j} + \alpha_{ik}(1 - 2w_j)\widehat{n_{ik,j}},\ j \in [2, N_{ik} - 1] \tag{10}$$

$$\widehat{n_{ik,j}} = \overrightarrow{mv_{ik,1}} / \| \overrightarrow{mv_{ik,1}} \| \tag{11}$$

$$\alpha_{ik} = \begin{cases} \alpha, & \text{if } (w_j = 0 \,\&\, R_{ik} \le T_{ik}) \text{or} (w_j = 1 \,\&\, R_{ik} > T_{ik}) \\ \beta, & \text{if } (w_j = 0 \,\&\, R_{ik} > T_{ik}) \text{or} (w_j = 1 \,\&\, R_{ik} \le T_{ik}) \end{cases} \tag{12}$$

α is $1 < \alpha < 2$ and β is $0 < \beta < 0.5$. Embedding vertices $v_{ik,j}(j \in [2, N_{ik}-1])$ are changed by formula (10) until the embedding condition is satisfied.

The watermark extracting process was performed by using the index $F = \{j | j \in [1, N_w]\}$ of the target polyline set that are stored in the embedding process. And the watermarked polyline set $L_i^* = \{L_{i,k}^* | k \in [1, N_L]\}$ in the order of the watermark bit index.

Calculate the reference length $R_{i,k}^*$ and the embedding length T_{ik}^* in a $L_{i,k}^*$ and extract the watermark by the difference between $R_{i,k}^*$ and T_{ik}^*.

Kim scheme selects the line distribution of polyline component to the embedding target. The watermark bit is embedded by compare the reference length with embedding length improve robustness against various geometric attacks. This scheme can only processing polyline type, and need repeatedly change vertices' coordinates while relatively large dependence on points.

3.3 Chang's Watermarking Algorithm

Chang [7]-[9] proposes a blind watermarking scheme for the copyright protection of GIS vector maps by using the geometrical characteristics of the polygon based on DXF file format. The method that select the layer where watermark is embedded. And then, she calculate each surface area of polygons and MAPG (mean areas of polygon group) in the selected layer. Using the calculated MAPG, all polygons in the layer have been grouped into similar area distributions. Finally the watermark is embedded by changing each mean area distribution of the grouped polygons with transparency and robustness of the watermark.

The algorithm then groups polygon data P_i into N_G sets according to the distribution of the polygon area. In the grouping process, all areas $A_i = \{A_{i,k} | k \in [1, N_{P_i}]\}$ are obtained by using formula (13).

$$A_{i,k} = \frac{1}{2} \sum_{n=0}^{N_{P_{i,k}}} (x_{n,i,k} \cdot y_{n+1,i,k} - x_{n+1,i,k} \cdot y_{n,i,k}) \tag{13}$$

The mean area $\overline{A_i}$ of all polygons is calculated by using formula (14).

$$\overline{A_i} = \sum_{k=1}^{N_{P_i}} A_{i,k} / N_{P_i} \tag{14}$$

The value of the area A within $[0, 2 \times \overline{A_i}]$ is classified into N sections. Polygon sets $S = \{S_l | l \in [1, N]\}$ with N sections by grouping all polygon data into each section according to their area range.

$$S_l = \left\{ P_{l,k} | (l-1) \frac{2\overline{A_i}}{N} < A_{l,k} < l \frac{2\overline{A_i}}{N}, k \in [1, N_l] \right\} \tag{15}$$

Here, $P_{l,k}$ is polygon data in S_l that its area $A_{l,k}$ is within area range $[(l-1)2\overline{A}/N, l2\overline{A}/N]$ of l th. Section N_l is the number of polygon data grouped into a set S_l.

This algorithm selects randomly N_W sets among total N sets by using index function $f(j)$. Thus, the embedding target sets S^*.

$$S^* = \{S_{f(j)} | j \in [1, N_w]\} \text{ where } f(j) = l, f(j_1) \neq f(j_2), j_1, j_2 \in [1, N_w] \tag{16}$$

$$S_{f(j)} = \left\{ P_{f(j),k} | (f(j)-1) \frac{2\overline{A_i}}{N} < A_{f(j),k} < f(j) \frac{2\overline{A_i}}{N}, k \in [1, N_{f(j)}] \right\} \tag{17}$$

The index information $f(j)_{j \in [1, N_w]}$ is stored to extract the watermark.

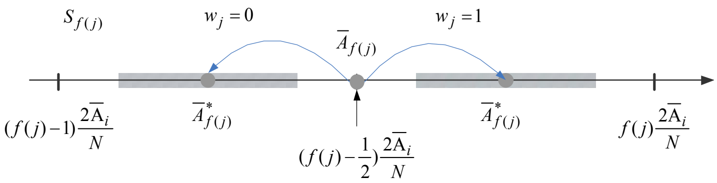


Figure 11. Embedding a watermark bit w_j into local mean $\overline{A_{f(j)}}$ of any section $S_{f(j)}$.

The watermark bit $w_{j\in[1,N_w]}$ is embedded into the local mean area $\overline{A_{f(j)}}$ of all polygon data in jth set $S_{f(j)}$ as follows:

$$\overline{A_{f(j)}} = \Sigma_{k=1}^{N_{f(j)}} A_{f(j),k} / N_{f(j)} \tag{18}$$

$$\overline{A'_{f(j)}} \in \begin{cases} [(f(j)-1)\frac{2\overline{A_i}}{N}, (f(j)-\frac{1}{2})\frac{2\overline{A_i}}{N}], & \text{if } w_j = 0 \\ [(f(j)-\frac{1}{2})\frac{2\overline{A_i}}{N}, f(j)\frac{2\overline{A_i}}{N}], & \text{if } w_j = 1 \end{cases} \tag{19}$$

Thus, the watermarked $\overline{A'_{f(j)}}$ can be calculated by formula (20).

$$A'_{f(j),k} = \frac{1}{2}\Sigma_{n=0}^{N_{P_{f(j),k}}} \alpha_{f(j),k}(x_{n,f(j),k} \cdot y_{n+1,f(j),k}) - \frac{1}{2}\Sigma_{n=0}^{N_{P_{f(j),k}}} \alpha_{f(j),k}(x_{n+1,f(j),k} \cdot y_{n,f(j),k}) \tag{20}$$

The watermarked 2D coordinates $(x_{n,f(j),k}, y_{n,f(j),k})$ can be finally calculated by formula (21) and (22).

$$x'_{n,f(j),k} = \sqrt{\frac{1}{2}[1 + (f(j) - \frac{3}{4} + \frac{1}{2}w_j)\frac{2\overline{A_i}}{NA_{f(j),k}}]} \cdot x_{n,f(j),k} \tag{21}$$

$$y'_{n,f(j),k} = \sqrt{\frac{1}{2}[1+(f(j)-\frac{3}{4}+\frac{1}{2}w_j)\frac{2\overline{A_i}}{NA_{f(j),k}}]}\cdot y_{n,f(j),k} \tag{22}$$

The watermark extracting process was performed by using the index of the target layer and the index $F=\{f(j)|j\in[1,N_w]\}$ of the target polygon set that are stored jin the embedding process.

Firstly, this algorithm selects the target layer I_i in the suspected GIS digital map $\hat{I}=\{\hat{I}_n|n\in[1,N_I]\}$ and groups all polygon data in I_i. After obtaining the watermarked polygon sets $\widehat{S^*}=\{\widehat{S_{f(j)}}|j\in[1,N_w]\}$ in the order of the watermark bit index, the local mean area $\widehat{A_{f(j)}}$ of polygon data in each set $\widehat{S_{f(j)}}$ is calculated. The watermark bit $\widehat{w_j}$ can be extracted by the range where in $\widehat{A_{f(j)}}$ is.

$$\widehat{w_j}=\begin{cases}0, & \text{if } (f(j)-1)\frac{2\overline{A_i}}{N}<\widehat{A_{f(j)}}<(f(j)-\frac{1}{2})\frac{2\overline{A_i}}{N} \\ 1, & \text{if } (f(j)-\frac{1}{2})\frac{2\overline{A_i}}{N}<\widehat{A_{f(j)}}<f(j)\frac{2\overline{A_i}}{N}\end{cases} \tag{23}$$

Chang scheme focuses on the watermark embedding using the mean area of polygon in the selected layer improve robustness against RST, data or layer cutting and rearrangement, and other geometric attacks. This scheme can only processing polygon layer, and the mean area after watermarking need repeatedly computing for satisfy formular (19).

Table 3 shows the performance of the three algorithm as above statement. Each algorithm has some deficiency. This proposed paper will solve these disadvantage, as well as the proposed scheme satisfy invisibility and robustness.

Table 3. The performance of Ohbuchi, Kim and Chang algorithm.

	Ohbuchi	**Kim**	**Chang**
Invisibility	Yes	Yes	Yes
Process type	polyline/polygon	polyline	polygon
Simplification	Yes	No	No
Geometric attacks	Yes	Yes	Yes

Ⅳ. Proposed Watermarking Algorithm

As mentioned, GIS data structure can be made of different formats. The proposed scheme is based on ESRI shape file which is a popular geospatial vector data format for GIS map. Here, we use the polyline/polygon type data of ESRI shape file.

To embed watermark, we calculate the length/perimeter of all polylines/ polygons in a map, and find the minimum and maximum length/perimeter. Through dividing the dynamic range $[L_{\min}, L_{\max}]$ into N_W groups, we classify the polylines/polygons to N_W groups with the uniform step $(L_{\max} - L_{\min})/N_W$. The minimum length/perimeter ($L_{\min}$) and maximum length/perimeter ($L_{\max}$) will be saved as keys for extracting watermark while the number of groups (N_W) be saved. For improving the safety, we do PRNS processing with the watermark bits. And then, we embed a processed watermark bit to the local mean length/perimeter of polylines/polygons in a suitable group. The suitable group is mean that how many polylines/polygons in a groups is suitable for embedding watermark. The more polylines/polygons in a groups, the more safety and robustness, but the less enable groups for watermarking. The suitable values is defined by user, here we defined it 10. Lastly, we embed the watermark to the vertices' coordinate of polyline/polygon. When the watermarked map is created, we need calculate the average length/perimeter of all polylines/polygons as keys. Through saved keys we can obtain the watermarks without original map in the extracting process of watermark while satisfy invisibility. Figure 12 shows the process of watermark embedding. Figure 13 shows the process of watermark extracting.

