Spatial Data Mining Approaches for GIS Vector Data Processing

By

Ahmed Abubahia

September 2018

The thesis is submitted in partial fulfillment of the requirements for the award of the degree of Doctor of Philosophy of the University of Portsmouth.

School of Computing
Faculty of Technology
University of Portsmouth

Abstract

Geographical Information Systems (GIS) are very useful system in capturing, storing, manipulating, analysing, managing, and presenting spatial data. GIS systems can be used for solving problem and making decision in various applications. Data mining is the automated process of discovering patterns in data. This thesis outlines the issues and challenges of GIS data to advance the use of data mining techniques in the context of GIS applications. This thesis focuses mainly on two domains of applications: first is the digital vector map copyright protection and second is the digital vector map partitioning. Further more, this thesis presents an efficient approach for identifying the resilient locations for embedding the watermark; improving the robustness of the watermarking approach against defined set of attacks; investigating the impact of clustering approaches on the application of vector map protection; defining an effective metric for measuring the topological distortion in the watermarked GIS maps; and developing a spatial clustering approach that takes into consideration the GIS map properties. The experimental results show the reliability of using data mining techniques in combination with GIS map properties in advancing the GIS applications with more focus on spatial data protection and partitioning.

Dedication

To mum and dad

Declaration

I declare that: 'Whilst registered as a candidate for the above degree, I have not been registered for any other research award. The results and conclusions embodied in this thesis are the work of the named candidate and have not been submitted for any other academic award'.

Acknowledgments

Several people supported me in conducting and completing the research presented in this thesis.

First and foremost, I would like to thank my supervisor Dr. Mihaela Cocea for her support and guidance throughout all the stages of this research. She has been a real mentor, helping me improve my scholarly thinking and research skills. I am grateful for her encouragement and support, and for the invaluable time spent discussing research ideas.

I wish to thank my family and friends for their encouragement and belief in my abilities. Also, special thanks to my wife for her patience, understanding and support.

Publication List

Journal Publications

- **Abubahia, A.**, Cocea, M. "Evaluating the Topological Quality of Watermarked Vector Maps" 2018. Accepted in *Applied Soft Computing.*
- **Abubahia, A.**, Cocea, M. "Advancements in GIS map copyright protection schemes - a critical review." *Multimedia Tools and Applications*, 76(10):12205–12231, 2017.

Conference Publications

- **Abubahia, A.**, Cocea, M. "Vector Map Properties for GIS Data Copyright Protection". *The 27th IEEE International Conference on Tools with Artificial Intelligence.* PP.575–582, Vietri sul Mare, Italy, 09-11 November 2015.
- **Abubahia, A.**, Cocea, M. "A Clustering Approach for Protecting GIS Vector Data". *The 27th International Conference on Advanced Information Systems Engineering.* PP.133 - 147, Stockholm, Sweden, 8-12 June 2015.
- **Abubahia, A.**, Cocea, M. "Partition Clustering for GIS Map Data Protection". *The 26th IEEE International Conference on Tools with Artificial Intelligence.* PP.830 – 837, Limassol, Cyprus, 10-12 November 2014.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Geographic information systems (GIS) data offer good promise for many applications such as: public health, road networks, business management and military applications. GIS map data is an emerging area of data science research, due to the GIS data characteristics of high-cost of production, locations accurate values, small-size storage and efficient real-world representation. The need for protecting the copyright of digital vector maps has become an emergent topic within the GIS (Geographic Information System) research community that stemmed from the rapid growth of intelligent tools and devices (Chang, 2012; Longley et al., 2011). One of the main economic, social and legal aspects of using GIS data is defined by copyright protection (Wu et al., 2013b). In addition, the growing demand for GIS data calls for high computation reliability, as the GIS data grow rapidly in volume and require more complex computation (Wei et al., 2015; McKenney and Schneider, 2007; Shuliang et al., 2013).

Geographical data have become widely available in digital format due to the advancement in computer devices, database systems, mapping applications and IT (Information Technology) (Burrough et al., 2013; Wang et al., 2007).

Geographical data can be categorized into two types: vector and raster data (Okabe, 2016; Abubahia and Cocea, 2017). Vector data represents geographical information by using basic geometrical shapes such as points, lines and polygons (Huo et al., 2011a), while raster data represents information in a matrix of cells or pixels of uniform size (e.g. satellite image data). Most geographical systems represent data in vector format (Lee and Kwon, 2013; Bong-Joo et al., 2014).

1.1 GIS Vector Data Structure

Geographic information systems (GIS) are computer-based systems that facilitate the input, storage, manipulation and output of geographic location-based data. GIS data models are classified into two categories: raster and vector data models. In GIS context, satellite images are the most known example of the raster model. GIS vector data, which is the focus on this thesis, has three components: spatial data, attribute data and index data. Spatial data describes the map itself and always takes the form of three basic geometrical entities, which are:

- Points – point entities are used to define a single location of an object; they are used to represent real-world objects, such as bus stops, traffic lights and street lights.
- Polylines – line entities define linear objects; they can range from two-point lines to complex strings that have many vertices; lines are used to represent real-world objects, such as rivers and roads.
- Polygons – polygon entities define area-based objects; they can range from rectangles to multi-sided shapes with many vertices; polygons are used to represent real-world objects, such as lakes, shopping areas, buildings and city boundaries.

All these map entities are formed by many organized vertices; spatial data is actually a sequence of coordinates of these vertices based on a certain geographical coordinate system. The most used formats of GIS spatial data are:

1. ESRI (Environmental Systems Research Institute) shape file. The ESRI shape file (ESRI, 1998) has become an industry standard in geospatial data format due to its compatibility, to some extent, with recently released GIS software products.
2. CAD (Computer Aided Design) drawing. CAD drawings are used in many disciplines such as engineering, architecture, surveying, and mapping to define real-world objects in the context of geographic information systems. DXF (Drawing Interchange File) (AutoCAD, 2007) files are a popular format for storing and exchanging vector-based spatial information.

The attribute data describes the properties of map entities through links to the location data. Attributes can be, for example, names or matching addresses. The most known example of GIS attribute data format is the ESRI database file that is associated with the ESRI shape file and needs to have the same prefix as the shape file (ESRI, 1998).

Table 1.1: Vector data versus image/raster data

Vector Data	Image Data
Use points and lines to represent features	Represented as 2-dimensional array of brightness values for pixels
Resolution is determined by precision of vertices' coordinates	Resolution is determined by pixel size
Efficiently represents sparse data	Efficiently represents dense data
Spatial relations exist	Spatial relations do not exist
Efficient storage of sparse data	Requires large amounts of storage space
Small redundancy to hide watermark	Considerable redundancy to hide watermark
Explicit representation of linear features	Deals poorly with linear features

Last but not least, in the GIS context, the index data describes a file structure, such as total file length, for either spatial or attribute data. The ESRI index file (ESRI, 1998) is the best known example of index files.

The digital vector map is one type of important digital resources which has been widely used in navigation, urban planning, and many other areas. Due to high processing cost in the acquisition of digital map data, it becomes a valuable resource to its owner and has high price. Nevertheless, as one kind of digital data, a digital map could be easily copied. Therefore, copyright protection techniques for digital map have received extensive research attention in recent years.

The distinguishable structure of vector data from image data adds more challenges to be taken into account in the research field of digital watermarking. Table 1.1 outlines the major differences between vector data and image data in the context of digital watermarking applications (Li et al., 2012b; Longley et al., 2005; Niu et al., 2007; Wang et al., 2012a).

1.2 Key Terms and Conceptions

This section introduces the definitions of used terms in this thesis, such as: capacity, fidelity, robustness and computational time.

- Capacity – refers to the amount of embedded bits within the digital vector map.
- Fidelity – refers to the perceptual similarity between the watermarked data and its original form.
- Robustness – refers to the resilience of the inserted watermark to any processes (attacks) aimed at either removing or distorting it.

- Computational time – refers to the period of time that is required to perform the embedding process and obtaining the watermarked data.
- Attacks – refer to a set of modifications on the watermarked map in aiming to remove the watermark.

1.3 Problem Statement

The existing problems and issues in the related areas of research can be summarised as follows:

- There is little research considering the nature of GIS data in digital watermarking research, and only a few approaches are using data mining techniques in this research. In other words, GIS data received less attention than images, audio, texts and videos in the field of watermarking research, as pointed out in several recent review papers.
- There are certain differences between general multimedia watermarking and vector map data watermarking in many aspects such as the principle for data embedding and extracting, the criteria of data quality evaluation, the manners of possible attacks.
- The lack of adequate evaluation metrics and the lack of consideration for the trade-off between fidelity and capacity requirements of the watermark. At present, many authors obtain the results by following image watermarking evaluation standards, then apply it to vector map data for testing their algorithm, and then they claim the algorithm performance is good. Also, the trade-off between capacity and fidelity, which has an important implication on the watermarking research performance, is not considered in the research literature.

1.4 Aims and Objectives

The aim of this thesis is to advance the use of data mining techniques in the context of GIS applications. This thesis focuses mainly on two domains of applications: first is the digital vector map copyright protection and second is the digital vector map partitioning.

Objectives of this thesis are:

- Identifying the watermark embedding position by using a clustering approach;
- Improving the robustness to attacks/modifications such as: the simplification (removing some vertices from GIS vector data) and interpolation (adding new vertices to GIS vector data);

- Comparing different clustering approaches in identifying the watermark embedding locations in the map;
- Defining a metric for measuring topological quality of polygon-based vector maps;
- Developing workload balance based spatial clustering approach for partitioning GIS polygon based maps with massive number of vertices and complex shapes.

1.5 Contributions

In this thesis, the new contributions to knowledge are:

1. The use of k-medoids clustering technique in the process of identifying the locations for embedding the watermark has an influence on the security of the watermarked map measured in terms of capacity, fidelity, computational time and robustness.
2. The use of k-medoids clustering technique in combination with the bounding box centers has a significant implication on the trade-off between the fidelity and the capacity metrics, and resulted in higher fidelity as capacity increased. In addition to the improvement of the trade-off between fidelity and capacity, the use of bounding box centers adds more robustness to the simplification and interpolation attacks due to their independence from the number of vertices in a polygon.
3. The use of partitioning clustering techniques in combination with the bounding box property of GIS vector map for locating the watermark bits into polygons' vertices has a significant implication on protecting the GIS vector map copyright, especially in terms of addressing the vulnerability to simplification and interpolation attacks, while preserving a good trade-off between fidelity and capacity.
4. The use of partitioning clustering techniques in combination with the vector polygon based map topology rules leads to defining a metric that allow comparisons between watermarked maps of different sizes and of different watermark sizes, and, thus, can be used to assess the quality of watermarked vector maps.
5. The use of evolutionary computation technique in combination with GIS map properties advances the implementation of partitioning clustering approach in the context of GIS vector maps.

1.6 Thesis Structure

This thesis is organized as follows: Chapter 2 covers the background and literature review in regards to the application domains of GIS vector map watermarking and GIS vector map partitioning. Chapter 3 introduces the CRISP-DM methodology, and illustrates the way of applying this methodology in the domains of GIS vector map watermarking and GIS vector map partitioning. Chapter 4 presents the approach of selecting the embedding positions and implementing the embedding and extraction strategy. Chapter 5 presents the approach of using the bounding box properties for protecting the copyright of GIS vector data. Chapter 6 presents the k-means clustering approach in combination with the map bounding box property for protecting the copyright of GIS vector data. Chapter 7 presents the proposed metric for evaluating the watermarked vector maps. Chapter 8 presents the proposed spatial clustering approach for partitioning the GIS vector maps. Chapter 9 concludes the thesis.

Chapter 2

Literature Review

This Chapter gives an outline of previous research work on GIS map data. This thesis focuses mainly on two domains of applications: first is the digital vector map copyright protection and second is the digital vector map partitioning.

2.1 Digital Map Copyright Protection Approaches

Developments in computer technologies and Geographic Information Systems (GIS), i.e. computer-based systems for managing and displaying locational data related to positions on Earth's surface (Longley et al., 2005), increased the amount of digital vector maps that are available on the world wide web. GIS vector maps are highly accurate, they document attribute and topological information through the use of geometrical shapes, and are more compact in size compared with GIS raster maps such as satellite images. While these properties make GIS vector maps of high quality, their complexity and level of detail also means that they incur a high production cost[1].

GIS vector maps are widely used in environmental, social and economic applications such as disaster management, navigation, infrastructure and utilities allocation, and business planning. They are also used in military/security applications. Due to the value of these vector maps, their protection is necessary not only to prevent attackers gaining economic advantage (by using them without paying copyright fees), but also to prevent their unethical use in situations related to national and international security.

To prevent GIS vector maps being illegally modified and exchanged, different copyright techniques have been used, which fall mainly in two categories: encryption and information hiding. Encryption is part of a cryptographic system that has the purpose to protect the

[1] https://www.ordnancesurvey.co.uk/support/understanding-gis/raster-vector.html

content of a message/file. Information hiding is used in several sub-disciplines, of which the most important are *steganography* and *watermarking.* In steganography, the purpose of information hiding is to keep secret the existence of information, while in watermarking, the purpose is to make the hidden information imperceptible. The interested reader can find a more detailed distinction between these fields in (Lopez, 2002). From these approaches, the watermarking approach is the most popular for marking the copyright of GIS vector maps.

A digital map watermarking system consists of three modules: embedding, evaluation and extraction, as shown in Figure 2.1. The embedding module involves hiding the watermark bits inside the original map content and often involves the use of a secret key (which is needed in the extraction stage). The evaluation module is responsible for judging the quality of the map watermarking approach through particular evaluation metrics. The extraction module involves the extraction of the watermark and is important for making assertions about the data ownership.

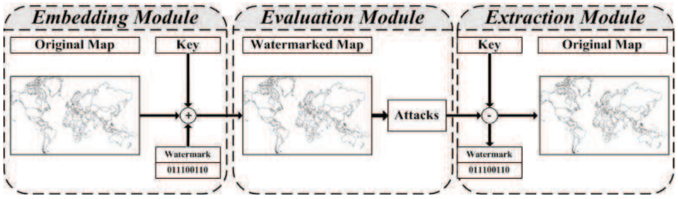


Figure 2.1: The General System of Digital Vector Map Watermarking.

GIS maps of raster data format (e.g. image) received more attention than digital GIS vector maps in watermarking research (Abbas and Jawad, 2013a; Wu et al., 2013c; Zheng et al., 2009); however, due to the importance of vector maps, the research for watermarking this type of maps has increased in the last decade. This article surveys and classifies the GIS vector map watermarking research articles published between 2000 and 2014, towards a thorough understanding of the-state-of-the-art, addressing significant limitations of previous review articles, highlighting the key differences between images and GIS vector maps, and giving recommendations for future research directions the research community should address.

The next section gives an overview and critical appraisal of previous review articles in the field of vector maps watermarking. Section 2.1.2 explains the methodology used for collecting the research articles that have been reviewed in this Chapter.

Section 2.1 classifies existing GIS watermarking methods and gives an overview of the distribution of published articles according to the categories of this classification.

2.1.1 Previous Reviews

A number of reviews were previously published in the area of digital map watermarking research. These reviews are discussed in the following [2]. Table 2.1 gives a summary of the review articles, including the number of references used and the period of time they covered.

Table 2.1: Summary table of the previous review articles

Articles	No. of used references	Period
Lopez (2002)	43	1993-2000
Chang et al. (2003)	26	1996-2002
Niu et al. (2006)	28	1999-2004
Niu et al. (2007)	23	2000-2004
Li et al. (2008a)	23	1996-2006
Zheng et al. (2009)	30	1999-2007
Zheng et al. (2010a)	31	1999-2007
Abbas and Jawad (2013a)	26	1998-2012
Wu et al. (2013c)	27	2001-2012

In 2002, Lopez (2002) presented a review article with 43 references covering work published until 2000 to analyze the state-of-the-art of digital watermarking research including images, vector, text and databases data formats. He highlighted some key differences between the watermarking research and other research work such as cryptography and steganography. He also reported some legal aspects in watermarking research for the United States and Europe regions.

In 2003, Chang et al. (2003) reviewed digital image watermarking research by utilizing 26 references published until 2002 to highlight possible ways of extending some image watermarking techniques to the context of 2D/3D vector map watermarking research.

In 2006, Niu et al. (2006), used 28 references published until 2004 to distinguish the features of vector map data from raster/ image data.

In 2007, Niu et al. (2007) used 23 references published until 2004 to outline some key features of GIS vector maps, reviewed the state-of-the-art of the vector map watermarking research and classified this research into three sub-research areas: robust watermarking, reversible data hiding and fragile watermarking.

In 2008,Li et al. (2008a) used 23 references published until 2006 to summarize the status and prospects of watermarking research in GIS vector maps in terms of the basic concept, watermark generation, real-time detection and embedding strategies.

In 2009, Zheng et al. (2009) used 30 references published until 2007 to classify digital map copyright protection schemes, and to propose some directions for further research. This review covered only the watermark embedding process.

[2] A number of review articles were not included in the discussion below because they were not published in English (Min et al., 2009; Peng, 2010; Sun et al., 2009b; Xu et al., 2007; Xun et al., 2004; Zhu et al., 2010))

In 2010, Zheng et al. (2010a) used 31 references published until 2007 and discussed some types of embedding techniques in the context of vector graphics. They highlighted some merits and drawbacks of a given set of image-based techniques with the purpose of suggesting some adaptations of these techniques for the vector map watermarking research context.

In 2013, two review articles are found in the literature. In the first one, Abbas and Jawad (2013a) used 26 references published until 2012 to review a set of digital vector map watermarking techniques, and to define some possible attacks for removing the embedded watermark. In the second review, Wu et al. (2013c) used 27 references published until 2012 to classify the map watermarking components into two modules: embedding location selection module and integrity decision module. Neither of the two reviews covered the entire watermarking process.

Although all previous review articles paid attention to either differentiating vector map data from raster image data or adapting some image-based watermarking techniques to the context of GIS vector maps, nevertheless, they suffer from two major drawbacks: (a) they do not cover the entire watermarking process and (b) they do not outline their search method, nor give an indication of their coverage in relation to the total number of published articles.