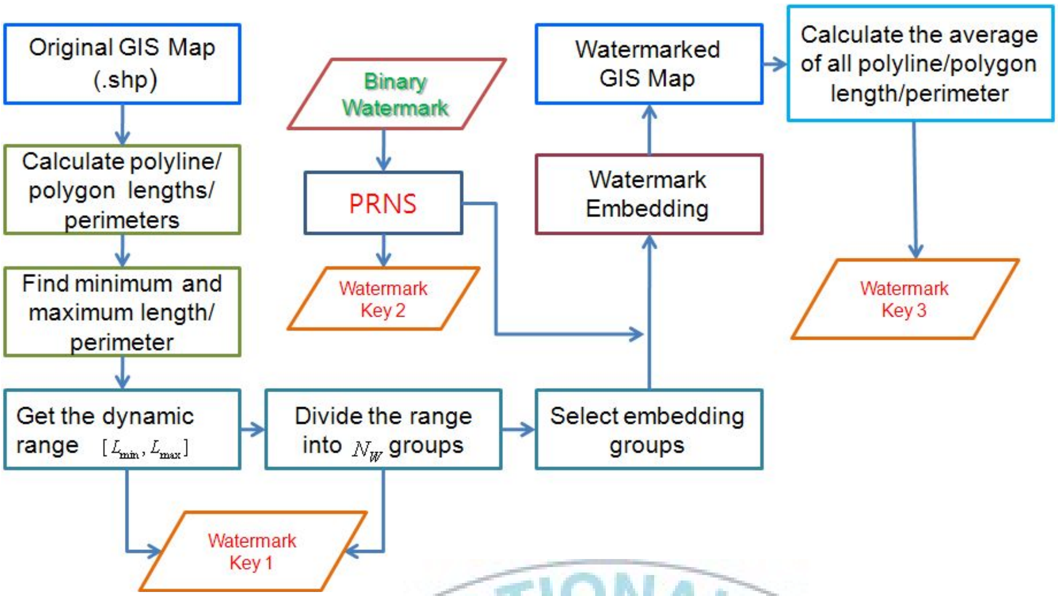


Figure 12. The process of watermark embedding.

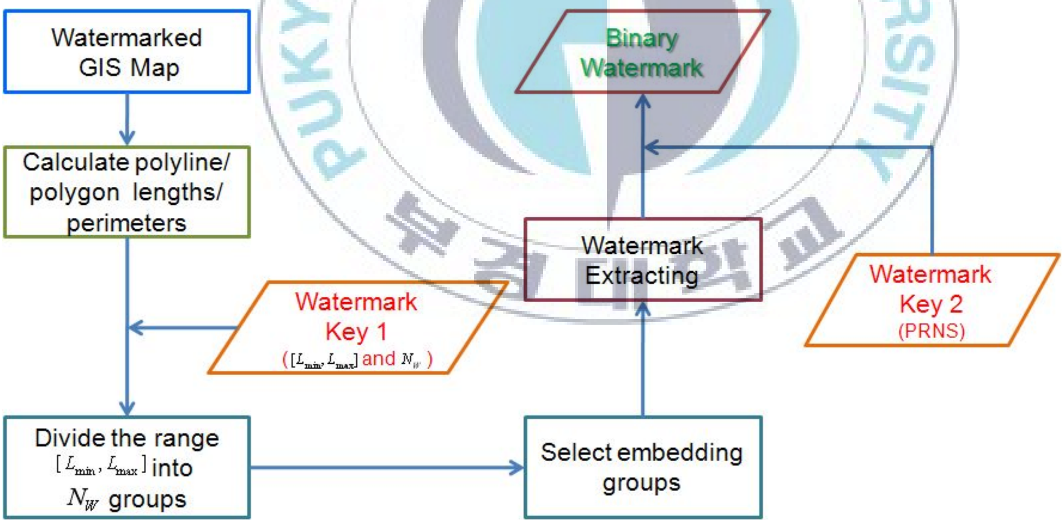


Figure 13. The process of watermark extracting.

4.1 Watermark Embedding

This paper offers a method that using the mean length/perimeter of polylines/polygons in GIS vector map. The proposed scheme divided the polyline/polygon into N_W group according to their length. Then embed the watermark bit into some groups.

In this paper, we using polyline/polygon of ESRI shapefile format. In shapefile format, a polyline consists of two or more vertices and the first and last vertex are not equal. Whereas, a polygon consists of above four vertices and the first and last vertex are equal. It is so-called as a closed polyline. And Figure 14 shows an example of the polyline and polygon in a GIS data.

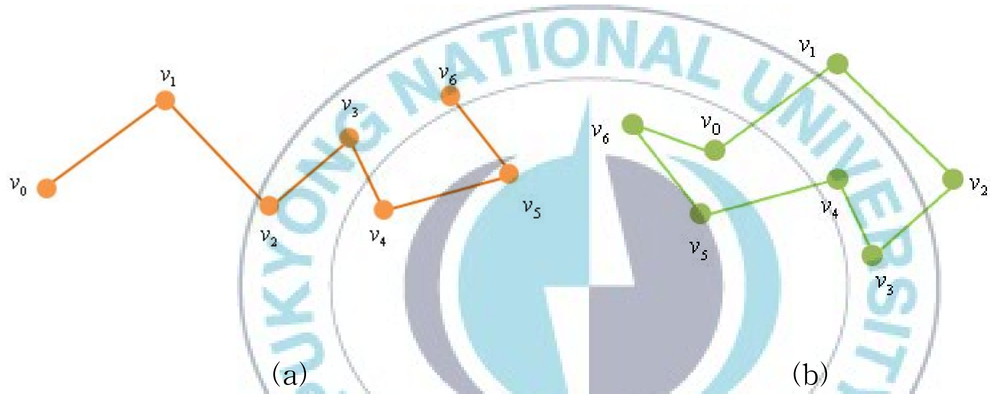


Figure 14. An example of (a) a polyline and (b) a polygon.

In order to satisfy robustness of various attacks in the previous section, we provide a watermarking method about the polyline length distribution or polygon perimeter distribution. According to the distribution of the length/perimeter, all polylines/polygons are assigned to different groups by their length/perimeter. The length of polyline or the perimeter of polygon are changed in invisibility range.

This proposed scheme is based on the polyline's length or polygon's perimeter. First of all, we obtain lengths L_i of all polylines $P_{li} =$

$\{v_{ik}|k\in[1,N_i], v_{i1}\neq v_{iN_i}\}$ or perimeter L_i of all polygons $P_{gi} = \{v_{ik}|k\in[1,N_i], v_{i1}=v_{iN_i}\}$.

The length of each polyline or the perimeter of each polygon is calculated using the formula (24).

$$L_i = \Sigma \parallel \overrightarrow{v_{ik}v_{ik+1}} \parallel \tag{24}$$

And then, find minimum length/perimeter $L_{\min}$ and maximum length/perimeter $L_{\max}$ from all of length/perimeter of polylines/polygons, so get the dynamic range $[L_{\min}, L_{\max}]$ of the length/perimeter with maximum length/perimeter $L_{\max}$ and minimum length/perimeter $L_{\min}$ and then divide this range to N_W groups by using the uniform interval $\triangle$. All polylines/polygons are allocated to any group by using their lengths/perimeters. Figure 15 shows N_W groups that are divided by the uniform interval.

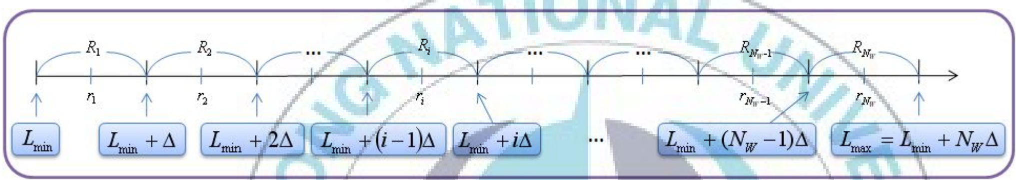


Figure 15. Groups of polylines′ length/polygons′ perimeter divided by the uniform interval.

$$\triangle = (L_{\max} - L_{\min})/N_W \tag{25}$$

From this process, we get N_W groups $R=\{R_i|i\in[1,N_W]\}$ while polylines or polygons of each group are expressed as formula (26) and (27), respectively.

$$R_i = \{P_{li,j}|L_{\min} + (i-1)\triangle \le L_{ij} < L_{\min} + i\triangle, j\in[1,N_{li}]\} \tag{26}$$

$$R_i = \{P_{gi,j}|L_{\min} + (i-1)\triangle \le L_{ij} < L_{\min} + i\triangle, j\in[1,N_{gi}]\} \tag{27}$$

Thus, R_i is ith. group that includes polylines $P_{li,j}$/polygons $P_{gi,j}$ with the length/perimeter L_{ij} in the range $[L_{\min}+(i-1)\triangle, L_{\min}+i\triangle]$, and N_{li} and N_{gi} are the number of polylines and polygon that are allocated to this group.

Here, we need store the range $[L_{\min}, L_{\max}]$ and group number N_W that as key through the key generation process. So that we would not need the original GIS map in the watermark extraction processing.

Note that there are total N_W groups, but some groups may not include polylines/polygons. We selected some groups that include ten or more allocated polylines/polygons for considering safety and robustness. Assume that there are L_w usable groups, we can embed L_w numbers of watermark bit into the selected groups by one bit one group.

For increasing robustness of the watermark against attacking, we use multiplicity time's watermark message repeating method. It repeats the entire message bit string multiplicity times according to usable embed groups L_w. The more repeat times, the more robustness, but less watermark message. Assume that the repeat times is C, so the watermark message length is $N_{wm}=L_w/(8*C)$. We can set the repeat times according to how long the watermark message and how much the robustness we wish. Although the effective watermark bits is $8*N_{wm}$, we will continue repeat the watermark message to make each usable groups embedded a watermark bit as we said above. Thus, the number of embedding watermark bits is L_w..

We also can input whatever watermark message we wish in the limit length N_{wm}. Then we change the watermark messages into watermark bits sequence $W(W_i \in \{0,1\}, i \in [1, L_w])$. For considering the safety, we will change the watermark bits value by pseudo-random number sequence (PRNS). The PRNS length can set by us in the range of one to L_w. If the length of PRNS is L_{PRNS}, we generate the PRNS $p_i(p_i \in \{0,1\}, i \in [1, L_{PRNS}])$ and save it as key. Then we use $w_i = W_i \odot p_j (i \in [1, L_w], j = i\% L_{PRNS}])$ get new watermark bit sequence $w(w_i \in \{0,1\}\ i \in [1, L_w])$ which had encrypted by PRNS.

After that we embed a watermark bit $w_i \in \{0,1\} (i \in [1, L_w])$ by using the mean length/perimeter $\overline{L_i}$ and the center value r_i of R_i group. So we calculate the mean length/perimeter $\overline{L_i}$ in each usable group R_i and the center value r_i of R_i group. The mean length/perimeter $\overline{L_i}$ of $P_{li,j}/P_{gi,j}$ in R_i group is calculated using the formula (28).

$$\overline{L_i} = \Sigma L_{ij}/N_i \tag{28}$$

The center value r_i of R_i group is calculated using the formula (29).

$$r_i = L_{\min} + (i-1/2)\triangle \tag{29}$$

We will change the mean length/perimeter of polyline/polygon according to the embedded watermark w_i. Thus, if $w_i = 0$, then the mean length/perimeter $\overline{L_i}$ should be less than the center value r_i . Or if $w_i = 1$, then the mean length/perimeter $\overline{L_i}$ should be equal or greater than the center value r_i. We change the mean length/perimeter $\overline{L_i}$ to $\overline{L_i^*}$ for satisfying the following embedding condition (formula 30).

$$\overline{L_i^*} = \begin{cases} L_{\min} + (i-1)\triangle \le \overline{L_i^*} < r_i, & \text{if } w_i = 0 \\ r_i \le \overline{L_i^*} < L_{\min} + i\triangle, & \text{if } w_i = 1 \end{cases} \tag{30}$$

If the mean length/perimeter $\overline{L_i}$ in a group is within the specified range satisfying the embedding condition, it should be not changed. Otherwise, the mean length/perimeter should be changed for satisfying the above condition. In this case, we should adjust the mean length/perimeter depending on the distance between the mean length/perimeter $\overline{L_i}$ and the center value r_i, for considering the invisibility.

$$\text{if } w_i = 0 \tag{31}$$

$$\text{if } \overline{L_i} \in [r_i, m_{i2}) \qquad \overline{L_i^*} = r_i - \frac{m_{i2} - \overline{L_i}}{2}$$

$$\text{if } \overline{L_i} \in [m_{i2}, L_{\min} + i\triangle) \qquad \overline{L_i^*} = r_i - \frac{(L_{\min} + i\triangle) - \overline{L_i}}{4}$$

$$\text{if } w_i = 1 \tag{32}$$

$$\text{if } \overline{L_i} \in [L_{\min} + (i-1)\triangle, m_{i1}) \qquad \overline{L_i^*} = r_i + \frac{\overline{L_i} - [L_{\min} + (i-1)\triangle]}{4}$$

$$\text{if } \overline{L_i} \in [m_{i1}, r_i) \qquad \overline{L_i^*} = r_i + \frac{\overline{L_i} - m_{i1}}{2}$$

$$m_{i1} = L_{\min} + (i - 3/4)\triangle \tag{33}$$

$$m_{i2} = L_{\min} + (i - 1/4)\triangle \tag{34}$$

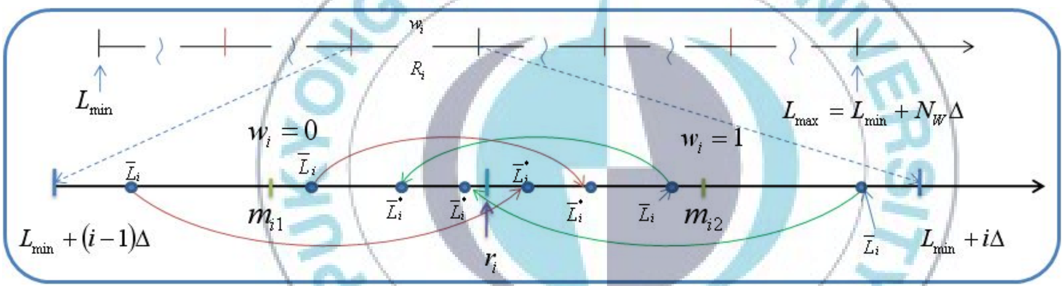


Figure 16. The specific diagram representation of the embedding method.

In this method, we should know two other values m_{i1} and m_{i2} with the center value r_i together to decide the new mean length/perimeter. Here, m_{i1} and m_{i2} are the median of the range $[L_{\min} + (i-1)\triangle, r_i]$ and $[r_i, L_{\min} + i\triangle]$, respectively.

Since the proposed scheme using polyline/polygon of ERSI shapefile. Here, polyline and polygon are based on the points, so we should change the coordinates of vertices in polylines $P_{li,j} = \{v_{ij,k} | k \in [1, N_{lij}]\}$ or polygon $P_{gi,j} = \{v_{ij,k} | k \in [1, N_{gij}]\}$ that are allocated in a group R_i. Finally, the watermarked vertices $v^*_{ij,k}$ can be obtained by (35) and (36).

$$v^*_{ij,k} = \alpha_i v_{ij,k} + v_{ij,1}(1-\alpha_i) \quad \forall k \in [2, N_{lij}] \text{ or } \forall k \in [2, N_{gi,j}] \tag{35}$$

$v_{ij,1}$ is the first vertex in a polyline/polygon P_{lij}/P_{gij}, α_i is change rate.

$$\alpha_i = \overline{L^*_i}/\overline{L_i} \tag{36}$$

By the above process, we change all vertices of polylines/polygons in each group according to the embedding condition.

$$\overline{L^*} = \sum L^*_i / P_{li} \quad \text{or} \quad \overline{L^*} = \sum L^*_i / P_{gi} \tag{37}$$

After embedded the watermarks, we should calculate the mean length $\overline{L^*}$ of all polyline P_{li} or the mean perimeter $\overline{L^*}$ of all polygon P_{gi} in the watermarked map. And we need store the mean length/perimeter $\overline{L^*}$ that as key, it is used when watermarked map is scaled.

4.2 Watermark Extracting

The watermark extracting process was performed by using the keys stored in the embedding process. The specific extracting process of watermark in a pirated GIS map is similar as the embedding process.

Firstly, we calculate the lengths/perimeters L'_i of all polylines/polygons in a watermarked map.

$$L'_i = \sum \| \overrightarrow{v'_{ik} v'_{ik+1}} \| \tag{38}$$

Then, we use key that is the range $[L_{\min}, L_{\max}]$ and group number N_W that are stored in the watermark embedding processing, divide this range to N_W groups by using the uniform interval $\triangle$ as the same as embedding process. All polylines/polygons in watermarked map are allocated to any group by using their lengths/perimeters.

In the same way, we calculate the mean length/perimeter $\overline{L_i'}$ of R_i usable group in watermarked map. Because of we using the range $[L_{\min}, L_{\max}]$ and group number N_W are not change, the center value r_i' of R_i group in watermarked map that is r_i.

Extract a watermark bit w_i' by comparing the mean length/perimeter $\overline{L_i'}$ and the center value r_i of R_i usable group. If the mean length/perimeter $\overline{L_i'}$ less than the center value r_i of the corresponding group, the embedded watermark shall be 0; If the mean length/perimeter $\overline{L_i'}$ equal or greater than the center value r_i of the corresponding group, the embedded watermark shall be 1.

$$w_i' = \begin{cases} 0, & \text{if } L_{\min} + (i-1)\triangle \le \overline{L_i'} < r_i \\ 1, & \text{if } r_i \le \overline{L_i'} < L_{\min} + i\triangle \end{cases} \tag{39}$$

After we extract watermark bits sequence $w' = \{w_i' | i \in [1, L_W]\}$, we need use PRNS to decrypt it. The decrypt watermark bits $W'(W_i' \in \{0,1\}, i \in [1, L_w])$ will be got by $W_i' = \overline{w_i' \oplus p_j}(i \in [1, L_w], j = i\% L_{PRNS}])$. We can judge whether a pirated map is copied or not by verifying BER of W' and W.

In watermark embedding part, we illustrate a method that use multiplicity time's watermark message repeating method that obtain the very well application in the case of the map was attacked. If the watermarked map was attacked, the length/perimeter of some polylines/polygons may be changed. In other word, some watermark bits may be false. In order to illustrating the process of checking watermark bits, we assume that the length of watermark is L_b (the total watermark bits is $8 \times L_b$) and the watermark repeat times is K.

We set the first group of watermark as the standard, and starting from the second group until the K group, compare each bit with the first group's bits. By counting the number of bits N_b ($N_b \in [1, K]$, $b \in [1, 8 \times L_b]$, including the first section bits) which are same as the corresponding bits of first group, we can get each bit's rate (N_b/K) under the first group standard. Because the first group may not the correct watermark message, the bits' rate is not really correct rate. If the rate of bit is greater than 50%, we consider that is correct and the bit is true bit, and conversely, the rate of bit is less than 50%, we consider that is improper and the true bit is contrary to this bit. After check each bit, we can get a group of correct watermark bits, and also, we can get the correct watermark message. In addition, as a group corresponding to embedding a bit and a group of polylines/polygons corresponding to a group, we can know which group of polylines/polygons is serious attacked according to the corresponding bit is error.

Ⅴ. Experimental Results and Analysis

To evaluate invisibility and robustness of proposed watermarking technique, we select 10 maps (5 polylines and 5 polygons) of Masan are based on ERSI shapefile format.

As in part Ⅳ, we described watermark embedding, we divided all polylines/polygons into groups according to their length. Because some groups may not include polylines/polygons, we selected some groups that include ten or more polylines/polygons and embedded the watermark into the selected groups. In the 10 maps, they contain different numbers of polylines or polygons, so these maps are divided into different groups and embedded different watermark numbers. Specific data as shown Table 4.

In the invisibility experiment, we shows two examples in Figure 17 and Figure 18. Figure 17 (a), (b), (c) and Figure 18 (a), (b), (c) are original maps of polyline and polygon, respectively. Figure 17 (d), (e), (f) and Figure 18 (d), (e), (f) are watermarked maps of Figure 17 and Figure 18 (a), (b), (c), respectively. In the visually, the original maps and watermarked maps are difficult to distinction.

To evaluate the invisibility of the image, audio and video using PSNR (peak signal-to-noise ratio). PSNR is an engineering term for the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. Because many signals have a very wide dynamic range, PSNR is usually expressed in terms of the logarithmic decibel scale [28]. The PSNR is defined as formula (40).

$$PSNR \ \ [\mathrm{dB}] = 20\log_{10}\left(\frac{\mathrm{MAX}(V_{x,y})}{RMSE}\right) \tag{40}$$

$$RMSE = \sqrt{\sum((V_{x,y} - V^{*}_{x,y})^2 / N_{total})} \tag{41}$$

$V_{x,y}$ and $V^{*}_{x,y}$ are vertex of original map and watermarked map, respectively. N_{total} is all vertices numbers of GIS map.

Table 4: The embedding object, the number of watermark and PSNR in each test map.