The watermarking system is composed of three main components: embedding, evaluation and extraction. A comprehensive overview of the current knowledge of the digital map watermarking research progress can only be obtained by reviewing all three components of the process. In addition, without a documented search method used for the selection of the articles to review, it would not be possible to ensure the coverage and relevance of the reviewed research.

In this survey, we attempt to address these two major drawbacks of the current surveys by considering the three components of the watermarking system, and providing details of the methodology used for selecting the articles included in this review work. This survey article covers 215 articles published between 2000 and 2014, thus being more comprehensive than any of the previous surveys on the subject.

2.1.2 Search Methodology

The search for relevant publications was performed using the following electronic libraries and databases: (i) Springer Digital Library, (ii) IEEE Xplore Digital Library, (iii) ACM digital library, (iv) Google Scholar, and (v) Elsevier Digital Library.

The search was limited to articles that have been published in English in the period between 1 January 2000 and 31 December 2014. It was done using a Boolean search containing the following terms: “GIS watermark” OR “zero watermark” OR “2D map watermark” OR

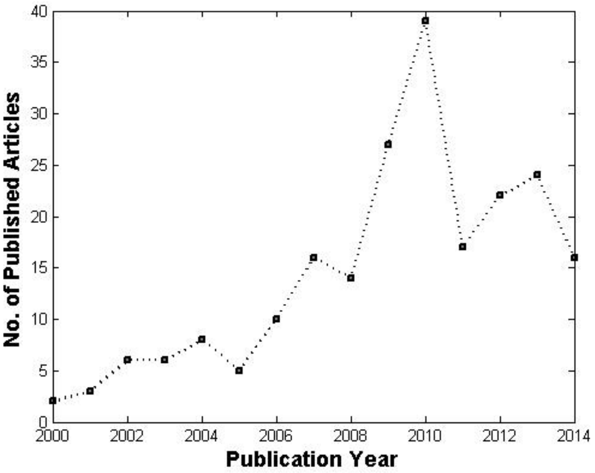


Figure 2.2: Publications in Watermarking from 2000 to 2014

"copyright protection" OR "vector data" OR "geospatial watermark" OR "vector data" OR "graph watermarking".

Initially, any article containing the search terms was considered as a potential candidate for including into the database of the GIS map watermarking publications. To supplement the automated search, a manual search was also done. The manual procedure involved searching the reference sections of the articles identified by the automated search. Any relevant references within those articles were followed up. Inclusion criteria for the review were any theoretical or applied work concerning an integration of the GIS vector data and watermarking/ copyright protection methods.

A number of articles were identified in the search as title-only articles without access to the full text (Calagna and Mancini, 2007; Min, 2007; Shi and Yang, 2008; Zhao et al., 2010b; Kan et al., 2010; Zhu et al., 2011). These were included in the count of published articles, but were not included in the classifications discussed in Section 2.1.

Through the search methodology described above, 215 articles were identified. Figure 2.2 presents the distribution of these articles by year, indicating a trend of growth in the number of publications.

Digital map copyright protection schemes are composed of three main modules: embedding, evaluation and extraction. Before describing in detail each of these modules, an overview of different terms used for different watermarking approaches is given in Table 2.2. Not all the reviewed articles explicitly stated which of these approaches they used - the table includes only the publications that explicitly stated their approach.

Table 2.2: List of the used terms in the watermarking research

Term	Definition	References
Zero watermarking	Aims to utilise some key characters of the host data in generating the watermark data	Abubahia and Cocea (2014), Cao et al. (2011), Zhang et al. (2009b), Du and Peng (2008), Li et al. (2008c), Xun et al. (2012), Wang et al. (2012b), Li et al. (2008b)
Adaptive watermarking	Attempts to shape the watermark according to some local characteristics of the original data	Zhang et al. (2008b), Peng et al. (2006)
Multiple watermarking	Refers to the use of more than one watermark to be embedded in the host data	Cui et al. (2013), Cao et al. (2010a), Xun et al. (2004), Bhanuchandar et al. (2013)
Reversible/Loss-less watermarking	Aims to achieve a good balance between the embedding process and the quality of the watermarked data, and aims to restore the original data after watermark extraction	Niu et al. (2007), Cao et al. (2013a), Zhao et al. (2010a), Wang and Men (2013), Abbas and Jawad (2013b), Wang et al. (2007), Deng and Xiao (2010), Men et al. (2010b), Fei et al. (2013), Wu (2012), Wu et al. (2009b), Peng et al. (2011), Jianguo et al. (2013a), Wang and Chiu (2012), Voigt et al. (2005), Men et al. (2009), Wang and Men (2012), Geng et al. (2012), Hu and Geng (2013), Wu and Wang (2009), Cao et al. (2013b), Cao et al. (2010b), Voigt et al. (2004), Neyman et al. (2013b), Jian-Guo et al. (2014), Jianguo et al. (2014), Peng et al. (2014a), Cao et al. (2014)
Classic watermarking	Refers to the field of applying watermarking techniques to the data of image type	Chang et al. (2003)
Additive watermarking	The process of adding the watermark bits directly to the value of the coordinates of vertices	Katzenbeisser and Petitcolas (2000)

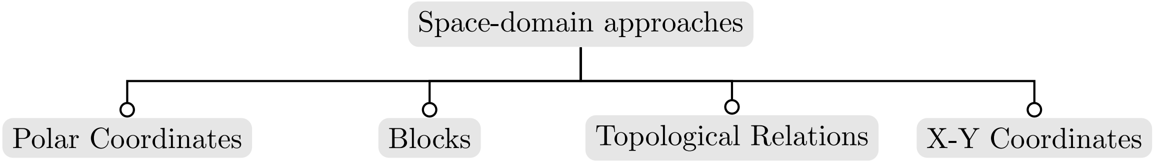


Figure 2.3: The classification of space-domain approaches

2.1.3 Watermark Embedding Module

The embedding module involves hiding the watermark bits inside the original map content without affecting the visual quality of the host map. The secret key (see Figure 2.1 in Chapter 1) should be used to enforce security and to prevent unauthorized parties from recovering and manipulating the watermark. This module involves both the embedding domains and the embedding strategies, which are discussed in the following subsections.

According to the embedding domain, a digital watermark can be embedded into two domains: space and transform domains. In the space domain, the watermark is embedded directly by modifying the values of vertices coordinates. In the transform domain, the watermark data is embedded not by directly modifying the coordinates of the vertices, but their transform coefficients instead. Space and transform domains are discussed in subsections 2.1.3 and 2.1.3, respectively.

Space-Domain Approaches

Space-domain watermarking approaches are applied to shift map's vertices within a predefined tolerance, and to embed the watermark based on different embedding strategies (see the next subsection). As shown in Figure 2.3, the embedding space could be represented by polar coordinates, blocks, topological relations or Cartesian coordinates. Table 2.3 shows the distribution of the published articles according to the different embedding spaces.

As shown in Table 2.3, the most popular approaches are the topological relations (35.5%) and the Cartesian coordinates (40%); 17.25% of the articles use blocks, while the least popular approach is the use of polar coordinates or angles (7.25%).

The topological relations embedding approaches refer to the process of inserting the watermark into map topologies instead of vertices' coordinates values (e.g. distance between the map vertices) to gain the advantage of preserving GIS data quality against rotation and translation attacks (Huo et al., 2011b; Xun et al., 2012); details about these and other attacks are given in subsection 2.1.4. Mean/ average distance length is the best known research example of topological relations embedding space (Abubahia and Cocea, 2014; Huo et al., 2011b; Xun et al., 2012).

Table 2.3: Digital Map Watermarking Schemes in the Space Domain

Embedding Space	No. of Articles	References
X/Y Coordinates	44	Wu and Wang (2009), Peng et al. (2006), Zhao et al. (2010a), Fei et al. (2013), Wu (2012), Cao et al. (2013b), Abbas and Jawad (2013b), Wang and Men (2012), Zheng et al. (2010d), Schulz and Voigt (2004), Yan et al. (2011), Fu et al. (2013), Huan and Yufeng (2009), Zhou and Bi (2004), Zhao et al. (2008), Yan and Li (2012), Wang et al. (2009b), Min et al. (2012), Wang et al. (2009c), Lee and Kwon (2010), Kim and Hong (2009), Zhao et al. (2013a), Voigt and Busch (2002), Marques et al. (2007), Shujun et al. (2007), Abbas et al. (2013), Ohbuchi et al. (2002), Che and Deng (2008), Magalhaes and Dahab (2009), Zhang and Li (2009), Pu et al. (2006), Haowen (2011a), Aybet et al. (2009), Abubahia and Cocea (2014), Hou et al. (2014), Yue et al. (2014), Wang et al. (2014), Neyman et al. (2014a), Peng et al. (2014b), Jianguo et al. (2014), Lee et al. (2014), Peng et al. (2014a), Ren et al. (2014a), Cao et al. (2014)
Topological Relations	39	Wang et al. (2012a), Xun et al. (2012), Wang et al. (2012b), Du and Peng (2008), Zhang et al. (2008b), Neyman et al. (2013b), Cao et al. (2013a), Cao et al. (2010b), Wang et al. (2007), Wu et al. (2009b), Zhou et al. (2006), Wu et al. (2009a), Zhou and Pan (2006), Wang and Xu (2003), Wang et al. (2009a), Kim et al. (2011), Huo et al. (2010), Baiyan et al. (2008a), Chuanjian et al. (2009), Yan and Li (2011), Bazin et al. (2007), Sun et al. (2010a), Cheng et al. (2010), Zhong et al. (2006), Peng et al. (2010), Bird et al. (2009), Wang et al. (2010b), Zhang et al. (2009a), Pu et al. (2009), Lafaye et al. (2012), Jia et al. (2006), Shao et al. (2005), Horness et al. (2007), Huo et al. (2011b), Huo et al. (2011a), Lee and Kwon (2013), Jiang et al. (2013), Abubahia and Cocea (2014), Suk-Hwan et al. (2014)
Blocks	19	Niu et al. (2007), Xun et al. (2012), Wang et al. (2012b), Li et al. (2008b), Peng et al. (2006), Wang and Men (2013), Wang et al. (2007), Wang et al. (2009b), Wang et al. (2009c), Ohbuchi et al. (2002), Zheng and You (2009), Kang et al. (2001b), Zhang et al. (2007), Kang et al. (2001a), Wu et al. (2010), Kang et al. (2002), Zheng et al. (2010c), Muttoo and Kumar (2012), Voigt and Busch (2003)
Polar Coordinates or Angles	8	Mouhamed et al. (2012), Raafat et al. (2013), Zhang et al. (2008a), Kim (2010a), Kim (2010b),Zhou et al. (2006), Wu et al. (2009a), Li et al. (2014a)

The Cartesian coordinates embedding approaches use directly the vertices' coordinates values for inserting the watermark (Zhao et al., 2013a). Most of these approaches utilize a specified digit place after the decimal point in the vertex coordinate value for adding the watermark bits, also defined as additive watermarking (Katzenbeisser and Petitcolas, 2000) and related to the Least Significant Bit embedding strategy (see the next subsection).

The blocks-based embedding approaches divide the vector map into a number of parts (blocks) which help in achieving more robustness against noise and simplification attacks (Niu et al., 2007). These approaches can maintain the fidelity of the watermarked vector map to some extent, and relatively locate the watermark bits in a certain block (Zheng and You, 2009).

The polar coordinates embedding approaches involve the use of another form of vertices' coordinates values for directly embedding the watermark. These approaches like Cartesian coordinates-based approaches achieve good robustness to attacks such as translation, rotation and equal scaling (Mouhamed et al., 2012; Wu et al., 2009a).

The advantages of space-domain schemes are: (a) simplicity ; (b) low computational complexity; (c) potential for high capacity of the watermark (i.e. the size of the watermark). The main disadvantage of space-domain schemes is the vulnerability to certain attack, i.e. low robustness.

Transform-Domain Approaches

Unlike the space domain, in the transform-domain embedding schemes the watermark is not embedded by modifying the coordinates of the vertices, but their transform coefficients. As shown in Figure 2.4, the most frequent types of transforms are: wavelet transform (WT), Fourier transform (FT) and cosine transform (CT). Table 2.4 shows the distribution of the published articles according to the transform domain.

Table 2.4: Digital Map Watermarking Schemes in the Transform Domain

Transform Type	No. of Articles	References
Wavelet	18	Li et al. (2012b), Deng and Xiao (2010), Peng et al. (2011), Li and Xu (2003), Zhu et al. (2008), Ling et al. (2012), Wang (2008), Mustafa (2011), Zhang and Wang (2011), Sangita and Venkatachalam (2012a), Zhang et al. (2010), Men et al. (2010a), Sangita and Venkatachalam (2012b), Li and Xu (2004), Im et al. (2008), Yang and Zhu (2007), Sangita and Venkatachalam (2012c), Jian-Guo et al. (2014)
Fourier	16	Huang and Gu (2006), Doncel et al. (2007), Kang and Zhang (2009), Sun et al. (2009a), Kitamura et al. (2001), Solachidis et al. (2000a), Vlachos et al. (2008), Tao et al. (2009), Solachidis et al. (2000b), Junfeng and Bing (2011), Lucchese et al. (2010), Huber et al. (2010), He et al. (2009), Giannoula et al. (2002), Solachidis and Pitas (2004), Neyman et al. (2014b)
Cosine	10	Niu et al. (2007), Voigt et al. (2005), Voigt et al. (2004), Wang et al. (2010a), Men et al. (2010c), Wu et al. (2013a), Zhang and Gao (2009), Tian et al. (2004), Liang et al. (2010), Wang et al. (2011)

WT is a kind of transform that analyzes the digital vector map into different bands and

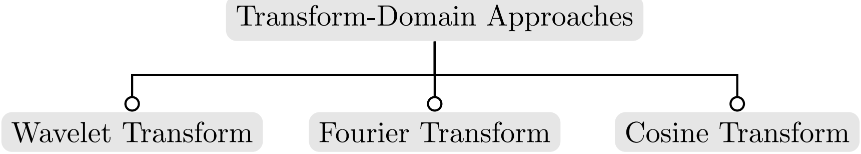


Figure 2.4: The classification of transform-domain approaches

levels. The wavelet-based method is robust against noise, rotation and scaling (Li and Xu, 2003).

FT is a digital transform that offers the possibility of controlling the frequencies of the host vector map, which helps in selecting the adequate positions for embedding the watermark bits into the vector map to meet the best compromise between invisibility and robustness. The main advantage of FT is its invariance property against some geometric attacks like translation, scaling and rotation (Junfeng and Bing, 2011; Lucchese et al., 2010).

CT is another digital transform that separate the vector map into parts of different frequency with respect to the vector map visual quality. The basic characteristic of CT is the high concentration of energy in low frequency coefficients with relative low computational cost (Men et al., 2010c; Zhang and Gao, 2009).

As shown in Table 2.4, the WT approach is the most popular approach used in 41% of the articles. CT is the second most popular at 36%, while FT is the least popular with 23% of articles reporting the use of this approach.

Transform-domain approaches are robust against geometric attacks such as rotation, translation and scaling; however, they have the disadvantages of being hard to implement and of having high computational complexity.

Embedding Strategies

There are a variety of strategies that have been used for the embedding process. These strategies are: significant bits, difference expansion, and quantization modulation. Table 2.5 lists the published articles according to the use of embedding strategies.

Table 2.5: Embedding Strategies

Embedding Strategy	No. of Articles	References
Least significant bits	9	Niu et al. (2007), Wang et al. (2012a), Wang et al. (2007), Haowen (2011a), Wang et al. (2009a), Jiang et al. (2013), Neyman et al. (2014a), Yan et al. (2011), Zhou et al. (2010)
Most significant bits	4	Wu et al. (2009a), Wang et al. (2010b), Lafaye et al. (2007a), Yue et al. (2014)
Difference Expansion	9	Niu et al. (2007), Wu and Wang (2009), Wang et al. (2007), Li et al. (2012a), Lafaye et al. (2007b), Hu and Geng (2013), Li et al. (2014a), Wu et al. (2009b), Neyman et al. (2013b)
Quantization Modulation	8	Peng et al. (2011), Wang et al. (2010b), Huo et al. (2011b), Lafaye et al. (2007a), Lafaye et al. (2007b), Guo and Peng (2010), He et al. (2009), Ohbuchi et al. (2003)

The significant bits embedding strategy refers to the process of selecting appropriate digits within the vertex coordinate value for inserting the watermark bit. This approach represents

43% of the published articles, and can be used in two different ways: least significant bits (LSB) (30%) or most significant bits (13%) (MSB).

LSB deals with the digits after the decimal point, and can be a useful hiding strategy in terms of: simplicity, invisibility, low computational time and allowing a large amount of watermark bits. LSB, however, is vulnerable to geometric distortion. LSB is mostly used in space-domain schemes with the exemption of the proposed scheme of (Li et al., 2012b) that used a LSB strategy in the wavelet transform-domain.

Some existing schemes used the MSB strategy that deals with the digits before the decimal point to control the modification of vertices' coordinate according to the precision tolerance. More precisely, this approach should meet two conditions: small modifications of the coordinates should not change the shape, and two adjacent shapes should not share the same identifier.

Difference expansion is a method for inserting the watermark into any kind of high correlation data (Wang et al., 2007). Digital vector maps consist of a sequence of the coordinates of the vertices. Due to the density of the vertices, the positions of two adjacent vertices are usually very close and the differences between their coordinates are very small. Consequently, the sequence of vertices' coordinates can also be considered high correlation data (Li et al., 2012a). Since higher correlation means lower distortions and higher capacity, the difference between two adjacent vertices is used as embedding space (Niu et al., 2007).

The quantization modulation strategy is a nonlinear method used to hide the watermark and scale some map objects to derive the watermarked data (Lafaye et al., 2007a). This embedding strategy offers a good performance in balancing the trade-off between watermark fidelity, robustness and capacity (Guo and Peng, 2010). An example of using the quantization modulation method is the watermark embedding according to odd-even index of map coordinates or topological relations (Huo et al., 2011b; Peng et al., 2011; Wang et al., 2010b).