Type	Test map	The Num. of Polyline/ Polygon	The Num. of Groups	Message Repeat Times	The length of Watermark	PSNR [dB]
Polyline	Masan #1	2831	150	3	70	103.67
	Masan #2	3563	150	5	105	102.76
	Masan #3	3734	300	5	107	100.32
	Masan #4	10622	200	5	94	101.28
	Masan #5	46911	1000	7	382	121.12
Polygon	Masan #6	10346	260	5	107	128.59
	Masan #7	13486	300	5	128	130.27
	Masan #8	15260	200	5	135	130.92
	Masan #9	8003	300	5	110	129.49
	Masan #10	46106	1000	7	434	128.93

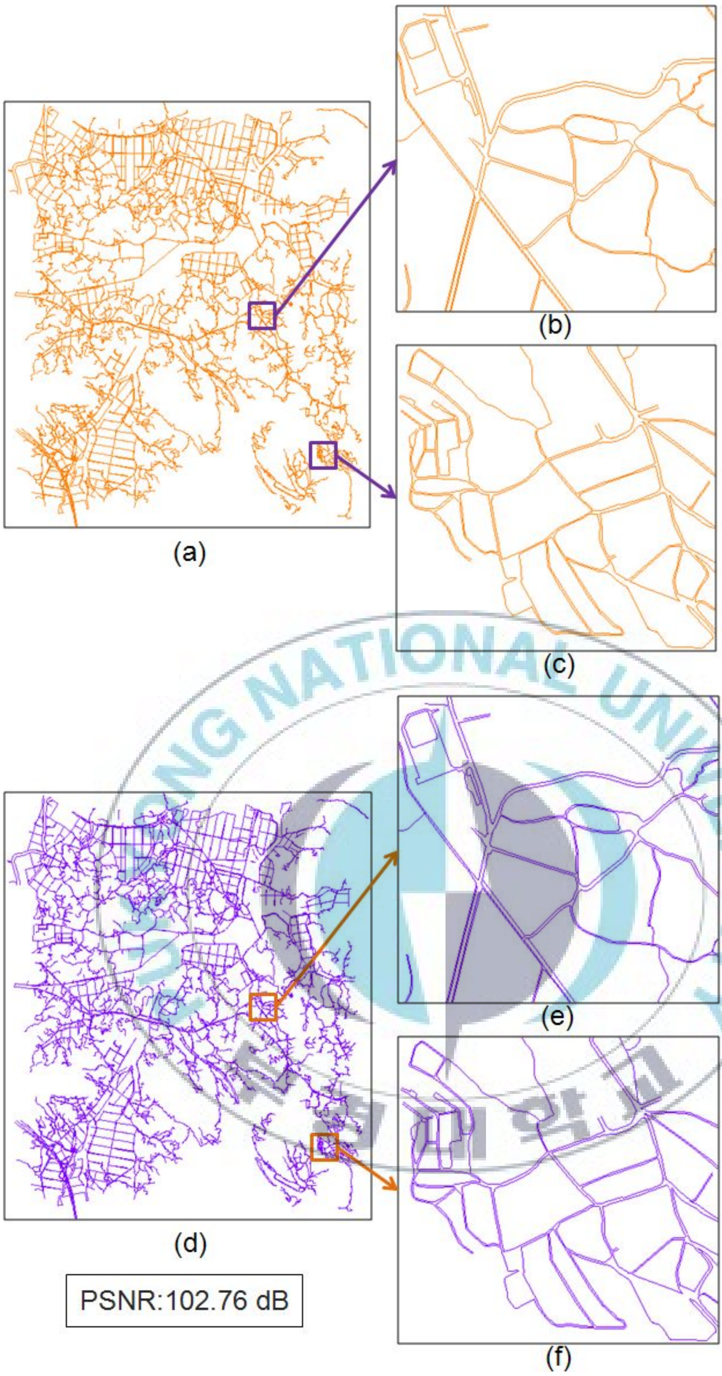


Figure 17. (a) Original polyline map; (b) and (c) magnified parts in (a); (d) watermarked polyline map; (e) and (f) magnified parts in (d).

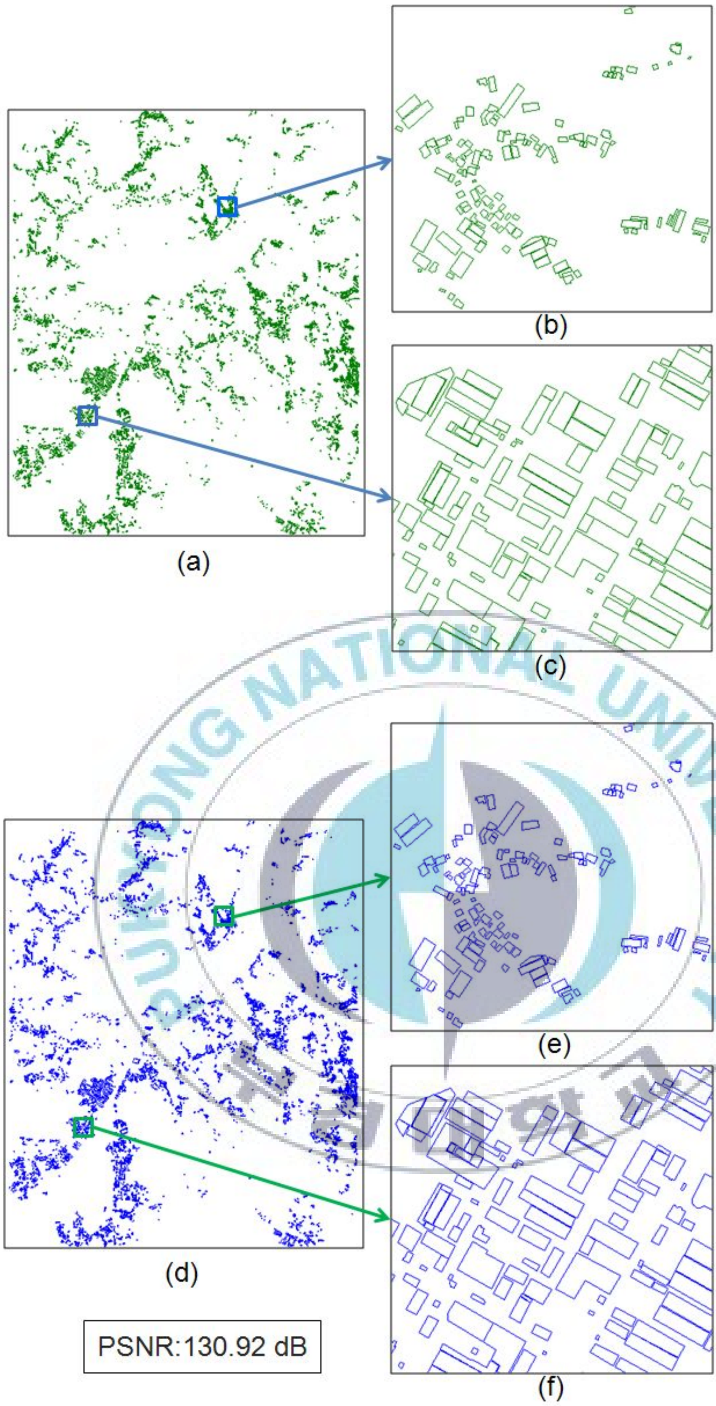


Figure 18. (a) Original polygon map; (b) and (c) magnified parts in (a), (d) watermarked polygon map; (e) and (f) magnified parts in (d).

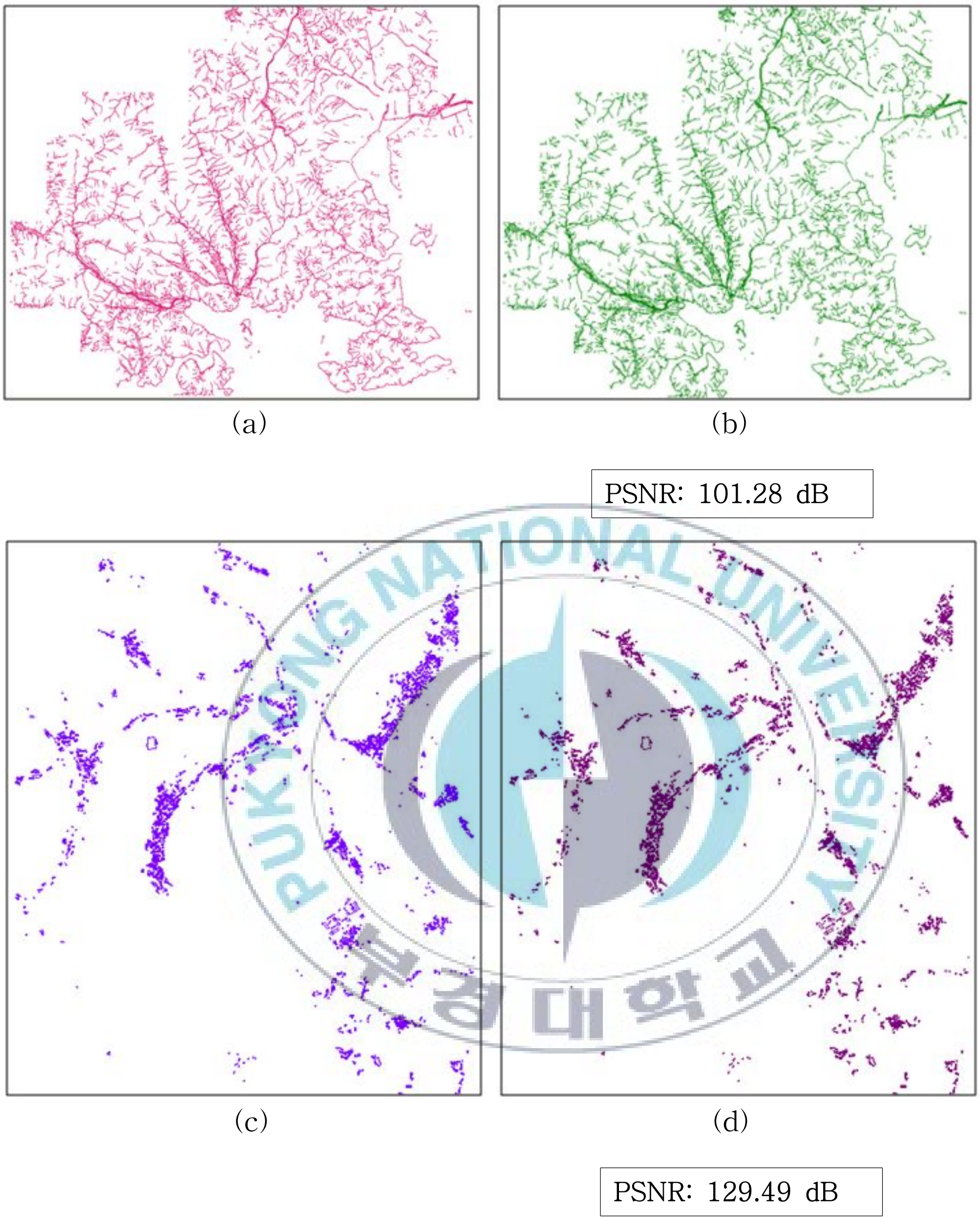


Figure 19. (a) Original polyline map; (b) watermarked polyline map, PSNR is 101.28 dB; (c) original polygon map; (d) watermarked polygon map, PSNR is 129.49 dB.

In the robustness experiment, we used two editing tools in AutoCAD Map 3D 2009 software and Arcview software for the watermark attacking. Then we compute BER (bit error rate) for evaluating the detecting precision. There are many kinds of attacks that can be performed in watermarked GIS data. The attacks are that rotation scaling translation (RST), object order scrambling (OOS), swapping, additive random noise, insert and delete vertex, and cropping. The experimental result of attacked compared with the method of Ohbuchi, J.H. Kim and H.J. Chang in robustness.

The following is various attacks to our watermarked maps. Figure 20 shows that rotated the watermarked maps of polyline and polygon. Figure 21 (b) is enlarged 2 times watermarked polyline map and Figure 21 (d) shows that shrunk 1/2 times watermarked polygon map. Figure 22 and Figure 23 are translated watermarked polyline and polygon map, respectively. Even if the watermarked map was attacked like as above condition, the watermark can be extracted.