2.1.4 Watermarking Evaluation Module

The evaluation module assesses the quality of the watermarking approach by measuring several aspects: (a) the quality of the map after the insertion of the watermark (fidelity); (b) the resistance of the watermarked map to attacks (robustness); (c) the coverage of the watermark (capacity); (d) the computational complexity of the approach (complexity) and (e) the security of the watermark locations within the map (security). These aspects are discussed in the following subsections.

Fidelity

Fidelity is defined as the relative similarity between the non-watermarked host object and the one after the watermarking operation (Abbas et al., 2013) and refers to the perceptual similarity between the watermarked data and its original data (Neyman et al., 2013a). The fidelity issue is a crucial problem in the digital maps watermarking research, as the watermarked maps need to preserve their quality.

Several metrics has been proposed in the literature, as shown in Table 2.6. Such metrics are: Root Mean Square Error (RMSE), Peak Signal to Noise Ratio (PSNR) (based on RMSE), Bit Error Rate (BER), Normalized Correlation (NC), Correspondence Ratio (CR) and Likelihood Ratio (LR), as well as Horizontal and vertical shift (HV shift).

Table 2.6: List of published articles according to the fidelity metrics

Used Metrics	No. of Articles	References
RMSE	29	Cao et al. (2013a), Wang and Chiu (2012), Wu and Wang (2009), Wu et al. (2009b), Zhao et al. (2010a), Cao et al. (2013b), Cao et al. (2010b), Wang et al. (2007), Wang and Men (2012), Geng et al. (2012), Huang et al. (2010), Kim et al. (2011), Kim and Hong (2009), Mouhamed et al. (2012), Raafat et al. (2013), Kim (2010a), Kim (2010b), Kang et al. (2001b), Kang et al. (2001a), Kang et al. (2002), Zhong et al. (2006), Peng et al. (2010), Huo et al. (2011a), Mustafa (2011), Hou et al. (2014), Neyman et al. (2014b), Neyman et al. (2014a), Li et al. (2014a), Peng et al. (2014a)
PSNR	12	Lee and Kwon (2010), Kang et al. (2001b), Kang et al. (2001a), Kang et al. (2002), Huo et al. (2011a), Mustafa (2011), Zhang and Wang (2011), Doncel et al. (2007), Huo et al. (2011b), Lucchese et al. (2010), Huang et al. (2010), Abubahia and Cocea (2014)
BER	7	Wang et al. (2009c), Voigt and Busch (2002), Pu et al. (2009), (Huo et al., 2011b), Wang et al. (2009a), Kitamura et al. (2001), Tao et al. (2009)
NC	5	Cao et al. (2013a), Cao et al. (2013b), Cao et al. (2010b), Raafat et al. (2013), Zhang and Wang (2011)
CR	4	Kim and Hong (2009), Kim (2010a), Kim (2010b), Kim et al. (2011)
LR	2	Zhou and Pan (2006), Giannoula et al. (2002)
HV shift	1	Neyman et al. (2013a)

The use of RMSE metric represents 48% of the published research, while the PSNR metric is used in 20% of the published research. 12% of the research approaches use the BER metric, and 8% use the NC metric. The least popular metrics in the published literature are CR (7%), LR (3%) and HV shift (2%).

Most of these metrics are borrowed from image watermarking and are based on theories of signal processing. These are not necessarily the most appropriate metrics for measuring the quality of the watermarked map, as will be discussed in Chapter 7.

Robustness

Robustness is the resilience of the inserted watermark to any processes (attacks) aimed at either removing or distorting it (Abbas et al., 2013; Lin and Li, 2010). Regarding the robustness requirements, watermarking schemes can be categorized into three categories: robust, fragile and semi-fragile schemes, as shown in Table 2.7.

Table 2.7: List of published articles according to the robustness degrees classification

Robustness Degree	No. of Articles	References
Robust	15	Xun et al. (2004), Xun et al. (2012), Wang et al. (2012b), Min et al. (2012), Abbas et al. (2013), Ohbuchi et al. (2002), Kim (2010a), Sangita and Venkatachalam (2012c), Wu et al. (2013a), Abubahia and Cocea (2014), Hou et al. (2014), Wang et al. (2014), Neyman et al. (2014b), Suk-Hwan et al. (2014), Ren et al. (2014b)
Fragile	9	Niu et al. (2007), Wang and Men (2013), Wang and Men (2012), Zheng and You (2009), Zheng et al. (2010c), Wang and Zhu (2012), Neyman et al. (2013b), Yue et al. (2014), Neyman et al. (2014a)
Semi-Fragile	4	Peng et al. (2010), Zhang and Gao (2009), Guo and Peng (2010), Ren et al. (2014a)

The digital watermark is robust if it withstands a designated manipulation on the vector map data (Abbas et al., 2013; Wu et al., 2013a; Xun et al., 2012).

Fragile watermarking allows the detection of any tampering with the vector map data (Wang and Men, 2012, 2013); however, any small change in the watermark would make it undetectable. This approach has a wide range of applications such as authentication and integrity protection of the vector maps (Zheng et al., 2009; Zheng and You, 2009). Semi-fragile schemes allow the detection of malicious tampering with the vector map data (Guo and Peng, 2010; Peng et al., 2010; Zhang and Gao, 2009); in these schemes, the watermark is still detectable after non-malicious transformations, however, it is not detectable after malicious attacks.

A successful attack refers to the success in removing the embedded watermark while preserving the validity of the vector map data (Niu et al., 2007). In literature, the attacks can be classified in two categories: (a) geometric attacks (Du and Peng, 2008; Wang et al., 2012b; Xun et al., 2012), and (b) signal operation attacks (Wang et al., 2012a; Wang and Men, 2012)).

The most known geometric attacks are rotation, translation, scaling and cropping – see Table 2.8. Rotation means turning the vector map around its center by a specific angle (Lee and Kwon, 2013). Translation means moving the whole map by a specific distance towards a specific direction (Xun et al., 2012). Scaling refers to altering the size of the map, in both axes by a specific value (Lee and Kwon, 2013). Cropping refers to the process of cutting some parts of the map. For vector maps, such attacks are virtually reversible transformations of coordinates where almost little or no information would be lost (Niu et al., 2007).

Signal operation attacks can be simplification, interpolation and reordering operations, or noise addition (Wang et al., 2012a; Wang and Men, 2012) – see Table 2.9. Simplification, also known as Douglas compression, is the process of removing some vertices. This is often used to enhance the speed of handling the vector map data (Niu et al., 2007). Interpolation is the process of adding new vertices into a digital vector map (Niu et al., 2007). The reordering operation involves changing the order of entities (i.e. points, polylines and polygons) in the map. This could be implemented by changing the order of points within a polyline/ polygon, or by changing the order of polylines/ polygons (Niu et al., 2007). Noise addition is used intentionally by the attacker to destroy the embedded watermark. Noise can also be added unintentionally by converting the map file into different formats (Abbas and Jawad, 2013a).

The resilience to both geometric and operational attacks is measured by comparing the extracted watermark with the original watermark by using the metrics that are shown in Table 2.10. These are different from the fidelity metrics, which compare the watermarked map with the original map. Thus, although the same metric could be used for both purposes, the robustness metrics focus on the watermark, while the fidelity ones focus on the map.

Many researchers use the same metrics for measuring both the robustness and the fidelity, as it can be seen by the overlap between Table 2.10 and Table 2.6, i.e. all metrics from Table 2.10 are also in Table 2.6 and several articles are in both tables, thus indicating that the same metric is used for the two different purposes.

From the robustness metrics, the use of the NC metric represents 47% of the published research. 27% of the published research are approaches that use the BER metric, while the use of PSNR metric is represented by 13% of the published research. The least popular metrics in the published literature are CR (8%) and RMSE (5%).

Capacity, Complexity and Security

The watermark capacity refers to the amount of embedded bits within the digital vector map (Abbas et al., 2013; Cao et al., 2013a), or the total number of vertices that carry the watermark bits (Abubahia and Cocea, 2014; Jianguo et al., 2012; Men et al., 2010c). Computational complexity refers to a specific formula for measuring the embedding algorithm complexity (Ramaswmay and Srinivasarao, 2010). In other words, it stands for measuring the required time for implementing the watermark embedding approach (Abubahia and Cocea, 2014; Dakroury et al., 2010). The security of a watermarking technique is defined as the level of unpredictability in identifying the watermark bits positions that are used to perform the watermark embedding process. A highly secure watermarking process would produce an output that does not contain any specific signatures that can be used to identify the watermark bits positions (Abbas et al., 2013). The secure watermarking approach should have a secret key for the embedded bits locations in the vector map vertices, to make it more difficult for an attacker to trace the distribution of the embedded watermark bits (Murti and Tadimeti, 2011).

Table 2.11 lists the published articles that discussed these aspects in their evaluation.

2.1.5 Watermark Extraction Module

The extraction module is important for data ownership verification. It is a very complicated task because of two main factors: (a) the wide variety of possible attacks that could take place before extraction (details on attacks are given in section 2.1.4), and (b) the (un)availability of the original map. According to the second factor, the watermarking approaches can be classified into three main categories: blind, semi-blind and non-blind approaches (Lin and Li, 2010). Table 2.12 lists the published articles according to these categories.

Blind/public approaches mean that the original map is not needed in the watermark extraction process, and this category represents 86% of published work. Semi-blind approaches refer to those approaches that do not use the original map, but use the original watermark in the watermark extraction process, and represent 3.5% of published work. Non-blind/private approaches mean that the original host data is needed in the watermark extraction process, and represent 10.5% of published work.

2.1.6 The Clustering based Map Protection Approaches

In recent years, a considerable amount of research has been carried out to solve the issue of copyright protection in the context of digital vector data (Abbas and Jawad, 2013a; Bhanuchandar et al., 2013; Wu et al., 2013b). A handful of research articles proposed watermarking approaches that use data mining tools in the context of digital vector data copyright protection (Huo et al., 2011b; Jianguo et al., 2013b; Haowen, 2011b; Raafat et al., 2013). Data mining tools could be very helpful in identifying the location for embedding the watermark to ensure the watermark resilience to the potential modifications. These approaches can be categorized into two main categories: clustering-based approaches (Huo et al., 2011b; Jianguo et al., 2013b; Haowen, 2011b) and classification-based approaches (Raafat et al., 2013).

There are few published watermarking methods that used data mining approaches to watermark GIS vector map data. In the following, we review these watermarking approaches and outline their advantages and disadvantages in relation to the evaluation metrics mentioned above.

Jianguo et al. (2013b) proposed an approach that used fuzzy spatial clustering analysis for embedding a 1-dimensional binary code watermark into a digital vector map. Their evaluation indicated that the algorithm outperforms some shifting, cosine transform and Fourier transform based algorithms in terms of data fidelity. Although this approach maintains the fidelity of the map data features, it is vulnerable to geometric attacks such as rotation, translation and scaling, which can easily result in the loss of the embedded watermark. This approach used optimization rules for selecting the watermark locations based on the coordinates' values and their associated attributes, which led to high fidelity, but low capacity.

Haowen (2011b) proposed a watermarking approach for embedding a 2-dimensional binary image watermark with a size of 32×96, into a vector point data set. The evaluation of the watermark robustness was measured by the similarity degree between the extracted watermark and the original watermark. In this approach, however, neither the capacity nor the trade-off between capacity and fidelity metrics were taken into account, which have crucial implications on the security of the digital map.

Huo et al. (2011b) used a k-means partitioning clustering method for inserting a watermark of 80 bits into GIS map data, based on ESRI shapefile format. They used the polygons' mean centres for locating the watermark bits into the GIS map. Also, they claimed that a high fidelity metric in terms of Peak Signal-to-Noise Ratio (PSNR) has been achieved by their scheme. Although the fidelity is relatively high, the capacity of the watermark was relatively low for the size of the map they used. Therefore, their approach, like the previous one, does not achieve a good trade-off between fidelity and capacity.

Another approach was presented by Lee and Kwon (2010) for watermarking CAD (computer-aided design) drawing by using the k-means++ clustering method. CAD drawing shares the vector structure format with geospatial data. In terms of speed and accuracy, k-means++ method outperforms the standard k-means in the way of selecting the initial centers. However, in this approach, only a small number of watermark bits can be embedded into the host data, thus leading to a low capacity.

Also in literature, only one article reported the use of classification-based data mining (Raafat et al., 2013). They proposed a watermarking approach for the authentication of 2D maps based on polar coordinates mapping, and used support vector machine classification to define optimum locations for embedding the watermark. Their approach only focused on geometric-based attacks, and did not consider signal-based attacks.

In contrast to the previous work, this thesis focuses more on using data mining approaches, clustering in specific, to address the issues of: identifying the watermark embedding

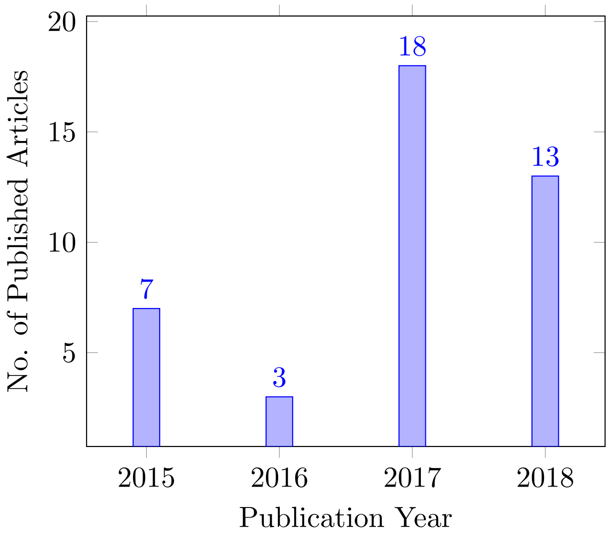


Figure 2.5: Publications in Watermarking from 2015 to 2018

position by using a clustering approach; improving the robustness to GIS vector relevant attacks/modifications; comparing different clustering approaches in identifying the watermark embedding locations in the map; defining a metric for measuring topological quality of polygon-based vector maps.

2.1.7 Recent Publications on Digital Map Copyright Protection 2015–2018

To up date the survey in this chapter, the articles that have been published in English in the period between 1 January 2015 and 31 December 2018 were included, as shown in Fig.2.5. The similar terms were used in Boolean search as illustrated in Section 2.1.2.

Apart from the articles that are published from this thesis, there are no newly articles that presented clustering based watermarking approach for protecting the digital vector map copyright.

2.2 The Large Map Clustering/Partition

Geographic information systems (GIS) are computer-based systems that facilitate the input, storage, manipulation and output of geographic location-based data (Longley et al., 2011). GIS data models are classified into two categories: raster and vector data models. Table 1.1 outlines the different properties of vector and raster data. In GIS context, satellite images are the most known example of the raster model. GIS vector data, which is the focus on this Chapter, has three components: spatial data, attribute data and index data. Spatial data describes the map itself and always takes the form of three basic geometrical entities, which are: points, lines/polylines and polygons. Points are used to define a single location of an object; they are used to represent real-world objects, such as bus stops, traffic lights and street lights. Lines/Polylines define linear objects; they can range from two-point lines to complex strings that have many vertices; lines are used to represent real-world objects, such as rivers and roads. Polygons define area-based objects; they can range from rectangles to multi-sided shapes with many vertices; polygons are used to represent real-world objects, such as lakes, shopping areas, buildings and city boundaries.

All these map entities are formed by many organized vertices; spatial data is actually a sequence of coordinates of these vertices based on a certain geographical coordinate system. The most used format of GIS spatial data is the ESRI (Environmental Systems Research Institute) shape file. The ESRI shape file (ESRI, 1998) has become an industry standard in geospatial data due to its compatibility, to some extent, with recently released GIS software products.

The attribute data describes the properties of map entities through links to the location data. Attributes can be, for example, names or matching addresses. The most known example of GIS attribute data format is the ESRI database file that is associated with the ESRI shape file and needs to have the same prefix as the shape file (ESRI, 1998). Last but not least, in the GIS context, the index data describes a file structure, such as total file length, for either spatial or attribute data. The ESRI index file (ESRI, 1998) is the best known example of index files.

Large regional partitioning is the process of dividing a large geographic area consisting of spatial objects, i.e points, lines or polygons (Joshi et al., 2012). This paper focuses on the polygon type of map entities. Partitioning a large map into sub-sets of spatial entities is not an easy task due to the nature of having spatial correlations and uneven distribution.

This problem has been investigated mostly in the redistricting field of GIS applications (Joshi et al., 2012; Photis, 2012; Bação et al., 2005). Some work has been done in the research of graph and GIS map clustering (Xu et al., 1998; Ng and Han, 2002; Cao et al., 2013c; Joshi et al., 2009b; Wang and Eick, 2014; Ericsson and WCDMA, 2011; Wang et al., 2010c; Barua et al., 2012; Eldawy et al., 2015; Boobalan et al., 2016; Kisore and Koteswaraiah, 2017; Liu et al., 2018; Gu et al., 2018), and more attention given to the clustering of polygon-based type of GIS maps (Zhang et al., 2005; Joshi et al., 2009a,c; Ji and Zhang, 2009; Jasim and Asadi, 2012).

The previous approaches focus on attribute data rather than spatial data, and used evolutionary computation techniques for optimizing the polygons' partitioning based on the attribute data, such as polygon area or polygon population.

According to the MapReduce model (Dean and Ghemawat, 2004; Puri et al., 2013; Li et al., 2014b; Eldawy and Mokbel, 2015; Araujo Neto et al., 2015), the workload balancing can be only achieved by distributing equal chunks of data records (i.e. number of vertices) to the MapReduce processors. Here the constraint is that the set of vertices that belong to the same polygon should not be separated in the mapping task (i.e. the first task of the MapReduce process).

In contrast to the previous work, this thesis focuses on spatial properties of GIS vector data, and considers both the nature of spatial data and the workload balancing requirement to extend the computation reliability for processing GIS complex data. This thesis proposes an approach to use evolutionary computation in combination with clustering techniques for developing workload balance based spatial clustering approach for partitioning GIS polygon based maps with massive number of vertices and complex shapes.