Figure 24 is swapped watermarked polyline and polygon map, respectively. Figure 26 and Figure 27 are cropped about 25% to watermarked polyline and polygon map from different direction. Figure 28 shows that noise added into watermarked polyline map. In these attacked case, we should extract a large part of the watermark. The robustness of algorithm will measure using BER (bit error rate). Here, the BER is the number of bit errors divided by the total number of embedded watermark.

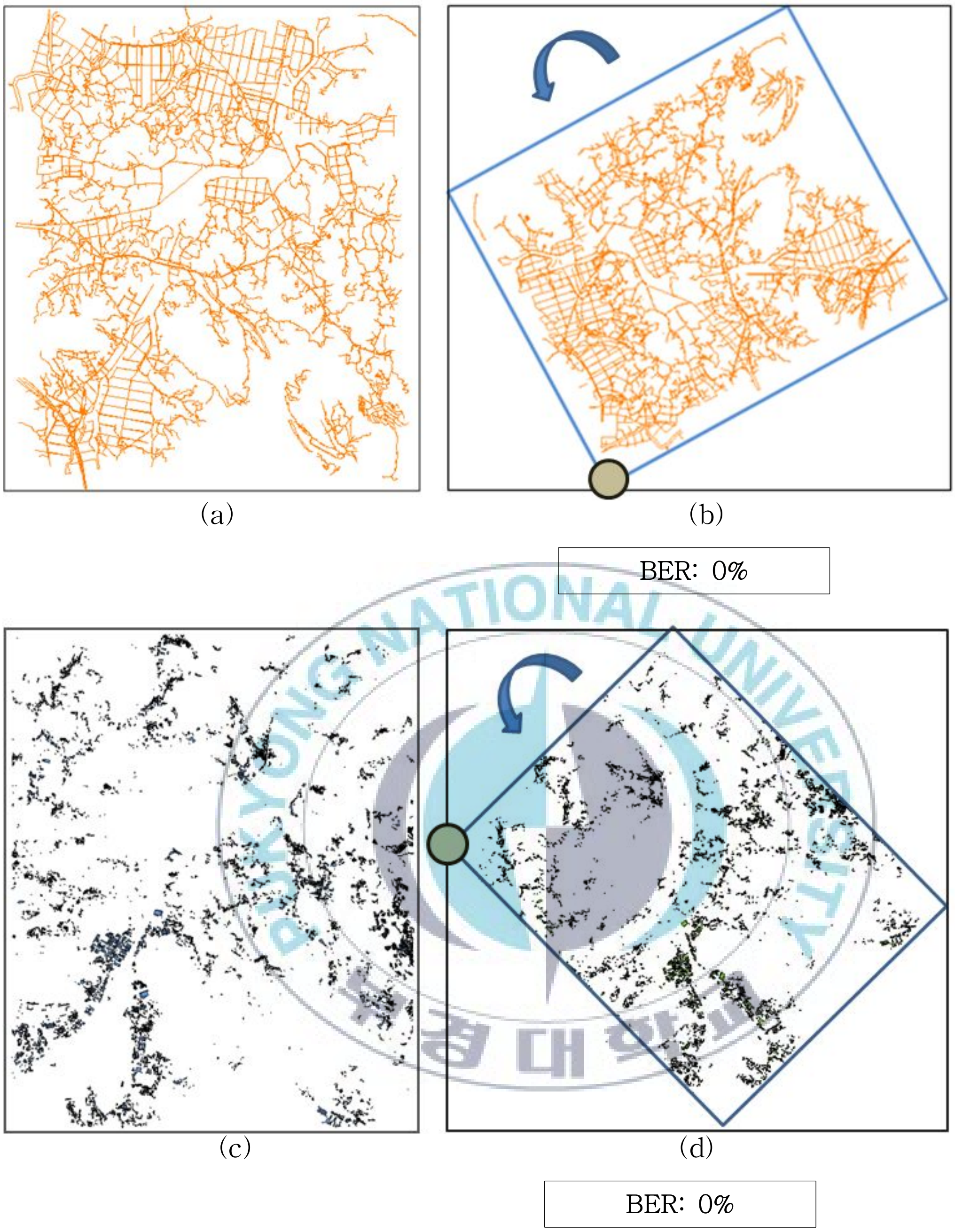


Figure 20. (a) Watermarked polyline map; (b) watermarked polyline map rotated to 120°; (c) watermarked polygon map; (d) watermarked polygon map rotated to 50°.

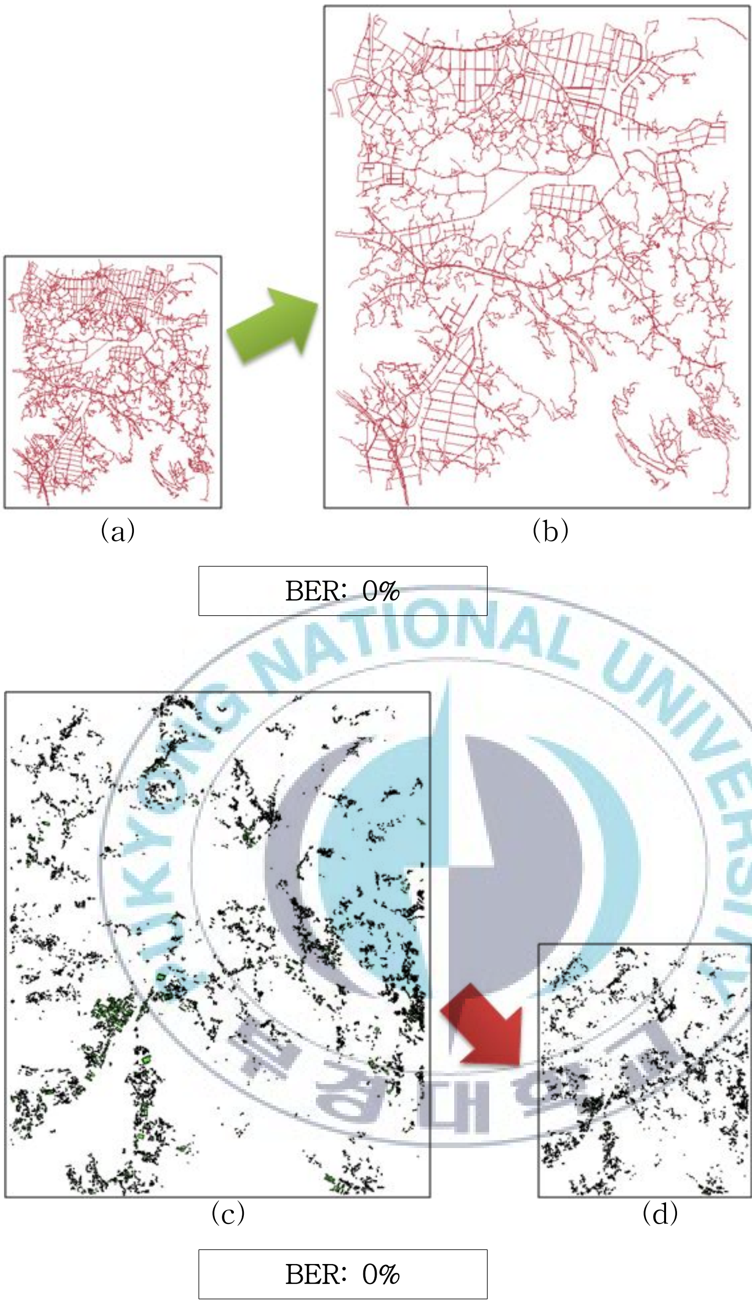
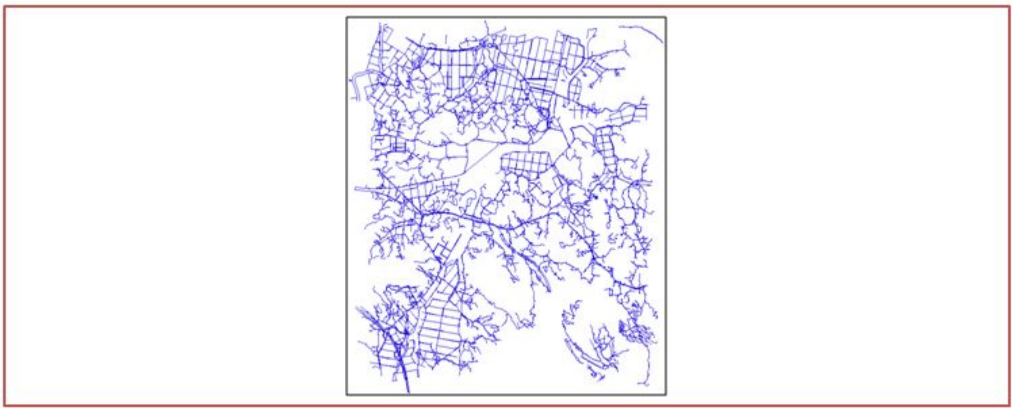
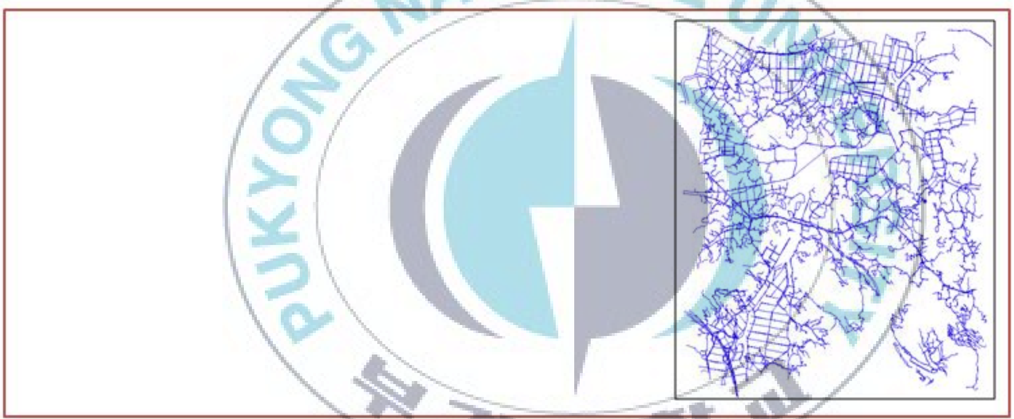


Figure 21. (a) Watermarked polyline map; (b) enlarged 2 times watermarked polyline map; (c) watermarked polygon map; (d) shrunk 1/2 times watermarked polygon map.



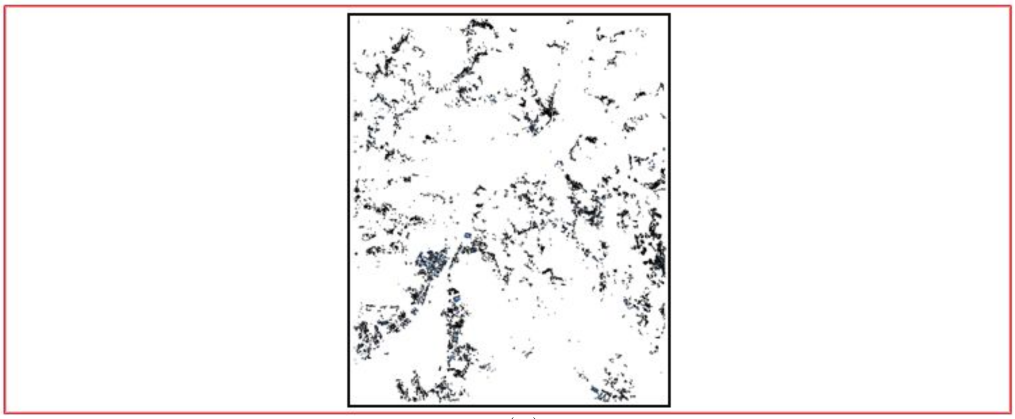
(a)



(b)

BER: 0%

Figure 22. (a) Watermarked polyline map; and (b) translated watermarked polyline map.



(a)

(b)

BER: 0%

Figure 23. (a) Watermarked polygon map; and (b) translated watermarked polygon map.

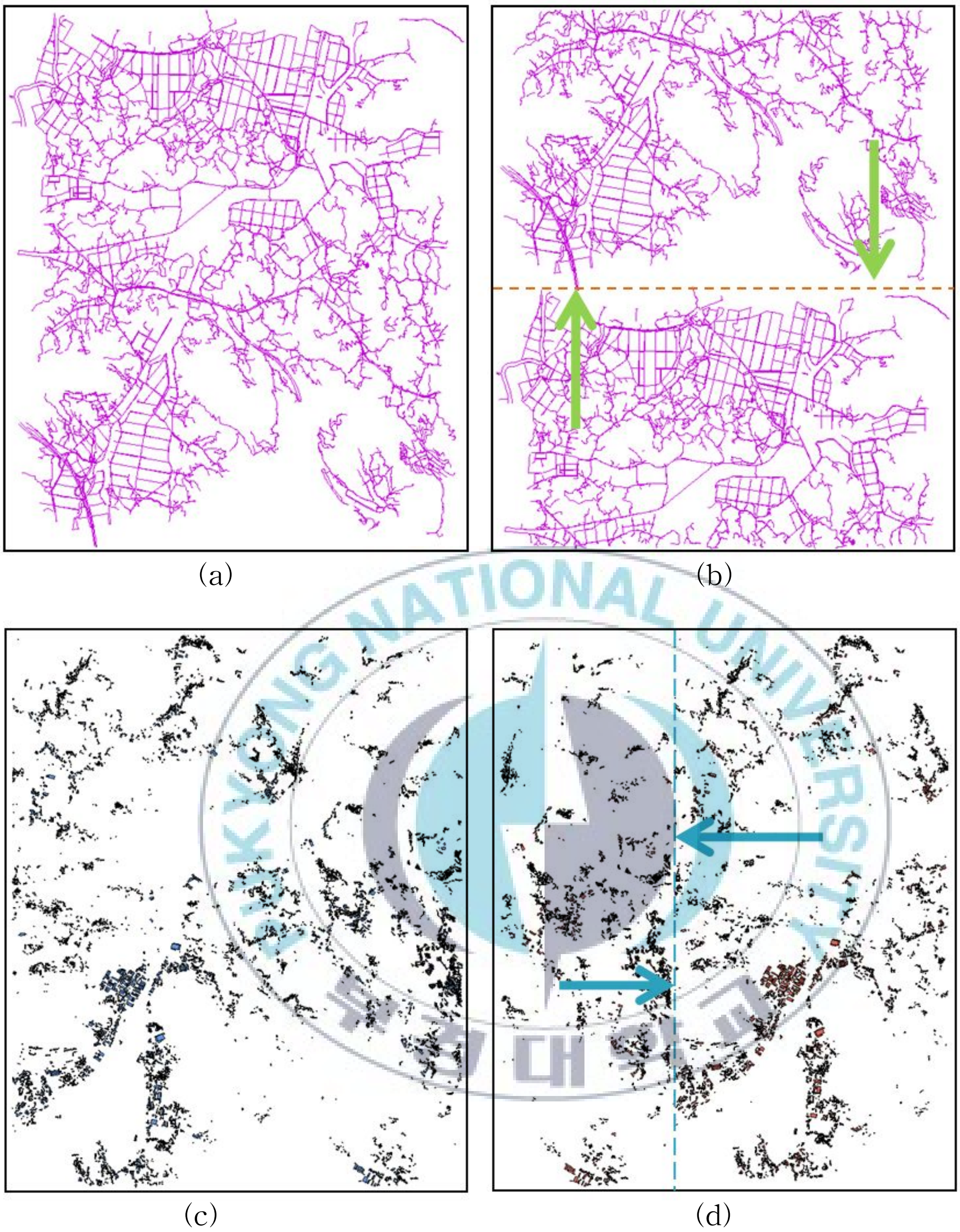
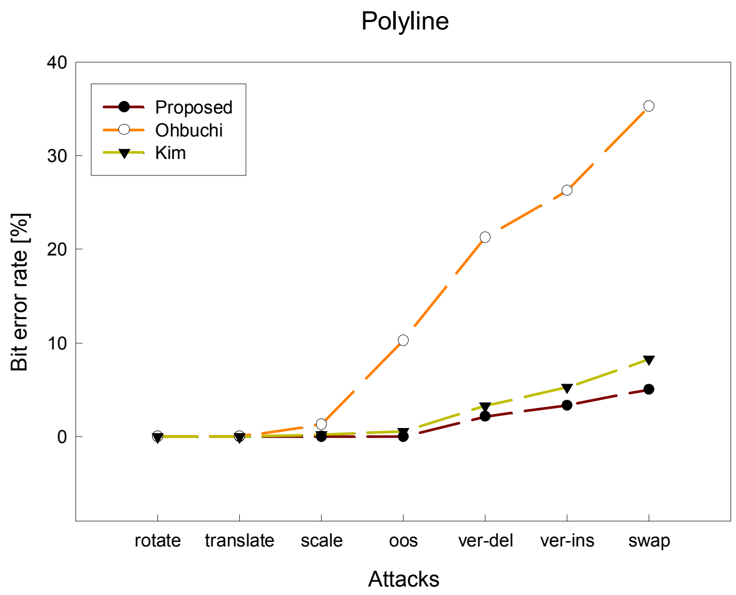
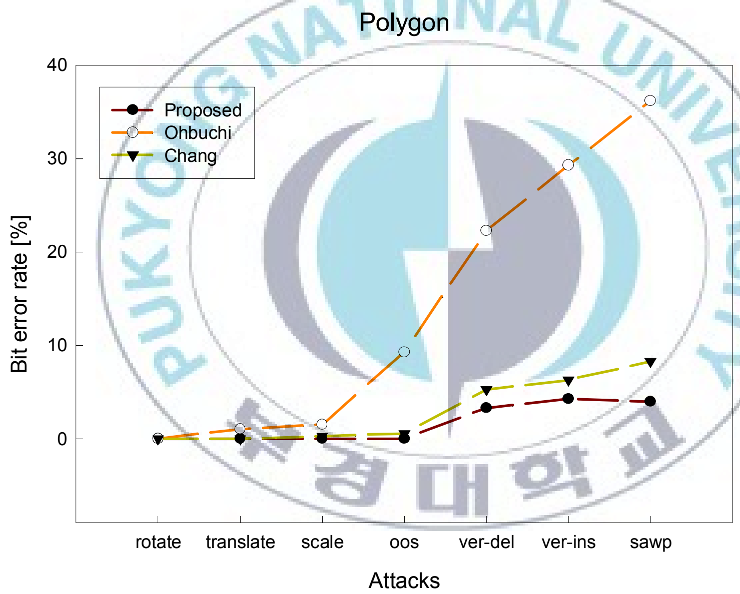


Figure 24. (a) Watermarked polyline map; (b) swapped watermarked polyline map; (c) watermarked polygon map; (d) swapped watermarked polygon map.



(a)



(b)

- oos: object order scrambling
- ver-del: vertex deletion
- ver-ins: vertex insertion

Figure 25. (a) BER of the detected polyline watermark on various attacks; and (b) BER of the detected polygon watermark on various attacks.

As shown in Figure 25, the BER of rotation scaling translation (RST) and object order scrambling (OOS) are 0. Because rotation and translation are only rotate overall map and change spatial location, the polylines' length/polygons' perimeter have not changed same as OOS. So the three attacks don't affect the extracted watermark. But scaling is different from rotation and translation, it is enlarge/shrink that change their length/perimeter of polylines/polygons for all map. In this case, firstly, we calculate the average length/perimeter $\overline{L'^*} = \sum L_i'$ for all polyline/polygon in attacked map. At the moment, we compare the $\overline{L^*}$ and $\overline{L'^*}$, if $\overline{L^*}/\overline{L'^*} = 1$, the attacked that is rotation or translation, if $\overline{L^*}/\overline{L'^*} \neq 1$, the attacked that is scaling. As we know the scaled rate $r = \overline{L'^*}/\overline{L^*}$, then we can easily obtain the length/perimeter of each polyline/polygon in the watermarked map, then the watermark extracting follow the part 4.2.

In our experiment, we crop the 10 maps into 1/2, 1/4 and 1/6 from multi angle. The results of comparing our scheme with other three methods is shown in Figure 29 (a) and (b).

For testing random noise attack, we use Gaussian noise to disturb the data of watermarked maps. Firstly, we random generate a decimal number z between 0 and 1. Then we calculate the probability of z appear in Gaussian distribution function $f(z)$. Here, we use standard Gaussian distribution. So the formula of $f(z)$ is:

$$f(z) = \frac{1}{\sqrt{2\pi}} e^{-\frac{z^2}{2}} \tag{42}$$

After that, we need generate a random decimal number r between 0 and 1. As r is in the average probability distribution of 0 to 1, the probability of $r \leq f(z)$ is just $f(z)$. When $r \leq f(z)$, z will be used to disturb the data of watermarked map. Assume A is the amplitude of noise, we can use formula (43) to generate noise.

$$noise = 2A \times z - A \qquad z \in [0,1] \tag{43}$$

Then we use formula (44) attack vertices.

$$\begin{cases} V^* = V + noise & noise \in [-A, A] \\ x^* = x + noise/2 & y^* = y + noise/2 \end{cases} \tag{44}$$

And then, we repeat this processing from random generating z to attack vertices. The probability of noise attack will be Gaussian distribution.

In noise attacks, we added random noise that having the amplitudes of 5 cm to 25 cm, respectively. The results is shown in Figure 29 (c) and (d).