Table 2.8: List of published articles according to the robustness to a set of geometric attacks

Attack Type	No. of Articles	References
Rotation	60	Xun et al. (2012), Wang et al. (2012b), Du and Peng (2008), Wang and Chiu (2012), Wu (2012), Peng et al. (2011), Zhao et al. (2008), Zhao et al. (2013b), Abbas et al. (2013), Mouhamed et al. (2012), Raafat et al. (2013), Zhang et al. (2008a), Kim (2010a), Kim (2010b), Wu et al. (2009a), Bazin et al. (2007), Cheng et al. (2010), Zhong et al. (2006), Peng et al. (2010), Shao et al. (2005), Horness et al. (2007), Huo et al. (2011c), Wang et al. (2009a), Huo et al. (2011a), Lee and Kwon (2013), Wang and Xu (2003), Kim et al. (2011), Huo et al. (2010), Ling et al. (2012), Zhang and Wang (2011), Zhang et al. (2010), Im et al. (2008), Wu et al. (2013a), Zhang and Gao (2009), Liang et al. (2010), (Wang et al., 2011), Huang and Gu (2006), Kang and Zhang (2009), Solachidis et al. (2000a), Vlachos et al. (2008), Solachidis et al. (2000b), He et al. (2009), Giannoula et al. (2002), Solachidis and Pitas (2004), Sun et al. (2009a), Guo and Peng (2010), Sonnet et al. (2003), Pan et al. (2013), Zhanchuan et al. (2005), Tie et al. (2007), Ohbuchi et al. (2003), Abubahia and Cocea (2014), Hou et al. (2014), Wang et al. (2014), Neyman et al. (2014b), Jian-Guo et al. (2014), Li et al. (2014a), Lee et al. (2014), Peng et al. (2014a), Ren et al. (2014b)
Scaling	53	Xun et al. (2012), Wang et al. (2012b), Du and Peng (2008), Wang and Chiu (2012), Wu (2012), Peng et al. (2011), Zhao et al. (2008), Zhao et al. (2013b), Abbas et al. (2013), Mouhamed et al. (2012), Raafat et al. (2013), Zhang et al. (2008a), Kim (2010a), Kim (2010b), Cheng et al. (2010), Zhong et al. (2006), Peng et al. (2010), Horness et al. (2007), Huo et al. (2011c), Wang et al. (2009a), Huo et al. (2011a), Lee and Kwon (2013), Wang and Xu (2003), Kim et al. (2011), Huo et al. (2010), Zhang and Wang (2011), Ling et al. (2012), Zhang et al. (2010), Im et al. (2008), Zhang and Gao (2009), Liang et al. (2010), Wang et al. (2011), Kang and Zhang (2009), Solachidis et al. (2000a), Vlachos et al. (2008), Solachidis et al. (2000b), He et al. (2009), Solachidis and Pitas (2004), Sun et al. (2009a), Guo and Peng (2010), Sonnet et al. (2003), Pan et al. (2013), Zhanchuan et al. (2005), Tie et al. (2007), Ohbuchi et al. (2003), Abubahia and Cocea (2014), Wang et al. (2014), Neyman et al. (2014b), Jian-Guo et al. (2014), Li et al. (2014a), Peng et al. (2014b), Lee et al. (2014), Peng et al. (2014a)
Translation	53	Xun et al. (2012), Wang et al. (2012b), Du and Peng (2008), Wang and Chiu (2012), Wu (2012), Zhao et al. (2008), Zhao et al. (2013b), Abbas et al. (2013), Pu et al. (2006), Mouhamed et al. (2012), Raafat et al. (2013), Zhang et al. (2008a), Kim (2010a), Kim (2010b), Wu et al. (2009a), Bazin et al. (2007), Cheng et al. (2010), Peng et al. (2010), Shao et al. (2005), Horness et al. (2007), Huo et al. (2011c), Wang et al. (2009a), Huo et al. (2011a), Lee and Kwon (2013), Wang and Xu (2003), Kim et al. (2011), Huo et al. (2010), Im et al. (2008), Wu et al. (2013a), Zhang and Gao (2009), Liang et al. (2010), Wang et al. (2011), Huang and Gu (2006), Solachidis et al. (2000a), Vlachos et al. (2008), Solachidis et al. (2000b), He et al. (2009), Solachidis and Pitas (2004), Sun et al. (2009a), Guo and Peng (2010), Sonnet et al. (2003), Pan et al. (2013), Zhanchuan et al. (2005), Tie et al. (2007), Ohbuchi et al. (2003), Abubahia and Cocea (2014), Hou et al. (2014), Wang et al. (2014), Neyman et al. (2014b), Jian-Guo et al. (2014), Li et al. (2014a), Lee et al. (2014), Peng et al. (2014a)
Cropping	34	Xun et al. (2004), Xun et al. (2012), Jianguo et al. (2013a), Zhao et al. (2008), Min et al. (2012), Zhao et al. (2013b), Marques et al. (2007), Ohbuchi et al. (2002), Che and Deng (2008), Zhang and Li (2009), Pu et al. (2006), Kim (2010a), Wu et al. (2010), Bazin et al. (2007), Wang et al. (2010b), Pu et al. (2009), Lafaye et al. (2012), Huo et al. (2011c), Huo et al. (2011a), Lee and Kwon (2013), Jiang et al. (2013), Kim et al. (2011), Huo et al. (2010), Zhang et al. (2010), Sangita and Venkatachalam (2012b), Lafaye et al. (2007b), Yun et al. (2004), Lele et al. (2013), Ohbuchi et al. (2003), Abubahia and Cocea (2014), Hou et al. (2014), Jian-Guo et al. (2014), Jianguo et al. (2014), Lee et al. (2014)

Table 2.9: List of published articles according to the robustness to a set of operational attacks

Attack Type	No. of Articles	References
Douglas compression	47	Niu et al. (2007), Shao et al. (2006), Wang et al. (2012a), Cao et al. (2011), Cao et al. (2013a), Men et al. (2010b), Jianguo et al. (2013a), Men et al. (2009), Schulz and Voigt (2004), Min et al. (2012), Wang et al. (2009c), Zhao et al. (2013b), Ohbuchi et al. (2002), Zhang and Li (2009), Zhang et al. (2008a), Wu et al. (2009a), Voigt and Busch (2003), Chuanjian et al. (2009), Zhong et al. (2006), Peng et al. (2010), Wang et al. (2010b), Zhang et al. (2009a), Pu et al. (2009), Lafaye et al. (2012), Shao et al. (2005), Huo et al. (2011c), Huo et al. (2010), Zhu et al. (2008), Zhang et al. (2010), Sangita and Venkatachalam (2012b), Sangita and Venkatachalam (2012c), Wu et al. (2013a), Liang et al. (2010), Huang and Gu (2006), Lafaye et al. (2007b), Park et al. (2002), Yun et al. (2004), Li et al. (2011), Zhang (2010), Ohbuchi et al. (2003), Hou et al. (2014), Wang et al. (2014), Li et al. (2014a), Peng et al. (2014b), Lee et al. (2014), Peng et al. (2014a), Ren et al. (2014a)
Noise addition	37	Xun et al. (2004), Wang et al. (2012a), Peng et al. (2006), Wu (2012), Yan et al. (2011), Yan and Li (2012), Min et al. (2012), Wang et al. (2009c), Zhao et al. (2013b), Marques et al. (2007), Abbas et al. (2013), Ohbuchi et al. (2002), Pu et al. (2006), Kim (2010a), Kim (2010b), Zhang et al. (2007), Yan and Li (2011), Bird et al. (2009), Wang et al. (2010b), Pu et al. (2009), Lafaye et al. (2012), Horness et al. (2007), Huo et al. (2011a), Kim et al. (2011), Huo et al. (2010), Zhu et al. (2008), Mustafa (2011), Sangita and Venkatachalam (2012c), Vlachos et al. (2008), Sun et al. (2009a), Lafaye et al. (2007b), Yamada et al. (2006), Davydov et al. (2011), Peng et al. (2014b), Lee et al. (2014), Ren et al. (2014a), Ren et al. (2014b)
Interpolation	31	Niu et al. (2007), Wang et al. (2012a), Peng et al. (2006), Wang and Men (2013), Wang and Men (2012), Yan and Li (2012), Wang et al. (2009c), Marques et al. (2007), Ohbuchi et al. (2002), Zhang et al. (2008a), Kim (2010a), Kim (2010b), Zhang et al. (2007), Cheng et al. (2010), Peng et al. (2010), Shao et al. (2005), Huo et al. (2011c), Huo et al. (2011a), (Lee and Kwon, 2013), Kim et al. (2011), Huo et al. (2010), Yang and Zhu (2007), Sangita and Venkatachalam (2012c), Park et al. (2002), Wang and Zhu (2012), Ohbuchi et al. (2003), Hou et al. (2014), Wang et al. (2014), Li et al. (2014a), Peng et al. (2014b), Peng et al. (2014a)
Reordering	13	Shao et al. (2006), Wang and Men (2012), Marques et al. (2007), Ohbuchi et al. (2002), Shao et al. (2005), Huo et al. (2011c), Huo et al. (2011a), Solachidis et al. (2000a), Solachidis et al. (2000b), Solachidis and Pitas (2004), Ohbuchi et al. (2003), Peng et al. (2014b), Peng et al. (2014a)

Table 2.10: List of published articles according to the robustness metrics

Used Metrics	No. of Articles	References
NC	21	Mouhamed et al. (2012), Zhong et al. (2006), Mustafa (2011), Sangita and Venkatachalam (2012c), Xun et al. (2012), Wang et al. (2012b), Abbas et al. (2013), Zhang and Li (2009), Zhang et al. (2009a), Li and Xu (2003), Zhu et al. (2008), Sangita and Venkatachalam (2012b), Liang et al. (2010), Wang et al. (2011), Tao et al. (2009), Sun et al. (2010b), Tao et al. (2009), Raafat et al. (2013), Zhang and Wang (2011), Neyman et al. (2014b), Peng et al. (2014a)
BER	12	Lee and Kwon (2010), Huo et al. (2011a), Doncel et al. (2007), Wang et al. (2010b), Huo et al. (2011a), Huo et al. (2010), Huo et al. (2011c), Wang et al. (2009c), Voigt and Busch (2002), Pu et al. (2009), Wang et al. (2009a), Jianguo et al. (2014)
PSNR	6	Li and Xu (2003), Kang et al. (2001b), Kang et al. (2001a), Kang et al. (2002), Zhang and Wang (2011), Abubahia and Cocea (2014)
CR	4	Kim and Hong (2009), Kim (2010a), Kim (2010b), Kim et al. (2011)
RMSE	2	Zhong et al. (2006), Hou et al. (2014)

Table 2.11: List of published articles according to the evaluation metrics

Evaluation Metric	No. of Articles	References
Security	29	Du and Peng (2008), Li et al. (2008b), Jianguo et al. (2013a), Fei et al. (2013), Cao et al. (2013b), Fu et al. (2013), Kim and Hong (2009), Aybet et al. (2009), Kim (2010a), Zhou et al. (2006), Voigt and Busch (2003), Jia et al. (2006), Li et al. (2012a), Sonnet et al. (2003), Sun et al. (2010b), Wang and Zhu (2012), Li et al. (2009), Ramaswmay and Srinivasarao (2010), Dakroury et al. (2010), Peng et al. (2012), Murti and Tadimeti (2011), Matheus (2005), Zheng et al. (2011), Zheng et al. (2010b), Bisher et al. (2007), Dollner (2005), Sangita and Venkatachalam (2013), Lee et al. (2014), Suk-Hwan et al. (2014)
Capacity	29	Niu et al. (2007), Du and Peng (2008), Cao et al. (2013a), Wang and Chiu (2012), Wu and Wang (2009), Zhao et al. (2010a), Fei et al. (2013), Wang et al. (2007), Wu et al. (2009b), Peng et al. (2011), Geng et al. (2012), Voigt et al. (2004), Schulz and Voigt (2004), Ohbuchi et al. (2002), Men et al. (2010c), Kitamura et al. (2001), Sonnet et al. (2003), Pan et al. (2013), Yamada et al. (2006), Jianguo et al. (2012), Li et al. (2009), Ohbuchi et al. (2003), Hu and Geng (2013), Neyman et al. (2013b), Abubahia and Cocea (2014), Hou et al. (2014), Wang et al. (2014), Peng et al. (2014a), Cao et al. (2014)
Complexity/Time	10	Li et al. (2008b), Geng et al. (2012), Marques et al. (2007), Magalhaes and Dahab (2009), Bazin et al. (2007), Lucchese et al. (2010), Ramaswmay and Srinivasarao (2010), Dakroury et al. (2010), Abubahia and Cocea (2014), Wang et al. (2014)

Table 2.12: List of published articles according to the classification of extraction methods

Detection Type	No. of Articles	References
Blind	98	Wang et al. (2012a), Xun et al. (2012), Wang et al. (2012b), Du and Peng (2008), Li et al. (2008b), Cao et al. (2013a), Wang and Chiu (2012), Zhao et al. (2010a), Wu (2012), Cao et al. (2013b), Cao et al. (2010b), Abbas and Jawad (2013b), Wang et al. (2007), Wu et al. (2009b), Wang and Men (2012), Zheng and You (2009), Zheng et al. (2010d), Schulz and Voigt (2004), Yan et al. (2011), Zhou and Bi (2004), Zhao et al. (2008), Yan and Li (2012), Wang et al. (2009c), Lee and Kwon (2010), Kim and Hong (2009), Zhao et al. (2013b), Voigt and Busch (2002), Shujun et al. (2007), Abbas et al. (2013), Zhang and Li (2009), Pu et al. (2006), Aybet et al. (2009), Mouhamed et al. (2012), Raafat et al. (2013), Zhang et al. (2008a), Kim (2010a), Kim (2010b), Zhou et al. (2006), Wu et al. (2009a), Kang et al. (2001b), Kang et al. (2001a), Wu et al. (2010), Kang et al. (2002), Zhou and Pan (2006), Wang et al. (2009a), Kim et al. (2011), Huo et al. (2010), Baiyan et al. (2008b), Chuanjian et al. (2009), Yan and Li (2011), Bazin et al. (2007), Cheng et al. (2010), Zhong et al. (2006), Peng et al. (2010), Bird et al. (2009), Wang et al. (2010b), Pu et al. (2009), Lafaye et al. (2012), Jia et al. (2006), Shao et al. (2005), Horness et al. (2007), Huo et al. (2011c), Huo et al. (2011a), Jiang et al. (2013), Li and Xu (2003), Ling et al. (2012), Mustafa (2011), Zhang et al. (2010), Im et al. (2008), Zhang and Gao (2009), Liang et al. (2010), Wang et al. (2011), Kang and Zhang (2009), Solachidis et al. (2000a), Vlachos et al. (2008), Solachidis et al. (2000b), Junfeng and Bing (2011), Lucchese et al. (2010), He et al. (2009), Giannoula et al. (2002), Solachidis and Pitas (2004), Sun et al. (2009a), Tao et al. (2009), Huber et al. (2010), Lafaye et al. (2007b), Sonnet et al. (2003), Zuo et al. (2010), Yamada et al. (2006), Li et al. (2010), Pan et al. (2013), Park et al. (2002), Abubahia and Cocea (2014), Hou et al. (2014), Yue et al. (2014), Wang et al. (2014), Neyman et al. (2014a), Peng et al. (2014b), Suk-Hwan et al. (2014)
Non-blind	12	Marques et al. (2007), Ohbuchi et al. (2002), Zhang et al. (2007), Zheng et al. (2010c), Zhang et al. (2009a), Zhu et al. (2008), Zhang and Wang (2011), Sangita and Venkatachalam (2012b), Sangita and Venkatachalam (2012c), Huang and Gu (2006), Doncel et al. (2007), Kitamura et al. (2001)
Semi-blind	4	Li et al. (2012b), Magalhaes and Dahab (2009), Haowen (2011a), Neyman et al. (2014b)

Table 2.13: List of recent published articles according to the publication year

Publication Year	No. of Articles	References
2015	7	Wang et al. (2015a), Abubahia and Cocea (2015b), Cao et al. (2015), Abubahia and Cocea (2015a), Zope-Chaudhari et al. (2015), Wang et al. (2015b), Peng and Yue (2015)
2016	3	Wang et al. (2016b), Chen et al. (2016), Wang et al. (2016a)
2017	18	Peng et al. (2017c), Wang (2017), Abubahia and Cocea (2017), Lan and Peng (2017), Tulapurkar et al. (2017), Hassan and Mohammed (2017), Zope-Chaudhari et al. (2017), Liu et al. (2017), Su et al. (2017), Wang et al. (2017), Lin et al. (2017), Peng et al. (2017a), Peng et al. (2017b), Zhang et al. (2018b), Yan et al. (2017), Zhang et al. (2018a), Jang et al. (2016), Wang and Zhao (2018a)
2018	13	Wang and Kankanhalli (2018), Wang and Zhao (2018b), Lin et al. (2018), Wang et al. (2018), Da et al. (2018), Hou et al. (2018), Gaata (2018), Tulapurkar et al. (2018), Abubahia and Cocea (2018), Qiu et al. (2018a), Qiu et al. (2018b), Zhou et al. (2018), Bansal and Upadhyaya (2018)

Chapter 3

Methodology

This Chapter gives an outline of research methodology that were adopted in the study. It discusses and elaborates how the CRISP-DM methodology could be used to effectively perform spatial data mining tasks. This Chapter outlines the use of CRISP-DM methodology in: identifying the watermark embedding position by using a clustering approach; improving the robustness to GIS vector relevant attacks/modifications; comparing different clustering approaches in identifying the watermark embedding locations in the map; defining a metric for measuring topological quality of polygon-based vector maps; and developing workload balance based spatial clustering approach for partitioning GIS polygon based maps with massive number of vertices and complex shapes.

3.1 CRISP-DM Methodology

CRISP-DM (Cross-Industry Standard Process for Data Mining) is a data mining process model that describes commonly used approaches that expert data miners use to tackle problems - see reference (Shearer, 2000). CRISP-DM provides a structured approach to planning a data mining project and, as shown in Table 3.1, it is composed of six phases: business understanding, data understanding, data preparation, modeling, evaluation, and deployment.

3.1.1 Business understanding

The business understanding phase involves several key steps, including determining business objectives, assessing the situation, determining the data mining goals, and producing the project plan.

3.1.2 Data understanding

The data understanding phase involves four main steps, including the collection of initial data, the description of data, the exploration of data, and the verification of data quality.