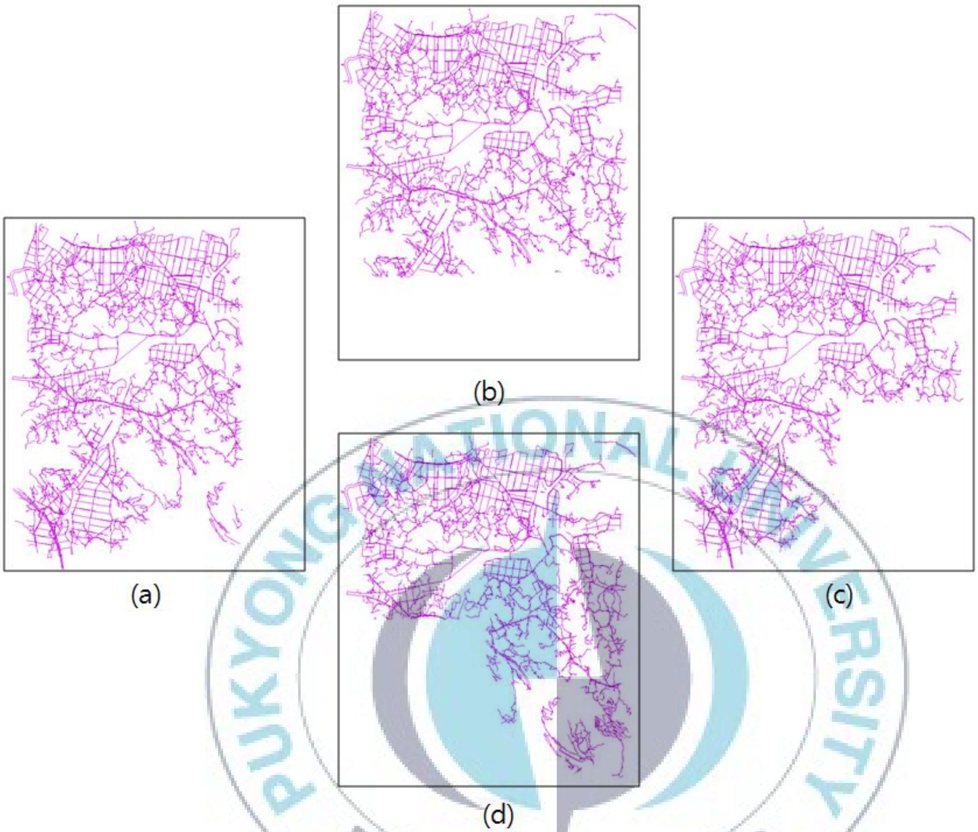


Figure 26. (a), (b), (c), (d) cropped about 25% watermarked polyline map from different direction.

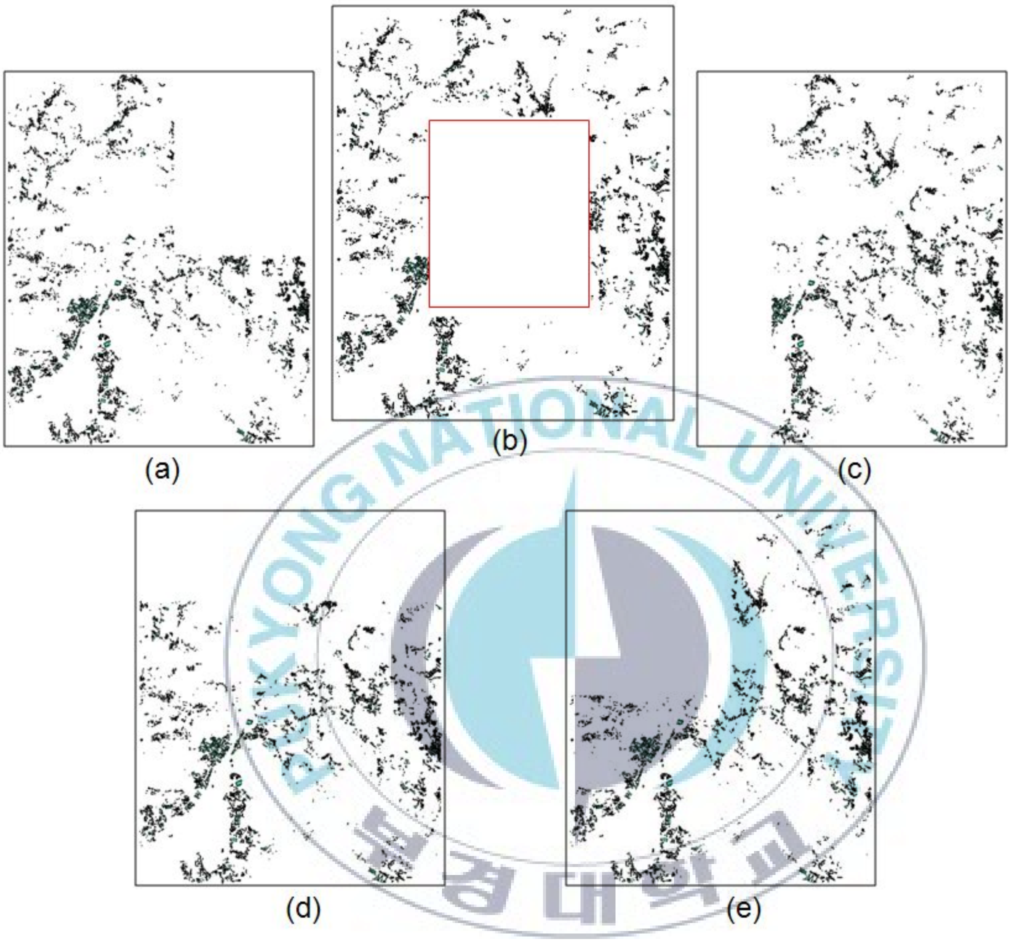


Figure 27. (a), (b), (c), (d), (e) cropped about 25% watermarked polygon map from different direction.

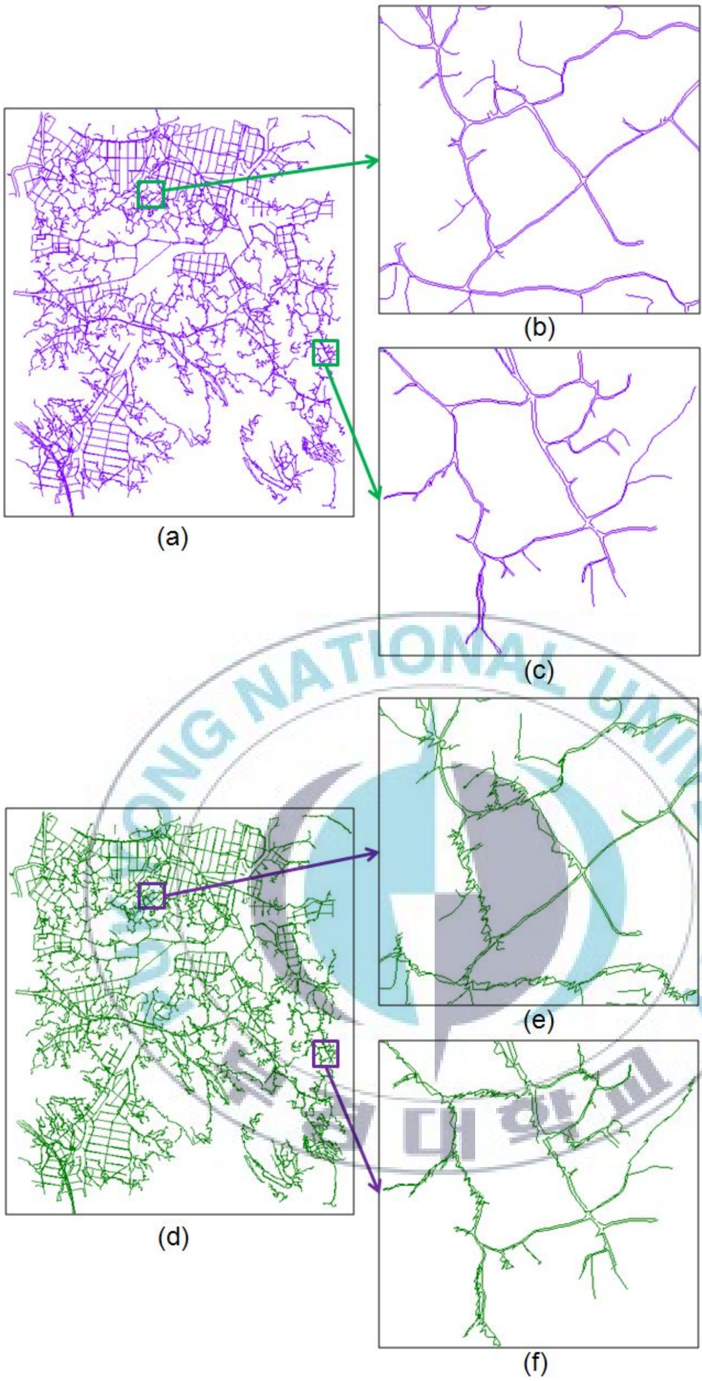


Figure 28. (a) Watermarked polyline map; (b) and (c) magnified part in (a); (d) noise added watermarked polyline map; (e) and (f) magnified part in (d).

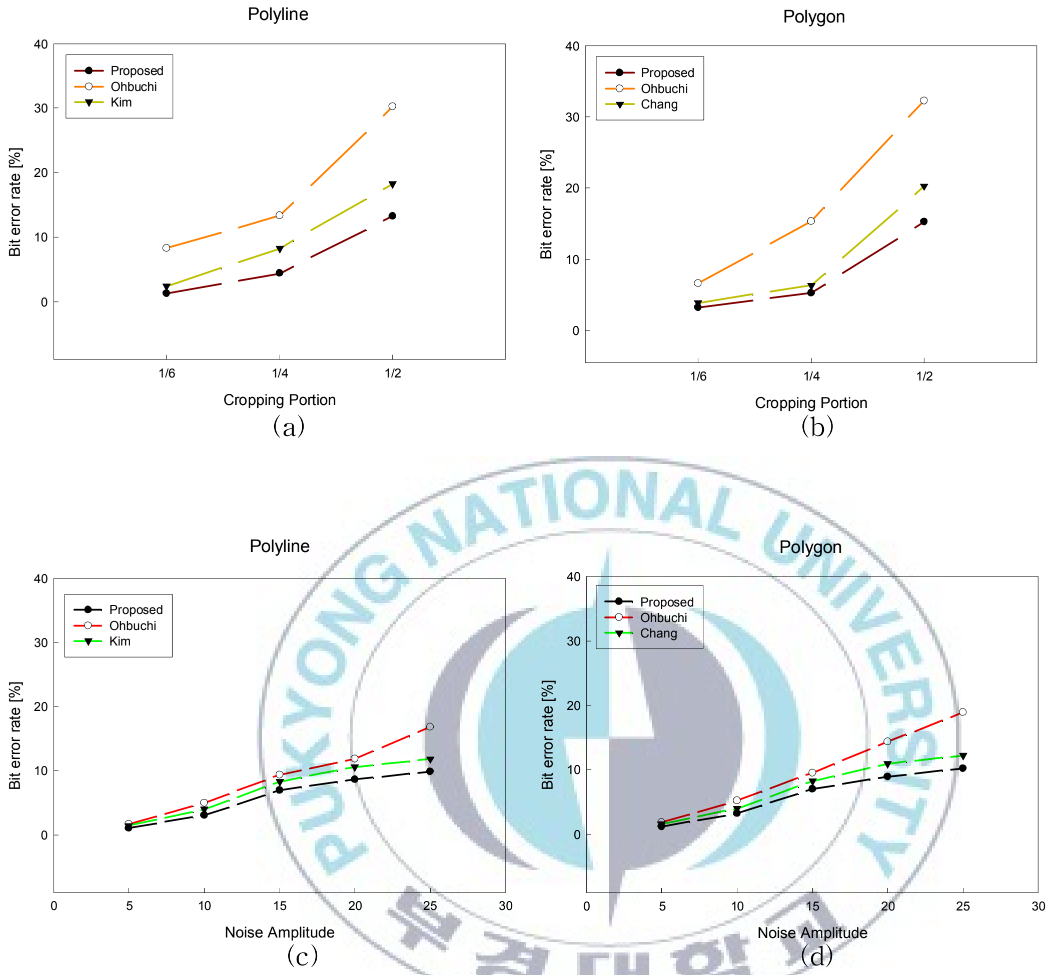


Figure 29. (a) BER of the detected polyline watermark on cropping attacks;
(b) BER of the detected polygon watermark on cropping attacks;
(c) BER of the detected polyline watermark on noise attacks;
(d) BER of the detected polygon watermark on noise attacks.