3.1.3 Data preparation

There are five steps that might be involved in data preparation, they are: the selection of data, the cleansing of data, the construction of data, the integration of data, and the formatting of data.

3.1.4 Modelling

Modelling steps include the selection of the modelling technique, the generation of test design, the creation of models, and the assessment of models.

3.1.5 Evaluation

In the evaluation phase, the key steps are: the evaluation of results, the process review, and the determination of next steps.

3.1.6 Deployment

In the deployment phase, the key steps are: plan deployment, plan monitoring and maintenance, the production of the final report, and review of the project.

The following subsections will add more details to the way of implementing the CRISP-DM methodology in the context of GIS vector data mining. The deployment phase is not used in this research because the research process is carried out on the iteratively basis. For all the research objectives, the deployment phase consists of written form of report (i.e. this PhD thesis and published papers) - see publication list in page 5. To avoid repetition, this is omitted from the following subsections.

3.2 The Watermark Embedding Locations Identification

3.2.1 Business understanding

One of the main research issues of digital vector map data is defined by copyright protection, and digital watermarking is a potential solution to this issue. This phase involves the following

Business Understanding	Data understanding	Data preparation	Modeling	Evaluation	Deployment
Determine business objectives. Assess the situation. Determine data mining goals. Produce a project plan.	Collect the initial data. Describe data. Explore data. Verify data quality.	Select data. Clean data. Construct data. Integrate data. Format data.	Select modeling technique. Generate test design. Build model. Assess model.	Evaluate results. Review process. Determine next steps.	Plan deployment Plan monitoring & maintenance. Produce final report. Review project.

Table 3.1: Cross-Industry Standard Process for Data Mining (Shearer, 2000)

key steps of: determining the business objective; assessing the situation; determining the data mining goals; and producing a project plan.

Step 1 – Determine the Business Objective: The business objective is to protect the valuable assets of vector maps from illegal use by solving the issue of vector map copyright protection. Moreover, increasing the security of the watermarked vector maps by employing more suitable data mining methods.

Step 2 – Assess the Situation: In recent years, a considerable amount of research has been carried out to solve the issue of copyright protection in the context of digital vector map data. However, few research attempts proposed watermarking methods that use data mining tools in the context of digital vector data copyright protection. These attempts show promising results for advancing the research of vector map copyright protection.

Step 3 – Determine the Data Mining Goals: The goal is set as using data mining techniques for identifying the best locations/positions for embedding the watermark. These embedding positions should balance the trade-off between the fidelity and capacity.

Step 4 – Produce a Project Plan: The project plan involves: the use of k-medoids partition clustering and compare its deployment with a previous watermarking scheme in which k-means partition clustering is used; using clustering approach to identify the embedding position that satisfy the trade-off balance; and using MATLAB programming for the watermark embedding and extraction processes and measuring the computational time. The kind of data were used is in the free vector map format of ESRI shapefiles.

3.2.2 Data understanding

This phase involves the following key steps of: collecting the initial data; describing the data; exploring the data; and verifying the data quality

Step 1 – Collect the Initial Data: The used maps are downloaded from the Map Library website [1]; which contains the maps of administrative boundaries for the African countries.

Step 2 – Describe the Data: The used map data format is shapefile (.shp); which was developed by ESRI company [2]. The shapefile is a popular format used in GIS applications due to its prominent characteristics. These characteristics can be summarized as:

- It requires less storage space than image data;
- It has considerable speed in drawing and editing shapes;
- It stores spatial features, in the form of coordinates, and their attribute information;

[1] http://www.mapmakerdata.co.uk/v4/Base%20map%20data.Africa.htm

[2] http://www.esri.com

- It supports all types of geometry, i.e points, lines and polygons;
- It is easy to read and write.

Step 3 – Explore the Data: The used maps represent different sizes in terms of number of polygons. These maps represent the administrative areas map of three countries: Tunisia (27 polygons), Swaziland (53 polygons) and Burundi (132 polygons).
Step 4 – Verify Data Quality: The used polygon based maps are checked to verify that the targeted polygons have no missing attributes, and also checked against the following topology rules: map polygons have no disclosures; map polygons are not overlapped; and map polygons have no gaps.

3.2.3 Data preparation

This phase involves the following key steps of: selecting Data and constructing Data. The step of cleaning data is not applicable, here, due to the high quality of data. Moreover, there is no need for data integration or formatting in the data preparation phase.
Step 1 – Select Data: The selected data are represented by X and Y coordinates of the polygons' vertices.
Step 2 – Construct Data: The data is constructed by the calculation of polygons' centers.

3.2.4 Modeling

This phase involves the following key steps of: selecting the modeling technique; generating test design; building and assessing the model.
Step 1 – Select the Modeling Technique: Two modeling techniques are involved at this step. One is the PAM (Partitioning Around Medoids) based clustering method; which is used for identifying the locations for embedding the watermark bits. Another one is the odd-even indexing; which is used for inserting the watermark bits into the identified locations.
Step 2 – Generate Test Design: To assess the difference introduced by the k-medoids partition clustering method, a set of experiments have been carried out in regards to: fidelity, robustness, computational time, capacity and security metrics. The fidelity metric aims to measure the perceptual similarity between the watermarked map data and the original map data. The robustness reflects the watermark's resistance to a set of attacks or modifications. The computational time refers to the time period, in seconds, for embedding the watermark bits into the host map. The capacity refers to the number of watermark bits that is embedded in the host map. The security refers to the used procedure for keeping the Watermark bits' position secure and inaccessible by the potential attacker.

Step 3 – Build the Model: These experiments were carried on the three maps and different proportions of map size, i.e. 25%, 33% and 50%, to verify the consistency of results. Also, the odd-even indexing is used for modelling the watermark insertion process.
Step 4 – Assess the Model: The fidelity is measured by using PSNR (Peak Signal to Noise Ratio), in decibels. The computational time was measured in seconds. The watermark capacity is expressed by the number of vertices that carry the watermark bits.

3.2.5 Evaluation

This phase involves the following key steps of: evaluating the results; reviewing the process; and determining next steps.
Step 1 – Evaluate Results: The experimental results show that both k-medoids and kmeans clustering approaches result in high fidelity, while the k-medoids based clustering approach achieves a more balanced trade-off between capacity and fidelity, as well as better computational efficiency due to the k-medoids characteristics.
Step 2 – Review Process: Both clustering and embedding approaches are checked and reviewed repeatedly; to ensure they are working appropriately.
Step 3 – Determine Next Steps: The next step is to improve the watermark resilience to a set of GIS map relevant attacks/modifications such as: the simplification (removing some vertices from GIS vector data) and interpolation (adding new vertices to GIS vector data).

3.3 The Watermark Resilience Improvement

3.3.1 Business understanding

This phase involves the following key steps of: determining the business objective; assessing the situation; determining the data mining goals; and producing a project plan.
Step 1 – Determine the Business Objective: The business objective is to improve the watermark resilience to a set of GIS map relevant attacks/modifications such as: simplification (removing some vertices from GIS vector data) and interpolation (adding new vertices to GIS vector data).
Step 2 – Assess the Situation: Although the previous approach, in subsection 3.2.2, achieved a considerable improvement in terms of the balance between capacity and fidelity, like the other approaches, it is still vulnerable to simplification and interpolation attacks, and has not been shown to work on larger maps.
Step 3 – Determine the Data Mining Goals: The data mining goal is to use a particular property of vector data, called a bounding box, in combination with k-medoids approach in

Chapter 4, and to addresses the vulnerability to the two aforementioned attacks, while also preserving a good trade-off between fidelity and capacity.

Step 4 – Produce a Project Plan: The project plan involves: using k-medoids clustering method in combination with the bounding box centres to strengthen the watermark resilience against simplification and interpolation attacks, and to preserve the trade-off between fidelity and capacity. Both watermark embedding and extraction processes are implemented by using MATLAB version R2013b (8.2.0.701). The free vector maps, in ESRI shapefile format, are used to assess the proposed approach.

3.3.2 Data understanding

This phase involves the following key steps of: collecting the initial data; describing the data; exploring the data; and verifying the data quality.

Step 1 – Collect the Initial Data: The used maps are downloaded from the Map Library website; which contains the maps of administrative boundaries for the African countries, as illustrated in subsection 3.2.2.

Step 2 – Describe the Data: As discussed in subsection 3.2.2, the used data format is shapefile (.shp); which was developed by ESRI company. The shapefile is a popular format used in GIS applications due to its prominent characteristics.

Step 3 – Explore the Data: The used GIS maps are polygon-based maps that represent administrative boundaries of 3 countries in Africa: Benin (222 polygons), Angola (501 polygons) and Burkina Faso (1046 polygons). These GIS vector maps are freely available, in ESRI shapefile format.

Step 4 – Verify Data Quality: The used polygon based maps are checked to verify that the targeted polygons have no missing attributes, and also checked against the following topology rules: map polygons have no disclosures; map polygons are not overlapped; and map polygons have no gaps.

3.3.3 Data preparation

This phase involves the following key steps of: selecting data and constructing data. The step of cleaning data is not applicable, here, due to the high quality of data. Moreover, there is no need for data integration or formatting in the data preparation phase.

Step 1 – Select Data: The selected data are represented by X and Y coordinates of the polygons' bounding boxes.

Step 2 – Construct Data: The data is constructed by the calculation of polygons' bounding box centres.

3.3.4 Modeling

This phase involves the following key steps of: selecting the modelling technique; generating test design; building and assessing the model.

Step 1 – Select the Modeling Technique: Two modelling techniques are involved at this step. One is the k-medoids in combination with the bounding box property; which is used for identifying the locations for embedding the watermark bits, and adding more robustness to the watermark. Another one is the odd-even indexing; which is used for inserting the watermark bits into the identified locations.

Step 2 – Generate Test Design: To assess the difference introduced by the k-medoids method in combination with bounding boxes property, a set of experiments have been carried out in regards to: fidelity and capacity. The fidelity metric aims to measure the imperceptibility of the watermark and reflects its degree of invisibility. Capacity refers to the number of vertices that carry the watermark bits.

Step 3 – Build the Model: These experiments were carried on the three maps and different proportions of map size, i.e. 25%, 33% and 50%, to verify the consistency of results. Also, the odd-even indexing is used for modelling the watermark insertion process.

Step 4 – Assess the Model: The fidelity is measured by using PSNR (Peak Signal to Noise Ratio), in decibels. The watermark capacity is expressed by the number of vertices that carry the watermark bits.

3.3.5 Evaluation

This phase involves the following key steps of: evaluating the results; reviewing the process; and determining next steps.

Step 1 – Evaluate Results: The experimental results show that both k-medoids and kmeans clustering approaches result in high fidelity, while the k-medoids based clustering approach achieves a more balanced trade-off between capacity and fidelity, as well as better computational efficiency due to the k-medoids characteristics.

Step 2 – Review Process: Both k-medoids based clustering in combination with the polygon bounding box centres and odd-even based embedding approaches are checked and reviewed repeatedly; to ensure they are working appropriately.

Step 3 – Determine Next Steps: The next step is to compare k-medoids and k-means clustering approaches in combination with the bounding box property to investigate their influences against the use of the bounding box property for protecting the copyright of GIS vector maps.

3.4 Map Properties and Clustering Approaches Comparison

3.4.1 Business Understanding

This phase involves the following key steps of: determining the business objective; assessing the situation; determining the data mining goals; and producing a project plan.

Step 1 – Determine the Business Objective: The business objective is to investigate the role of the bounding box property in addressing the vulnerability to simplification and interpolation attacks, and to investigate if the trade-off between fidelity and capacity is preserved.

Step 2 – Assess the Situation: In previous work we showed that using k-medoids clustering in combination with the bounding box property of vector maps in the embedding process leads to increased robustness against simplification (removing vertices from vector data) and interpolation (adding new vertices to the data) attacks.

Step 3 – Determine the Data Mining Goals: The data mining goal is to use k-means clustering approaches in combination with the use of polygon bounding box centres; to improve the watermark resilience to the defined set of GIS map relevant attacks/modifications.

Step 4 – Produce a Project Plan: The project plan involve: using k-means clustering method in combination with the bounding box centres to strengthen the watermark resilience against simplification and interpolation attacks, and to preserve the trade-off between fidelity and capacity. Both watermark embedding and extraction processes are implemented by using MATLAB version R2013b (8.2.0.701). The free vector maps, in ESRI shapefile format, are used to assess the proposed approach.

3.4.2 Data Understanding

This phase involves the following key steps of: collecting the initial data; describing the data; exploring the data; and verifying the data quality.

Step 1 – Collect the Initial Data: The used maps are downloaded from the Map Library website; which contains the maps of administrative boundaries for the African countries, as illustrated in subsection 3.2.2.

Step 2 – Describe the Data: As discussed in subsection 3.2.2, the used data format is shapefile (.shp); which was developed by ESRI company. The shapefile is a popular format used in GIS applications due to its prominent characteristics.

Step 3 – Explore the Data: The used GIS maps are polygon-based maps that represent administrative boundaries of 3 countries in Africa: Benin (222 polygons), Angola (501 polygons) and Burkina Faso (1046 polygons). These GIS vector maps are freely available, in ESRI shapefile format.

Step 4 – Verify Data Quality: The used polygon based maps are checked to verify that the targeted polygons have no missing attributes, and also checked against the following topology rules: map polygons have no disclosures; map polygons are not overlapped; and map polygons have no gaps.

3.4.3 Data preparation

This phase involves the following key steps of: selecting data and constructing data. The step of cleaning data is not applicable, here, due to the high quality of data. Moreover, there is no need for data integration or formatting in the data preparation phase.

Step 1 – Select Data: The selected data are represented by X and Y coordinates of the polygons' bounding boxes.

Step 2 – Construct Data: The data is constructed by the calculation of polygons' bounding box centres.

3.4.4 Modeling

This phase involves the following key steps of: selecting the modelling technique; generating test design; building and assessing the model.

Step 1 – Select the Modeling Technique: Two modelling techniques are involved at this step. One is the k-means in combination with the bounding box property; which is used for identifying the locations for embedding the watermark bits, and adding more robustness to the watermark. Another one is the odd-even indexing; which is used for inserting the watermark bits into the identified locations.

Step 2 – Generate Test Design: To assess the difference introduced by the k-means method in combination with bounding boxes property, a set of experiments have been carried out in regards to: fidelity and capacity. The fidelity metric aims to measure the imperceptibility of the watermark and reflects its degree of invisibility. Capacity refers to the number of vertices that carry the watermark bits.

Step 3 – Build the Model: These experiments were carried on the three maps and different proportions of map size, i.e. 25%, 33% and 50%, to verify the consistency of results. Also, the odd-even indexing is used for modelling the watermark insertion process.

Step 4 – Assess the Model: The fidelity is measured by using PSNR (Peak Signal to Noise Ratio), in decibels. The watermark capacity is expressed by the number of vertices that carry the watermark bits.

3.4.5 Evaluation

This phase involves the following key steps of: evaluating the results; reviewing the process; and determining next steps.
Step 1 – Evaluate Results: The experimental results show that the advantages of using the bounding box property are maintained even with k-means clustering approach, and argue that they would hold regardless of the method used for identifying the watermark embedding locations in the map.
Step 2 – Review Process: Both k-means based clustering in combination with the polygon bounding box centres and odd-even based embedding approaches are checked and reviewed repeatedly; to ensure they are working appropriately.
Step 3 – Determine Next Steps: The next step is to quantify fidelity measure that consider the nature of GIS vector maps.

3.5 Topological Quality Measurement

3.5.1 Business Understanding

This phase involves the following key steps of: determining the business objective; assessing the situation; determining the data mining goals; and producing a project plan.
Step 1 – Determine the Business Objective: The business objective is to define a metric that allows comparisons between watermarked maps of different sizes and of different watermark sizes, and, thus, can be used to assess the quality of watermarked vector maps.
Step 2 – Assess the Situation: Unlike image watermarking field of research, measuring the loss of precision only with error metrics, without checking the topology preservation, is not a good way to evaluate watermarked vector map quality.
Step 3 – Determine the Data Mining Goals: The data mining goal is to use k-means clustering approach in combination with polygon bounding box centres, odd-even indexing and map topology rules; to define a metric that can be used to assess the quality of watermarked vector maps.
Step 4 – Produce a Project Plan: The project plan involves using k-means clustering approach in combination with polygon bounding box centres, odd-even indexing and map topology rules to define and test the proposed metric in evaluating the topological quality of watermarked vector maps. Both watermark embedding and metric testing are implemented by using MATLAB version R2014b (8.4.0.150421) on a 64-bits Windows-PC. The free vector maps, in ESRI shapefile format, are used to assess the proposed metric.

3.5.2 Data Understanding

This phase involves the following key steps of: collecting the initial data; describing the data; exploring the data; and verifying the data quality.

Step 1 – Collect the Initial Data: The used maps are downloaded from the Map Library website; which contains the maps of administrative boundaries for the African countries, as illustrated in subsection 3.2.2.

Step 2 – Describe the Data: As discussed in subsection 3.2.2, the used data format is shapefile (.shp); which was developed by ESRI company. The shapefile is a popular format used in GIS applications due to its prominent characteristics.

Step 3 – Explore the Data: Four categories of datasets (of two maps each) combining high and low numbers of polygons and vertices are used, respectively:

- Dataset 1 includes maps with small number of polygons and small number of vertices. The used GIS maps under this category are: Morocco (47 polygons and 7523 vertices) and Swaziland (53 polygons and 7678 vertices)
- Dataset 2 includes maps with small number of polygons and large number of vertices. The used GIS maps under this category are: Congo-Brazzaville (46 polygons and 12511 vertices) and Guinea (56 polygons and 21304 vertices)
- Dataset 3 includes maps with large number of polygons and small number of vertices. The used GIS maps under this category are: Egypt (129 polygons and 5992 vertices) and Chad (347 polygons and 19542 vertices)
- Dataset 4 includes maps with large number of polygons and large number of vertices. The used GIS maps under this category are: Ghana (138 polygons and 243329 vertices) and Burkina Faso (351 polygons and 113996 vertices)

Step 4 – Verify Data Quality: The used polygon based maps are checked to verify that the targeted polygons have no missing attributes, and also checked against the following topology rules: map polygons have no disclosures; map polygons are not overlapped; and map polygons have no gaps.

3.5.3 Data preparation

This phase involves the following key steps of: selecting data and constructing data. The step of cleaning data is not applicable, here, due to the high quality of data. Moreover, there is no need for data integration or formatting in the data preparation phase.

Step 1 – Select Data: The selected data are represented by X and Y coordinates of the polygons' bounding boxes.
Step 2 – Construct Data: The data is constructed by the calculation of polygons' bounding box centres.

3.5.4 Modeling

This phase involves the following key steps of: selecting the modelling technique; generating test design; building and assessing the model.
Step 1 – Select the Modeling Technique: The used modelling techniques involve the use of k-means clustering approach in combination with polygon bounding box centres, odd-even indexing and map topology rules to define and test the proposed metric in evaluating the topological quality of watermarked vector maps.
Step 2 – Generate Test Design: To assess the proposed metric, a set of experiments have been carried out in regards to the map topology rules of: disclosures, gaps and overlaps. Disclosure occurs when the coordinates of the first and the last vertex, within same polygon, are different. The gap occurs when having voids within a polygon or between neighboring polygons. The overlap means that the interior of polygons must not overlap, and they can only share edges or vertices.
Step 3 – Build the Model: These experiments were carried on the three maps and different proportions of map size, i.e. 25%, 33% and 50%, to verify the consistency of results. Also, the odd-even indexing is used for modelling the watermark insertion process.
Step 4 – Assess the Model: The topological quality, of watermarked vector maps, is measured by averaging the three metrics that counts the numbers of disclosures, gaps and overlaps.

3.5.5 Evaluation

This phase involves the following key steps of: evaluating the results; reviewing the process; and determining next steps.
Step 1 – Evaluate Results: The experimental results indicate that the metrics allow comparisons between watermarked maps of different sizes and of different watermark sizes, and, thus, can be used to asses the quality of watermarked vector maps.
Step 2 – Review Process: The proposed metric is checked and reviewed repeatedly; to ensure the metric efficiency in assessing the topological distortions in the context of watermarked vector maps.

Step 3 – Determine Next Steps: The next step is to exploit the characteristics of evolutionary computation approaches for developing a clustering approach that consider the nature of GIS vector maps' properties.

3.6 Workload Balanced GIS Map Clustering

3.6.1 Business Understanding

This phase involves the following key steps of: determining the business objective; assessing the situation; determining the data mining goals; and producing a project plan.
Step 1 – Determine the Business Objective: The business objective is to increase the computation performance for processing GIS polygon based maps with massive number of vertices and complex shapes.
Step 2 – Assess the Situation: The current approaches focus more on attribute data rather than spatial data. They use evolutionary computation techniques for optimizing the polygons' partitioning based on the attribute data; e.g polygon area or polygon population.
Step 3 – Determine the Data Mining Goals: The business objective is to develop workload balanced spatial clustering approaches, by using evolutionary computation method that considers the nature of spatial data to increase the computation performance for processing GIS polygon based maps with massive number of vertices and complex shapes.
Step 4 – Produce a Project Plan: The project plan involve using polygons' bounding box centres in combination with evolutionary computation approach to increase the computation performance for processing GIS polygon based maps.

3.6.2 Data Understanding

This phase involves the following key steps of: collecting the initial data; describing the data; exploring the data; and verifying the data quality.
Step 1 – Collect the Initial Data: The used maps are downloaded from the Map Library website; which contains the maps of administrative boundaries for the African countries, as illustrated in subsection 3.2.2.
Step 2 – Describe the Data: As discussed in subsection 3.2.2, the used data format is shapefile (.shp); which was developed by ESRI company. The shapefile is a popular format used in GIS applications due to its prominent characteristics.
Step 3 – Explore the Data: Four categories of datasets (of two maps each) combining high and low numbers of polygons and vertices are used, respectively:

- Dataset 1 includes maps with small number of polygons and small number of vertices. The used GIS maps under this category are: Djibouti (11 polygons and 676 vertices) and Somalia (88 polygons and 3175 vertices)
- Dataset 2 includes maps with small number of polygons and large number of vertices. The used GIS maps under this category are: Guinea (56 polygons and 21304 vertices) and Zimbabwe (81 polygons and 32382 vertices)
- Dataset 3 includes maps with large number of polygons and small number of vertices. The used GIS maps under this category are: Liberia (305 polygons and 10521 vertices) and Chad (347 polygons and 19542 vertices)
- Dataset 4 includes maps with large number of polygons and large number of vertices. The used GIS maps under this category are: Burkina Faso (351 polygons and 113996 vertices) and Ethiopia (575 polygons and 261880 vertices)

Step 4 – Verify Data Quality: The used polygon based maps are checked to verify that the targeted polygons have no missing attributes, and also checked against the following topology rules: map polygons have no disclosures; map polygons are not overlapped; and map polygons have no gaps.

3.6.3 Data preparation

This phase involves the following key steps of: selecting data and constructing data. The step of cleaning data is not applicable, here, due to the high quality of data. Moreover, there is no need for data integration or formatting in the data preparation phase.

Step 1 – Select Data: The selected data are represented by X and Y coordinates of the polygons' bounding boxes.

Step 2 – Construct Data: The data is constructed by the calculation of polygons' bounding box centres.

3.6.4 Modeling

This phase involves the following key steps of: selecting the modelling technique; generating test design; building and assessing the model.

Step 1 – Select the Modeling Technique: The modelling technique involves using polygons' bounding box centres in combination with evolutionary computation approach to define and test the proposed clustering approach for partitioning the GIS vector maps.

Step 2 – Generate Test Design: To assess the proposed clustering approach, a set of experiments have been carried out in regards to: different number of partitions and different evolutionary computation operators.
Step 3 – Build the Model: These experiments were carried on the eight maps and different number of partitions and different evolutionary computation operators.
Step 4 – Assess the Model: The fitness function is identified by calculating the standard deviation value. The standard deviation is calculated at level of the set of map partitions, where each set contain number of polygons. The smallest value of the standard deviation indicate the better balance between the partitions according to the total number of vertices.

3.6.5 Evaluation

This phase involves the following key steps of: evaluating the results; reviewing the process; and determining next steps.
Step 1 – Evaluate Results: The experimental results show the capability of the proposed clustering approach in addressing the issue of workload balancing in the context of GIS vector map data.
Step 2 – Review Process: The proposed clustering approach is checked and reviewed repeatedly; to ensure the reliability of the proposed approach in partitioning different sizes and shapes of vector maps.
Step 3 – Determine Next Steps: The next step is experiment more on addressing the issue of time complexity. Also, more investigation will be undertaken on the possibility of using different vertices weights for implementing various partitioning aspects in the context of GIS vector maps.

Chapter 4

The Watermark Embedding Locations Identification

In recent years, the compelling need for protecting the copyright of digital vector maps has become an emergent topic within the GIS (Geographic Information System) research community that stemmed from the rapid growth of intelligent tools and devices (Chang, 2012; Longley et al., 2011). One of the main economic, social and legal aspects of using GIS data is defined by copyright protection (Wu et al., 2013b). This has been enforced and administrated internationally by UN-WIPO (United Nations - World Intellectual Property Organization), by considering the digital maps as software products (Fenwick and Locks, 2010).

Unlike other physical data, digital data has its own features of being intangible and dynamic, which make it easy to be copied, modified or distributed through different media such as CDs, DVDs, USBs or via internet servers (Bainbridge, 2014).

In the digital context, the copyright offers an exclusive right to secure and protect the livelihood of original work producers. This helps prevent illegal digital copies being distributed on internet web sites and used instead of the original productions. In case of copyright dispute, digital watermarking can be used for claiming ownership. Digital watermarking has been proposed, in recent years, as an effective solution to combat this threat of piracy.

In watermarking research, digital multimedia data such as images, text, audio and videos received more attention by researchers and scholars than digital vector map data (Bhanuchandar et al., 2013). The spatial structure and topological relations within the vector map type of data are features that make it different from other multimedia data. The key difference between vector data and image data, as illustrated in Table 4.1, is the small redundancy available to hide the watermark due to the precision intolerance of vertices' coordinates. In addition, digital vector data has great economic significance due to the value of its accurate content (Jian-Guo et al., 2014). Digital maps are developed for complex data, which makes them suitable to be used in many applications where accuracy is important, such as navigation, strategic planning, military services and decision making (Chang, 2012).

Table 4.1: Vector data versus image/raster data

Aspect	**Vector Data**	**Image Data**
Feature Representation	Points/Lines/Polygons	Array of pixels
Resolution Determination	Precise coordinates	Pixel size
Efficiency	Sparse data	Dense data
Spatial Relations	Exist	Do not exist
Storage Requirement	Small space	Large space
Redundancy Size	Small	Large

In recent years, a considerable amount of research has been carried out to solve the issue of copyright protection in the context of digital vector data, e.g., (Abbas and Jawad, 2013a; Bhanuchandar et al., 2013; Wu et al., 2013b). A handful of research papers proposed watermarking methods that use data mining tools in the context of digital vector data copyright protection (Huo et al., 2011b; Jianguo et al., 2013b; Haowen, 2011b; Raafat et al., 2013). These methods can be categorized into two main categories: clustering-based methods (Huo et al., 2011b; Jianguo et al., 2013b; Haowen, 2011b) and classification-based methods (Raafat et al., 2013). In the literature, clustering-based methods are more prevalent than classification-based methods; consequently, we focus on clustering methods.

In particular, we advocate that the clustering method used has an influence on the security of the watermarked vector map, where security is measured through specific evaluation metrics, which are outlined in Section 4.1. More specifically, we propose the use of a k-medoids partition clustering approach; there are several implementations of this approach, of which the PAM (Partitioning Around Medoids) method is the most popular (Han et al., 2009, 2012). This clustering method is useful in identifying the location for embedding the watermark. In addition, the use of clustering techniques can also ensure a good distribution of the embedding locations across the vector map. In this Chapter, we investigate whether the use of PAM leads to a more secure watermarked map in comparison with a k-means partition clustering method.

The rest of this Chapter is organized as described in the following. In section 4.1, a detailed overview of relevant previous work is presented. Section 4.2 describes the geospatial

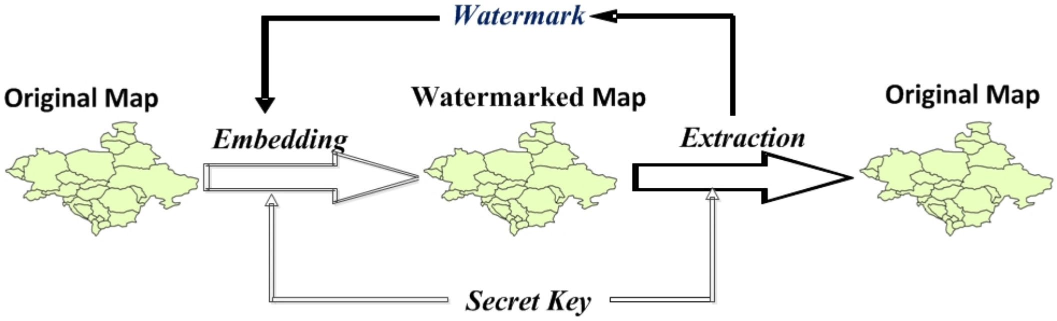


Figure 4.1: The General System of Digital Vector Map Watermarking.

data format and the platform that has been utilized for the experimental evaluation of the approach proposed in this Chapter. Section 4.3 presents the full explanation of the proposed approach including: selecting the embedding positions and implementing the embedding and extraction strategy. Section 4.4 describes the experiments and discusses the findings. Section 4.5 concludes this Chapter.

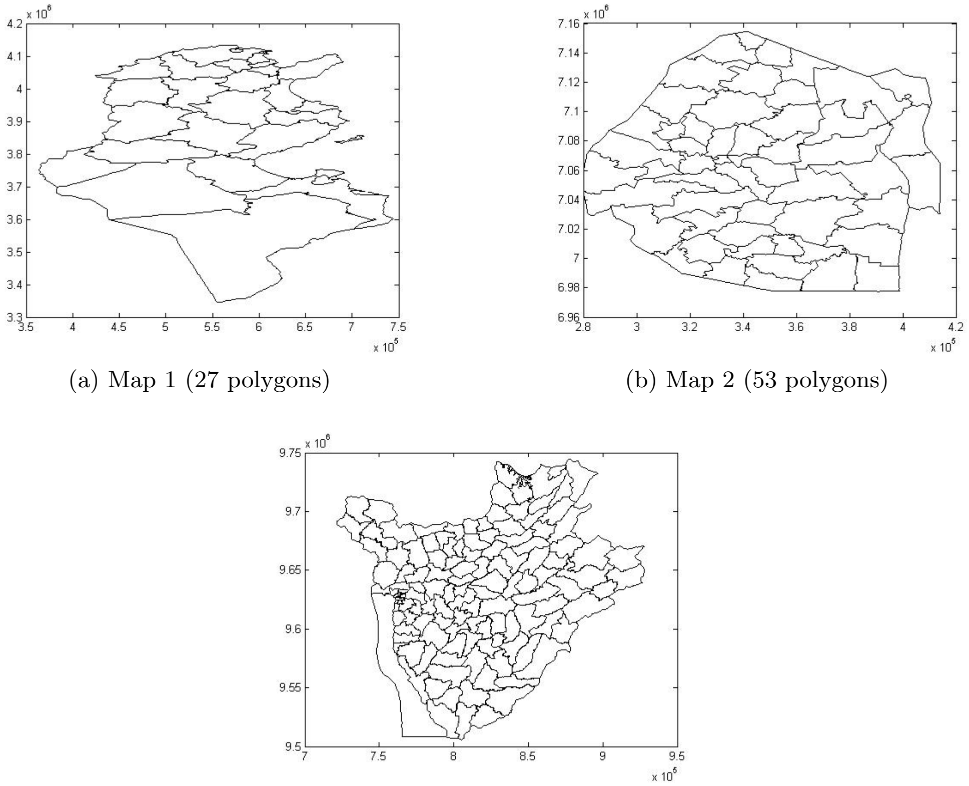
4.1 Related Work

In GIS vector map data, a sequence of vertices' coordinates is used to represent geographical locations of the digital map object, which can take one of three types of geometry shapes: point, polyline and polygon(Abbas and Jawad, 2013a).

A digital vector map watermarking system, as shown in Fig. 4.1, consists of two substantial stages: embedding and extraction. The embedding stage refers to the process of inserting copyright information, which is called a watermark, into the host data.

In the former stage, one or more secret keys are used for adding more security to the embedded locations in the digital map, as well as keeping these locations unknown to potential attackers. The stage of watermark extraction aims to obtain the watermark from the host data by using the aforementioned secret key(s). The purpose of extraction is to obtain the watermark so that the original map can be retrieved.

In the literature, digital map watermarking algorithms are classified into two main types: spatial domain and transform domain. Spatial domain algorithms are concerned with embedding the watermark directly into different spaces, such as Cartesian coordinates, polar coordinates, blocks and topology relations. Transform domain algorithms deal with inserting the watermark into a transformed form of data. The most frequently used data transformations in the watermarking context are wavelet transform, Fourier transform and cosine transform (Abbas and Jawad, 2013a).



(a) Map 1 (27 polygons)

(b) Map 2 (53 polygons)

(c) Map 3 (132 polygons)

Figure 4.2: The maps used in the experiments.

The security of a watermarked map is evaluated by looking at four aspects: capacity, fidelity, computational time and robustness (Abbas and Jawad, 2013a).

Capacity refers to the number of bits that can be embedded in the host data (Cox et al., 2007; Niu et al., 2006). In addition to the number of embedded bits in the host data, these bits should be spread across the whole map in order to provide more robustness to cropping attacks, which refer to cutting parts of the host map (Zhao et al., 2013a). The use of clustering methods in the process of watermarking ensure a good spread by identifying locations for embedding throughout the map.

Fidelity refers to the fact that the watermark embedding process should not affect the quality of the host data and that the watermark should not be noticeable to the human eye (Nin and Ricciardi, 2013).

There is a trade-off between capacity and fidelity: inserting many watermark bits, i.e., increased capacity, leads to a loss of fidelity or quality of the host map (Abbas and Jawad, 2013a). Consequently, there is a need to balance the capacity of the map with its fidelity to achieve good security without loss of quality.

Computation time/complexity refers to the period of time that is required to perform the embedding process and obtaining the watermarked data (Barni and Bartolini, 2004).

Robustness refers to the ability of the watermaked map to withstand any kind of modifications, called attacks, to the host data (Cox et al., 2007). Examples of these attacks are geometric modifications processes such as rotation, translation and scaling (Huo et al., 2011b).

In this Chapter we argue that the partition clustering method used in the process of identifying the location for embedding the watermark has an influence on the security of the watermarked map measured in terms of capacity, fidelity, computational time and robustness. To investigate this, we propose a k-medoids approach and compare it with the approach of Huo et al. (Huo et al., 2011b) as a best representative of partition clustering-based watermark embedding approaches, because it takes into consideration both the trade-off between capacity and fidelity, and the robustness to geometric attacks.

4.2 Materials

This section describes the data format used and the platform that has been utilized for implementing the proposed approach in this Chapter.

A particular data format is used, which is called shapefile (.shp) and was developed by ESRI[1], a major company supplying Geographic Information System (GIS) software and geodatabase management applications which are widely used in over 200 countries. The shapefile is a popular format used in GIS applications due to its prominent characteristics. These characteristics can be summarized as (ESRI, 1998):

1. It requires less storage space than image data;
2. It has considerable speed in drawing and editing shapes;
3. It stores spatial features, in the form of coordinates, and their attribute information;
4. It supports all types of geometry, i.e points, lines and polygons;
5. It is easy to read and write.

Three maps covering three countries in Africa were used for the research presented in this Chapter, which are illustrated in Fig. 4.2; these maps are freely available from the Map

[1] http://www.esri.com/

Library website[2]. As shown in Fig. 4.2, we used the administrative areas map of three countries: Tunisia (27 polygons), Swaziland (53 polygons) and Burundi (132 polygons).

For the watermark embedding and extraction processes, and for measuring the computational time, MATLAB version R2013b (8.2.0.701) and license No. 484067 was used with the personal computer of Windows 7 (32-bits) and RAM of 2GB. For more information regarding MATLAB, see the Mathworks website[3].