Ⅵ. Conclusion

In recent years, due to the development of the digital and network techniques, GIS has been developed rapidly. It is widely used in various fields such as utilities management, landscape assessment and planning, market analysis and navigation. So the copyright protection of GIS data has become an increasing complex topic.

This paper propose an invisible blind and robust watermarking scheme for the copyright protection of GIS vector digital map by using the polyline/polygon geometric characteristics. In our scheme, we calculate the length/perimeter of all polylines/polygons in a map and cluster polylines/polygons to some groups with the uniform step of length/perimeter dynamic range. For considering safety, we use PRNS encrypting the original watermark bits. And then we embed a encrypted watermark bit by changing the local mean length/perimeter of polylines/polygons of a suitable group. Lastly we embed the watermark message multiple times in all the map.

Our scheme is a watermarking method that has good invisible, safety and robustness for vector digital map. For blind watermarking, we change the lengths/perimeters of original polylines/polygons only in small ratio, our watermarked maps have high PSNR had proved this. For safety watermarking, we use PRNS encrypt original watermark and save some keys to protect the watermark message while do not need original map in the extracting process of watermark. For robustness watermarking, we use a group's polylines/polygons save one watermark bit and employ multiplicity time's watermark message repeating method. Our watermarked maps against various geometric attacks have good performance as show in experimental result.

References

[1] Geographic information system, http://en.wikipedia.org/wiki/Geographic_information_system.

[2] Digital watermarking, http://en.wikipedia.org/wiki/Digital_watermarking.

[3] I. Cox, J. Kilian, T.Leighton, and T.Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Trans. Image Processing*, vol. 6, no. 12,pp. 1673-1687,1997; see also *Proc. ICIP'*96, vol. III, pp. 243-246.

[4] I. J. Cox, J. Kilian, F. T. Leighton, and T. Shammon, "Secure spread spectrum for watermarking for multimedia," *IEEE Trans. Image Processing,* vol.6, pp.1673-1687, 1997.

[5] R. Ohbuchi, Hiroo Ueda, Shuh Endoh, "Robust watermarking of vector digital maps," *Proc. of the IEEE International Conference on Multimedia and Exp.*, Lausanne, Switzerland, vol. 1, pp.577-580, 2002.

[6] Jun-Hee Kim, Seung-Seob Park, Young Huh, Chun Og Lee, Pornchai Phukpattaranont, Ki-Ryong Kwon and Suk-Hwan Lee, "Adaptive Watermarking Scheme Using GIS Configuration Ingredients," ISSN 1975-4736, MITA 2009.

[7] H. J. Chang, J. H. Kim, S. H. Lee, K. S. Moon, B. K. Cha, and K. R. Kwon, "A Digital Watermarking of GIS Data Using the Polyline Characteristics," FCV2008, pp.8-13, Jan. 2008.

[8] Hye-Jung Chang, Bong-Ju Jang, Suk-Hwan Lee, Seung-Seob Park, Ki-Ryong Kwon, "3D GIS Vector Map Watermarking Using Geometric Distribution," Proc. of IEEE International Conference on Multimedia and Expo.NewYork, pp.1014-1017.

[9] Hye-Jung Chang, Bong-Joo Jang, Yong-Su Seo, Suk-Hwan Lee, Ki-Ryong Kwon, "Digital Watermarking on Geographic Information Data Using Geometric Characteristics and MAAG of Polygons," The Institute of Electronics Engineers of Korea, pp.639-640, June, 2008 (in Korea).

[10] Introduction to GIS, http://www.geom.unimelb.edu.au/gisweb/GIS

Module/GISTheory.htm.
[11] FUNDAMENTALS OF GEOGRAPHICAL INFORMATION SYSTEM, http://www.wamis.org/agm/pubs/agm8/Paper-6.pdf.
[12] Raster & Vector GIS, http://www.vcrlter.virginia.edu/~jhp7e/advgis/local/lectures/rastervsvector08.pdf?meid=51.
[13] ESRI Shapefile Technical Description, http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf.
[14] X. Niu, C. Shao, and X Wang, "Asurvey of digital vector map watermarking," International Journal of Innovative Computing, Information and Control, vol.2, no.6.2006.
[15] K. T. Park, K. I. Kim, H. I. Kang, and S. S. Han, "Digital geographical map watermarking using polyline interpolation," *In PCM '02: Proceedings of the Third IEEE Pacific Rim Conference on Multimedia*, pages58 - 65, London, UK, Springer-Verlag, 2002.
[16] M. Voigt and C. Busch, "Watermarking 2D-Vector data for geo-graphical information systems," *Proc. of the SPIE, Security and Watermarking of Multimedia Content*, San Jose, USA, vol. 4675, pp.621-628, 2002.
[17] G. Schulz and M. Voigt, "A High Capacity Watermarking System for Digital Maps," *Proc. of the 2004 Multimedia and Security Workshop on Multimedia and Security*, Magdeburg, Germany, pp.180-186, 2004.
[18] Yasser Dakroury, Ismail Abd El-ghafar, and Ashraf Tammam, "Protecting GIS Data Using Cryptography and Digital Watermarking," IJCSNS International Journal of Computer Science and Network Security, Vol.10, No.1, Jan. 2010.
[19] Anbo Li, Bingxian Lin, Ying Chen, and Guonian Lü, "Sduty on Copyright Authentication of GIS Vector Data Based on Zero-watermarking," The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. Vol. XXXVII. Part B4. Beijing 2008 Pacific Rim Conference on Multimedia, Springer-Verlag, pp.58-65, 2002.
[20] Y. Li and L. Xu, "A blind watermarking of vector graphics images,"

Proc. the Fifth International Conference on Computa - tional Intelligence and Multimedia Applications, pp.27-30, 2003.

[21] H. I. Kang, K. I. Kim, and S. S. Han, “Watermarking techniques using the Drawing Exchange Format (DXF) file,” *Proc. of the 2001 Multimedia and Security Workshop on Multimedia and Security*, Ottawa Canada, pp.19-22, 2001.

[22] M. Voigt and C. Busch, “Feature-based watermarking of 2D - vector data,” *Proc. of the SPIE, Security and Watermarking of Multimedia Content,* SantaClara, USA, vol. 5020, pp.359-366, 2003.

[23] M. D. Swanson, B. Zhu, and A. H. Tewfik, “Robust data hiding for images,” *Proc of the IEEE Digital Signal Processing Workshop*, pp.37-40, 1996.

[24] Xiao-Jiao Huo, Bong-Joo Jang, Kwon-Youl Ryu, Suk-Hwan Lee, and Ki-Ryong Kwon, "Information Hiding Scheme for GIS Shapefile," 2010 Korea Multimedia Society Spring Conference, May, 2010 (in Korea).

[25] Xiao-Jiao Huo, Bong-Joo Jang, Teak-Young Seung, Dae-Yeop Im, Sung-Bum Kim, Suk-Hwan Lee, Ki-Ryong Kwon, "A Polyline Watermarking for GIS Digital Vector Map," The Institute of Electronics Engineers of Korea, June, 2010 (in Korea).

[26] Xiao-Jiao Huo, Suk-Hwan Lee, and Ki-Ryong Kwon. "A Copyright Protection Scheme of GIS Vector Map Using the Length Distribution," Proceeding of 2010 China-Korea Joint Conference On Information and Communications (JCIC2010), August, 2010.

[27] Xiao-Jiao Huo, Teak-Young Seung, Bong-Joo Jang, Suk-Hwan Lee, Ki-Ryong Kwon. "A Watermarking Scheme Using Polyline and Polygon Characteristic of Shapefile", 2010 Third International Conference on Intelligent Networks and Intelligent Systems, pp649-652, November, 2010.

[28] Peak signal-to-noise ratio, http://en.wikipedia.org/wiki/Peak_signal-to-noise_ratio.

Acknowledgements

How time flies! Recollecting two years ago, I feel it was yesterday, but in a twinkling, I will graduate. Here, I am very glad to take this opportunity to express my gratitude to all the people who have ever helped me.

Firstly, I am extremely grateful to my advisor professor Ki-Ryong Kwon, who let me come here to continue my studying; who has a warm heart to tolerate, care, help me; and who use a strictly scholastic attitude to guide my studying. Therefore, in these two years, I had learned much knowledge about digital image processing and digital watermarking technique. Also, I give my sincere and hearty thanks to another advisor professor Suk-Hwan Lee, who has extensive knowledge; who is a kind and warm professor; and who is very patient for my instruction every time. I am very grateful to two professors for the helping and instructing in these two years.

In addition, I would like to thank my thesis examination professors--Bong-Kee Sin and Sang Uk Shin. Thanks their guidance and examination for my thesis. And, give my sincere thanks to all professors who had taught me. Thank you for your teaching.

Here I am very grateful to all members in MCSP laboratory. I am a Chinese, when I came here I could not speak in Korean, I had a lot of trouble in life and studying. But I got everyone's help, thank these brothers: Teak-Yong Seung, Bong-Joo Jang, Sung-Hye Jeon, Jin-Ho Kim, Dae-Yeop Im, Sung-Bum Kim; two sisters: Jun-Hee Kim, Hye-Jung Chang; a fresh Chinese student named Liu Yang; and other seniors and undergraduates. Thank you very much for your helping to my life and studying. Yours care and help make me feel MCSP Lab. is home, that a warm family it is.

Meanwhile, I am also extremely grateful to all my friends in here, especially my boyfriend, thanks him for accompanying me on my two years Master studying. And my Korean nationality friend--Feng-Ji Piao, who act as a good translation in many case and help me solve many problems. I would also like to thank my other friends--Chinese friends, Korean friends and other countries' friends. You make me much smoothly in my studying, and much splendid in my Korea life. I am really grateful to you.

Finally, many thanks give my family and my boyfriend's parents for their unfailing love, support and understanding. And also thanks to all who care about me for your concern.

My thesis is finish and my graduate life is also end, however, the friendship between us will continue. Give my most sincere appreciation to you and give my best wishes to you!