4.3 The Proposed Approach

The proposed approach aims to assess the influence of k-medoids in comparison with k-means partition clustering on the trade-off between the capacity and fidelity metrics, as well as on the computational complexity and robustness metrics. For this purpose, we use the approach of Huo et al. (Huo et al., 2011b) and vary the two aspects highlighted in Fig. 4.3.

The GIS map watermarking approach, as illustrated in Fig. 4.3, consists of determining the embedding positions (first four steps) and embedding the watermark into the host map by applying odd-even indexing method (last three steps). These steps are explained in sub-sections 4.3.1 and 4.3.2, respectively. Regarding the watermark extraction, the last part of the watermarking process illustrated in Fig. 4.1, it is explained in sub-section 4.3.3.

4.3.1 The Watermark Embedding Positions

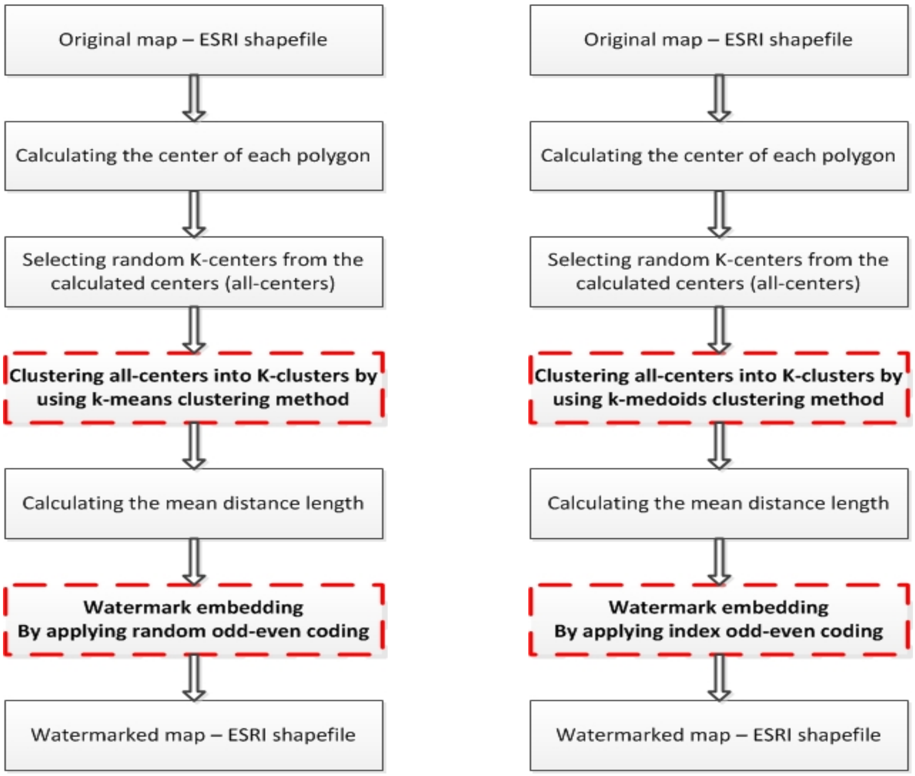
Embedding positions refer to a set of map locations to be modified by inserting the watermark bits. In the work of Huo et al. (2011b), the process of selecting the embedding positions includes the following steps: calculation of polygons' centers, selecting random centers to be used as initial cluster centers for k-means clustering and selecting the centers of polygons to be used for embedding. The last step is accomplished for each cluster, by choosing the closest point to the center of the cluster. Finally, the mean distance length is calculated, to be used in the watermark embedding method.

The approach uses k-medoids instead of k-means and is presented in detail below.

- The calculation of polygons' centers: Polygons' centres are calculated in both axes (Elhami et al., 2001), as shown in Equations (4.1) and (4.2), by summing all vertices coordinates for each polygon and then dividing by the number of vertices minus one; the subtraction of one is due to the last vertex coordinates being the same as for the first vertex, according to the polygon shapefile format (ESRI, 1998).

[2] `http://www.mapmakerdata.co.uk.s3-website-eu-west-1.amazonaws.com/library/stacks/Africa/index.htm`

[3] `http://www.mathworks.co.uk/`


Original map – ESRI shapefile
Calculating the center of each polygon
Selecting random K-centers from the calculated centers (all-centers)
Clustering all-centers into K-clusters by using k-means clustering method
Calculating the mean distance length
Watermark embedding By applying random odd-even coding
Watermarked map – ESRI shapefile
Original map – ESRI shapefile
Calculating the center of each polygon
Selecting random K-centers from the calculated centers (all-centers)
Clustering all-centers into K-clusters by using k-medoids clustering method
Calculating the mean distance length
Watermark embedding By applying index odd-even coding
Watermarked map – ESRI shapefile

(a) Huo et.al Scheme (Huo et al., 2011b)

(b) The Proposed Approach

Figure 4.3: The Compared Embedding Framework.

$$x_c = \sum_{i=1}^{n-1} \frac{x_i}{n-1} \tag{4.1}$$

$$y_c = \sum_{i=1}^{n-1} \frac{y_i}{n-1} \tag{4.2}$$

where: x_c and y_c are the coordinates of polygon's center in both x and y axes respectively; n is the number of all vertices within the same polygon; i is the order of the vertex in the polygon.

- Clustering of polygons' centers. In contrast to the scheme of Huo et al. (2011b), the proposed approach uses a k-medoids based partition clustering method called PAM (Partitioning Around Medoids). PAM method works firstly by arbitrarily assigning initial representative objects, called seeds. Subsequently, it replaces the seeds by other

Algorithm 1 k-medoids (PAM) method for partitioning based on medoid.

```
Input:
k: the number of clusters,
D: a data set containing n objects.
Output: A set of k clusters.
Method:
    – arbitrarily choose $k$ objects in $D$ as the initial representative objects or seeds;
    – repeat
    – assign each remaining object to the cluster with the nearest representative object in terms of Euclidean distance;
    – randomly select a non-representative object, $O_{random}$;
    – compute the total cost, $S$, of swapping representative object, $O_j$, with $O_{random}$;
    – if $S < 0$ then swap $O_j$ with $O_{random}$ to form the new set of $k$ representative objects;
    – until no change;
```

representative objects iteratively. This process continues until the resulting medoids, i.e. clusters' representative objects, can not be improved or changed (Han et al., 2012, 2009). Polygons' centers are clustered into k-clusters and the resulting medoids are kept as a secret key ($key1$). The k-medoids mechanism (Han et al., 2012) is summarized in **Algorithm 1**. Unlike k-means, the centers of clusters are actual polygon centers, not artificial points which did not exist in the initial data set of polygon centers (Han et al., 2012).

The k-medoids method outperforms the k-means method by its robustness to outliers, i.e. objects that are far from the majority of the data within the same cluster. Both k-means and k-medoids need the number of clusters to be specified by the user (Han et al., 2012; Kolatch, 2001), which has an advantage of controlling the number of watermark embedding locations, which have a good influence on increasing the capacity.

Another specific advantage of applying k-medoids method in the context of watermarking GIS vector data, is that clusters' centers are actual data points from the map data sets. In contrast, the clusters' centers in the k-means method are artificial points, which introduces an element of approximation that is not present in the k-medoids algorithm.

- Calculating the mean distance length. The mean-distance length is the average of distances from the polygon's center to each of its surrounding vertices within the same polygon (Huo et al., 2011b; Xun et al., 2012). The values of mean-distance lengths are kept as another secret key ($key2$) and used as targeted positions for watermark embedding. Equation (4.3) demonstrates the way of calculating the mean-distance length of polygons that are selected by using the k-medoids partition clustering method.

$$L_c = \frac{1}{n-1} \sum_{v=1}^{n-1} \sqrt{(x_c - x_v)^2 + (y_c - y_v)^2} \tag{4.3}$$

where: L_c is the mean distance length; n is the number of vertices in a polygon; v is the vertex order; x_c and y_c are the center coordinates in x and y axes, respectively; x_v and y_v are the vertex coordinates in x and y axes, respectively.

4.3.2 The Watermark Embedding Method

The watermark is structured on the basis of the zero watermark concept (Huo et al., 2011b). Zero watermarking aims to utilize some key characteristics of the host map data in order to generate a more robust watermark. In this case, the characteristic of the host map data that is used is the mean-distance length of polygons. The watermark is constructed by adding or subtracting a bit value of 1 from the mean-distance length of polygons.

The watermark is embedded by applying an odd-even indexing condition (Huo et al., 2011b; Baiyan et al., 2008a), as outlined in Equation (4.4). The index of each mean-distance value is used in this approach, instead of using an additional random sequence proposed by (Huo et al., 2011b), to simplify the implementation and also to have more consistent positions for embedding the watermark.

This indexing plays a vital role in combination with the clustering process by:

1. Maintaining the security of the watermark position by storing the index values as a key instead of utilizing a random sequence that is not relevant to the used data;

2. Ensuring that all selected polygons are used as watermark carriers to attain a maximum value of capacity;

3. The ability to increase the watermark capacity while preserving the map fidelity, whereas the use of random sequence and indexing condition in (Huo et al., 2011b) will limit that choice of control.

$$W_i = \begin{cases} T+1, & \text{if} \quad OES(I) = odd \\ T-1, & \text{if} \quad OES(I) = even \end{cases} \tag{4.4}$$

where: W_i is the ith bit value of the watermark; OES stands for Odd-Even Status; I is the order index of the mean-distance length value in the matrix; T is the value of the 4th digit of the mean-distance length value, after the decimal point (Huo et al., 2011b).

As shown in Equation (4.4), the watermark is embedded by comparing the OES (Odd-Even Status) of both I and T variables. The conditions are set based on two scenarios as following:

- If the OES of I is odd, 1 will be subtracted from the value of T.
- In contrast, if the OES of I is even, 1 will be added to the value of T.

After applying the OES to change the values of the mean-distance length L_c, the new values will be represented by L_c^*. This new mean-distance length values are stored as an additional secret key ($key3$), to secure the positions in which the watermark is embedded. Following, the change rate α_c is calculated as depicted in Equation (4.5):

$$\alpha_c = \frac{L_c^*}{L_c} \tag{4.5}$$

The change rate α_c is used to change all vertices of polygons that belong to each cluster's center on the basis of embedding condition, as given in equations 4.6 and 4.7:

$$v_x^* = \alpha_c v_x + x_c(1 - \alpha_c) \tag{4.6}$$

$$v_y^* = \alpha_c v_y + y_c(1 - \alpha_c) \tag{4.7}$$

where: v_x^* and v_y^* are the new vertices' coordinates after embedding the watermark according to the aforementioned condition, in Equation (4.4).

4.3.3 The Watermark Extraction Method

The watermark extracting process is flexible and quite similar to the embedding process. It is performed by using the keys stored during the embedding process. Firstly, we calculate the center of each polygon, then dividing all centers into k number of clusters by using the k-medoids partition clustering algorithm. In the next stage, the mean-distance length is computed for the watermarked map in the same way as given in the watermark embedding process.

Table 4.2: The compared results of C (Capacity) and F (Fidelity) between the proposed approach and the approach of Huo et al. (2011b).

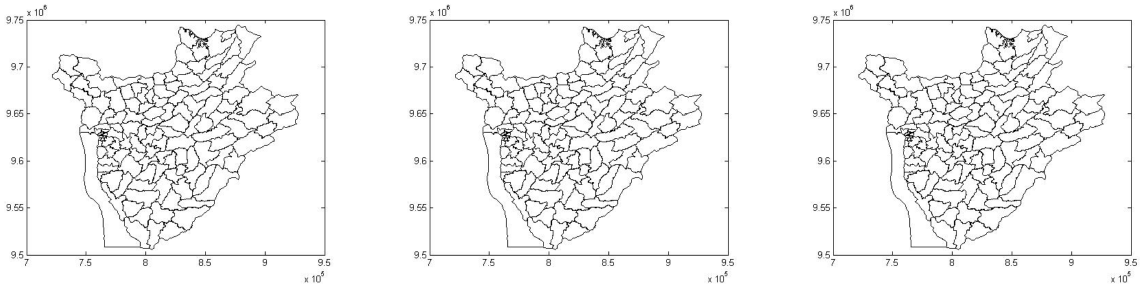
Proportions of map size	The Proposed Approach						Huo et al. (2011b)					
	Map 1		Map 2		Map 3		Map 1		Map 2		Map 3	
	C	F	C	F	C	F	C	F	C	F	C	F
25%	1318	INF	1881	INF	5451	INF	743	INF	1186	INF	2855	INF
33%	1870	INF	2478	INF	9604	INF	613	INF	1377	INF	6353	INF
50%	3315	INF	3806	INF	16417	INF	2288	INF	1912	INF	9123	INF

4.4 Experimental Results and Discussion

To assess the difference introduced by the k-medoids partition clustering method, we carried out a set of experiments regarding fidelity, robustness and capacity, which are described in the following sub-sections, i.e. 4.4.1, 4.4.2, 4.4.3 and 4.4.4 respectively. These experiments are carried on the three maps that are shown in Fig. 4.2.

To enable this comparison, we simulated the scheme of Huo et al. (2011b), as given in their paper and implemented the proposed approach as described previously. This enabled us to compare the two schemes and assess the improvement that could be achieved regarding map data protection.

Table 4.2 shows the experimental results of the implementation in terms of capacity and fidelity, which will be discussed in the following subsections in more detail; the compared results according to computation time, are given in Table 4.3, while robustness is discussed separately. We used different proportions of map size, i.e. 25%, 33% and 50%, to verify the consistency of results.



(a) Original Map

(b) Watermarked Map using Kmeans, 50%

(c) Watermarked Map using Kmedoids, 50%

Figure 4.4: Comparison between the original map (a) with the watermarked maps using k-means (b) and k-medoids (c).

Table 4.3: The compared computational time results of the proposed approach and the scheme of Huo et al. (2011b).

Proportions of map size	The Proposed Approach (seconds)			Huo et al. (2011b) (seconds)		
	Map 1	Map 2	Map 3	Map 1	Map 2	Map 3
25%	0.055854	0.077921	0.134387	0.123270	0.183788	0.300455
33%	0.064616	0.082182	0.143285	0.129038	0.188778	0.303841
50%	0.065580	0.086568	0.143847	0.148653	0.202478	0.319511

4.4.1 The Watermark Capacity Evaluation

Capacity refers to the number of watermark bits that is embedded in the host map. In this Chapter, the watermark capacity is expressed by the number of vertices that carry the watermark bits. Table 4.2 compares the effectiveness of the proposed approach against the approach of Huo et al. (2011b), in relation to the map size proportions of 25%, 33% and 50%, respectively.

These percentages represent the amount of watermarked polygons within the original map, and has a vital implication on adding more resilience to cropping attacks. Cropping refers to the process of cutting some parts in the host map (Zhao et al., 2013a). It is required that each cluster should contain more than one polygon's center, therefore it does not make sense to work with more than 50% of the map data. To illustrate the relation between the map size proportions and the number of clusters, Map 1, Map 2 and Map 3 are used. Thus, for Map 1, 25%, 33% and 50% corresponds to 7, 9 and 13 clusters, respectively, and for Map 2, 25%, 33% and 50% corresponds to 14, 18 and 27, respectively, while for Map 3, 25%, 33% and 50% corresponds to 33, 44 and 66 clusters, respectively.

Table 4.2 shows that the proposed approach results in a higher capacity compared with the approach of Huo et al. (2011b). Moreover, this is done without negatively affecting the fidelity. When using a quarter of the polygons, the capacity achievement of our approach was more than 58% higher than the compared approach, whereas using a third and half of the polygons, the capacity was raised by more than 50%.

As shown in Table 4.2, the proposed approach outperforms the compared approach due to the indexing mechanism of k-medoids partition clustering. This mechanism selects the centers of polygons in relation to surrounding vertices which result in embedding the watermark in more vertices.

4.4.2 The Map Fidelity Evaluation

The fidelity metric aims to measure the perceptual similarity between the watermarked map data and the original map data. It reflects the degree of invisibility the embedded watermark could have. The phrase of degree of invisibility is used to express the fidelity; which refers to the noticeable contrast between the original map and the watermarked map. Huo et al. (2011b) measured this invisibility by using PSNR (Peak Signal to Noise Ratio), in decibels. There is no specific range for PSNR values but a higher PSNR would normally indicate that the data is of higher quality (Huynh-Thu and Ghanbari, 2008). The typical values are considered to be between 30 and 50 dB, in the context of digital images (Hamzaoui and Saupe, 2006).

We used the same metric and Table 4.2 shows that although both the proposed approach and the compared approach give the same fidelity results, the proposed approach (k-medoids-based) outperforms the k-means-based approach in balancing the trade-off between the watermarked map fidelity and the watermark capacity. This is achieved by increasing the capacity without decreasing the fidelity of GIS map data. According to this, the fidelity value of infinity, as shown in Table 4.2, is definitely considered as ideal outcome for the required invisibility.

Fig. 4.4 compares the original map in Fig. 4.4(a) with the watermarked maps using the clustering methods of k-means in Fig. 4.4(b) and k-medoids in Fig. 4.4(c), respectively. The figure illustrates that in both approaches, the watermarked maps are not different from the original one to the human eye.

4.4.3 The Computation Time Evaluation

Computational time refers to the time period, in seconds, for embedding the watermark bits into the host map. Table 4.3 compares the proposed approach versus the scheme of Huo et al. (2011b), in terms of the time required to create the watermarked map. This table shows that our approach uses half the time in comparison to the approach of Huo et al. (2011b), making it more computationally efficient.

4.4.4 The Watermark Robustness Evaluation

Robustness reflects the watermark's resistance to a set of attacks or modifications. This Chapter focuses on geometric attacks such as rotation, translation and scaling because they are more relevant to the geometric nature of polygons in the digital maps context.

Using the mean-distance length values of the selected polygons as watermark carriers has a good implication on the effectiveness of the proposed watermarking approach due to the robustness of mean-distance values to both rotation and translation attacks, and having a way of estimating the scaling factor in the case of scaling attack. These characteristics of using the mean-distance values make the described watermarked approach robust to the geometric attacks (Huo et al., 2011b; Xun et al., 2012).

More specifically, attacks like rotation and translation have no effect on the embedded watermark because they affect equally all vertices' coordinate values, which, in turn, means that the distances between these vertices are not affected. Consequently, since the mean-distance length is used to construct the watermark, such attacks do not affect it.

In the case of a scaling attack, the scaling factor could be computed by dividing the mean-distance values of the modified/attacked map by the mean-distance values of the original map. Consequently, it is easy to retrieve the modified map to its original form before scaling was applied.

4.4.5 The Watermark Position Security Evaluation

In the described watermarking approach, securing the positions of the embedded watermark is achieved by the use of a set of secret keys. The first key is the values of clusters' centers, the second is the values of mean-distance lengths of the selected polygons by using the technique of OES, and the third key is the indexes of the of mean-distance values. These keys are stored for two main purposes: to be used in the extraction process, and for security purposes because they are kept secret from the attackers.

4.5 Summary

In this Chapter we investigated the influence of the partition clustering method used in the watermarking process on the security of the watermarked map. We worked with the scheme proposed by Huo et al. (2011b) by replacing (a) their k-means clustering step with a k-medoids clustering approach, and (b) changing the indexing condition. While in k-medoids partition clustering the centers of clusters are data points from the data sets, in k-means partition clustering, the centers of clusters are artificial points. Consequently, k-means comes with an element of approximation that is not present in the k-medoids approach.

To evaluate the influence the partition clustering method had on the security of the watermarked map, we looked at four aspects: capacity, fidelity, computational time and robustness. The experimental results show that both k-medoids and k-means approaches result in high fidelity, while the k-medoids-based approach achieves a more balanced trade-off between capacity and fidelity, as well as better computational efficiency due to the k-medoids characteristics. In terms of robustness, the results are similar, although an argument could be put forward that this is improved indirectly in the k-medoids approach because of the higher capacity.

For measuring fidelity, PSNR was used to be consistent with the approach of Huo et al. (2011b). This metric is used widely in image watermarking and has also been utilized for vector map data (Huang et al., 2010; Niu et al., 2006; Cox et al., 2007). The map is converted to an image format to meet the applicability of PSNR. In Chapter 7, we propose a different metric that would be more suitable for this type of data.

Chapter 5

The Watermark Robustness Improvement

Research in the area of Geographic Information Systems (GIS) has been growing in recent years, and digital GIS data is now widely available on numerous Internet websites. Consequently, this valuable GIS data is liable to be illegally copied, modified or distributed due to its digital nature. This stands for a compelling need of copyright protection to combat illegal use of GIS data. A popular solution for the protection of GIS data is using digital watermarking systems that enable the identification of unauthorized use of GIS data.

GIS data can be divided into two main models[1]: raster data model and vector data model. The raster model stores the geographic information into a form of grid cells, and each cell represents the natural corresponding value on the ground. On the other hand, the vector data model stores the geographic information into geometrical entities which have properties such as length, a starting point and an ending point (Kennedy, 2013). GIS vector data is defined by a sequence of coordinates, and includes shapes such as points, polylines and polygons (Abbas and Jawad, 2013a). This chapter focuses on the vector format of GIS data.

Data mining in general and clustering in particular, have been recently used for analysing GIS data for a variety of applications such as government and public services; business and service planning; logistics and transportation; and environmental studies (Choi et al., 2014; Croitoru et al., 2013; Miller and Han, 2009). There are, however, only a limited number of approaches using clustering methods in the watermarking field (Abubahia and Cocea, 2014).

In addition, although many watermarking methods have been proposed for digital multimedia data (e.g. images, audio, texts and videos) copyright protection, e.g. (Mohammed et al., 2014; Peng et al., 2014a; Urvoy et al., 2014), digital vector data received less attention,

[1] http://www.ordnancesurvey.co.uk/support/understanding-gis/raster-vector.html

as pointed out in several recent review papers (Abbas and Jawad, 2013a; Bhanuchandar et al., 2013; Wu et al., 2013b).

The previous work in chapter 4 is based on the use of k-medoids clustering for watermarking ESRI (Environmental Systems Research Institute) shapefiles of polygon type (Abubahia and Cocea, 2014); which is discussed in more detail in Section 5.1.

This chapter proposes an improvement to the previous chapter, by using the bounding box property of vector map data, to achieve: (a) robustness to simplification (i.e. deletion of some vertices) (Jianguo et al., 2013b) and interpolation (i.e. adding new vertices) (Wang and Men, 2013) attacks, and (b) preservation of the balance between the map fidelity (the imperceptibility of the inserted watermark) and capacity (distribution of the watermark bits within the data) for GIS vector map copyright protection. These terms are discussed in more detail in Section 5.1.

The rest of this chapter is organized as in the following. In Section 5.1, the GIS map watermarking process is briefly explained and a detailed overview of relevant previous work is presented. Section 5.2 describes the GIS vector data format and the platform used for the experimental evaluation of the proposed approach. Section 5.3 presents in detail the proposed approach, while Section 5.4 discusses the experimental results. Section 5.5 concludes this chapter.

5.1 Research Background

A digital GIS watermarking system consists of three main stages: embedding, attack/modification and extraction (Fig. 5.1). The embedding stage aims to insert a watermark (e.g. digital binary sequence) into the GIS vector map points, by using a specific computing approach; the embedding space is normally the Cartesian coordinates (Jianguo et al., 2013b; Niu et al., 2006). The attack or modification stage is the process of distorting the digital map content. The extraction stage refers to obtaining the watermark from the host GIS data in order to retrieve the original map. There are three key requirements for reliable GIS watermarking system: fidelity, capacity and robustness (Abbas and Jawad, 2013a; Bhanuchandar et al., 2013).

The fidelity requirement refers to the quality of the watermarked GIS data, in the sense that the watermark embedding process should not affect the quality of the host data and that the watermark should not be noticeable to the human eye (Nin and Ricciardi, 2013). The fidelity also indicates the similarity between the original data and the watermarked data. In the case of GIS raster data (image), which offers an extended range (color-scale) for a pixel, this can be solved easily by maintaining the pixel value within a specific range. In

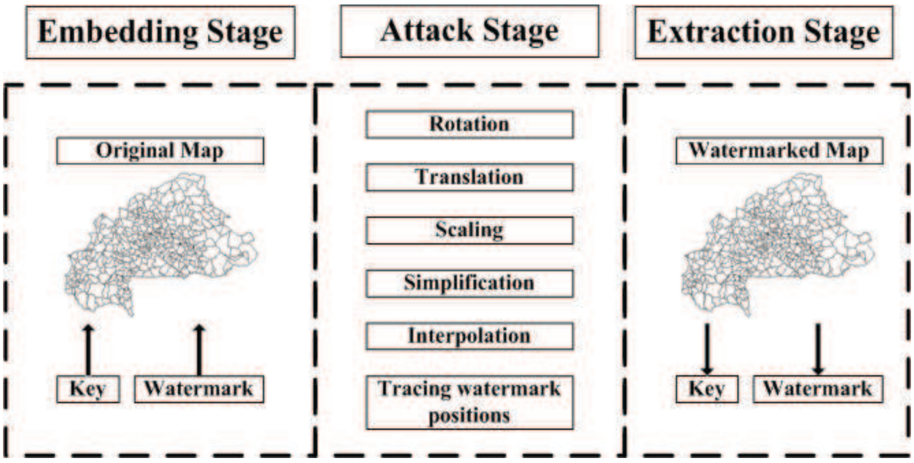
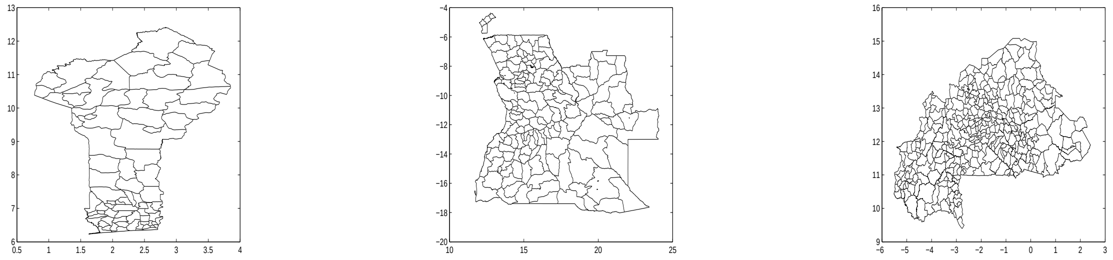


Figure 5.1: Digital GIS Map Watermarking System

contrast, the fidelity requirement stands as a crucial issue in GIS vector data context due to their Cartesian coordinates values sensitivity, which if changed will affect the map shape, and, consequently, will have a negative impact on the usability of GIS map.

The capacity requirement refers to the number of watermark bits that can be embedded in the host map data. The more watermark bits are embedded, the more secure the watermark becomes. Moreover, it is important not only to have high capacity, but also to have the watermark distributed across the entire map (Chapter 4), (Abubahia and Cocea, 2014). This could also leads to a loss of fidelity: the more watermark bits are embedded, the more the host vector map is changed, thus, leading to a loss of map quality. Consequently, the fidelity and capacity requirements need to be balanced to achieve both map quality and watermark quality, in order to ensure the effectiveness of the watermarking method. We refer to this relation between fidelity and capacity as a trade-off, given that an increase in one leads to a decrease in the other, and vice versa.

The robustness requirement refers to the ability of the watermarked data to withstand malicious modifications to the host GIS map, called attacks. There are many types of attacks (Sangita and Venkatachalam, 2012a), of which geometric modifications are particularly important for GIS vector data; such modification processes are rotation, translation and scaling. Rotation means turning the vector map around its center by a specific angle (Lee and Kwon, 2013). Translation means moving the whole map by a specific distance towards a specific direction (Xun et al., 2012). Scaling refers to altering the size of the map, in both axes by a specific value (Lee and Kwon, 2013). Other relevant types of attacks are interpolation (Wang and Men, 2013) and simplification (Jianguo et al., 2013b) attacks. Simplification attacks refer to the process of removing vertices from the map (Jianguo et al., 2013b), while



(a) Benin (222 polygons)

(b) Angola (501 polygons)

(c) Burkina Faso (1046 polygons)

Figure 5.2: The GIS maps used in the experiments.

interpolation attacks refer to the process of adding new vertices in the map (Wang and Men, 2013).

This chapter built on the previous work in chapter 4 to address the vulnerability to simplification and interpolation attacks and to show that the proposed approach is feasible for larger maps. Thus, we argue that using a particular property of vector data called a bounding box in combination with the proposed k-medoids approach in chapter 4, addresses the vulnerability to the two mentioned attacks, while also preserving a good trade-off between fidelity and capacity.

5.2 GIS Vector Data

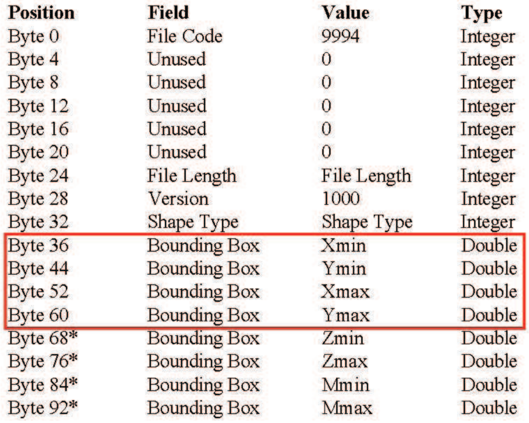
This section describes the GIS vector data that has been used for testing the proposed approach. As shown in Fig. (5.2a), (5.2b) and (5.2c), the used GIS maps are polygon-based maps that represent administrative boundaries of 3 countries in Africa: Benin, Angola and Burkina Faso. These GIS vector maps are freely available, in ESRI shapefile format, from the Natural Earth website.[2]

ESRI Shapefiles (.shp) are produced by ESRI [3], and considered as a popular format for geographic information system applications (Longley et al., 2011). It has several prominent features: small storage space, easy reading and writing, fast shape editing, storing both spatial and attribute information, and supporting point, polyline and polygon geometry types (ESRI, 1998).

Despite the use of ESRI shapefiles in GIS vector data watermarking research (Li et al., 2012b; Huo et al., 2011b), the advantage of the shape bounding box feature in the shapefile

[2] `http://www.mapmakerdata.co.uk.s3-website-eu-west-1.amazonaws.com/library/stacks/Africa/index.htm`

[3] `http://www.esri.com/`



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

Figure 5.3: The Header of Polygon-based Shapefile (ESRI, 1998)

header has not yet been exploited in this context. As shown in Fig. 5.3, the bounding box properties we are interested in are the minimum and maximum coordinates' values in both horizontal and vertical axes.

For the watermark embedding and extraction processes, we implemented the proposed approach in MATLAB version R2013b (8.2.0.701). For more information regarding MATLAB, see the Mathworks website[4].

The following section presents our approach based on k-medoids clustering and using the bounding box information in the ESRI shapefile. This chapter compare the results of this approach with the previous chapter, which used k-medoids clustering with mean polygon centers, to establish the role of the bounding box property in addressing the vulnerability to simplification and interpolation attacks, and to investigate if the trade-off between fidelity and capacity is preserved.

5.3 The Proposed GIS-Map Copyright Protection Approach

This section presents the proposed approach following the three stages outlined earlier in Fig. 5.1: embedding (Section 5.3.1), attack (Section 5.3.2) and extraction (Section 5.3.3).

5.3.1 Embedding Stage

The embedding approach, as illustrated in Fig. 5.4, consists of several steps. First, the locations for inserting the watermark are identified by computing the polygon's centers using the bounding box information for each polygon, and then applying k-medoids to cluster the

[4] http://www.mathworks.co.uk/

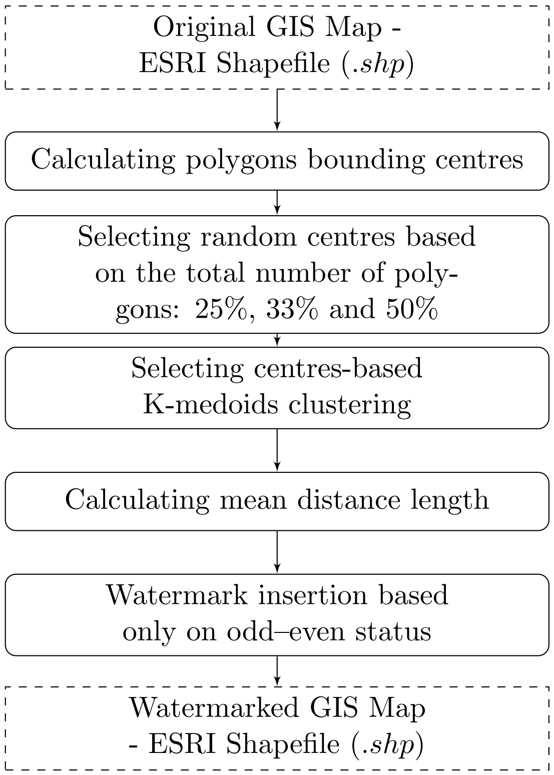

Original GIS Map - ESRI Shapefile (*.shp*)
Calculating polygons bounding centres
Selecting random centres based on the total number of polygons: 25%, 33% and 50%
Selecting centres-based K-medoids clustering
Calculating mean distance length
Watermark insertion based only on odd–even status
Watermarked GIS Map - ESRI Shapefile (*.shp*)

Figure 5.4: The Proposed Embedding-based Cluster Analysis Framework.

computed centers. The number of clusters establishes in how many polygons the watermark will be inserted. We experimented with three different proportions of numbers of polygons in the vector map, i.e. 25%, 33% and 50%. After identifying the locations for watermark insertion, the mean distance length is calculated for the selected polygons and the watermark is inserted into the means distance length by utilizing an odd-even indexing rule.

Embedding Location Identification The approaches given by the previous chapter and Huo et al. (2011b) calculate polygons' centers by summing up all vertices coordinates, in both axes, for each polygon and dividing the sum by the number of vertices minus one; the minus one is due to the the last vertex coordinates being the same as for the first vertex, according to the polygon shapefile format (ESRI, 1998).

In this approach we exploit polygons' bounding boxes property for calculating polygons' centers. Bounding boxes refer to the stored values that represent the extent of the geometry shape in the shape file (ESRI, 1998). Polygons' bounding box centers are calculated in both axes, as shown in Equation(5.1) and Equation(5.2), respectively.

$$x_c = \frac{x_{min} + x_{max}}{2} \tag{5.1}$$

$$y_c = \frac{y_{min} + y_{max}}{2} \tag{5.2}$$

where: x_c and y_c are the coordinates of polygon's center in both x and y axes respectively; x_{min} is the minimum vertex coordinate in x-axis; x_{max} is the maximum vertex coordinate in x-axis; y_{min} is the minimum vertex coordinate in y-axis; y_{max} is the maximum vertex coordinate in y-axis. x_{min}, x_{max}, y_{min} and y_{max} are each of 8-byte length (ESRI, 1998).

Algorithm 2 k-medoids method for GIS vector data clustering

```
Input:
k: the number of clusters,
$D_c^n$: a data set containing number of polygons' centers.
Output: k clusters.
Method:
    • select k polygons' centers in $D_c^n$ as the initial representative polygons' centers; arbitrarily
    • repeat
    • each remaining polygon's center is assigned to the cluster with the nearest representative polygon's center, measured by Euclidean distance;
    • choose, randomly, a non-representative polygon's center, $C_p^{random}$;
    • calculate the total cost, T, of swapped representative polygon's center, $C_p^j$, with $C_p^{random}$;
    • if $T < 0$ then swap $C_p^j$ with $C_p^{random}$ to form the new set of k representative polygons' centers;
    • continue until no change;
```

The key characteristics of the k-medoids partitioning clustering method are robustness to outliers and the fact that the medoids (representative objects) of clusters are represented by actual points in the dataset (Han et al., 2012, 2009), unlike other methods, such as k-means, where the representative objects of clusters are artificial points which are not present in the dateset (Huo et al., 2011b). Therefore, the k-medoids approach can efficiently manage most forms of GIS Vector data.

We use a k-medoids based clustering method called PAM (Partitioning Around Medoids),

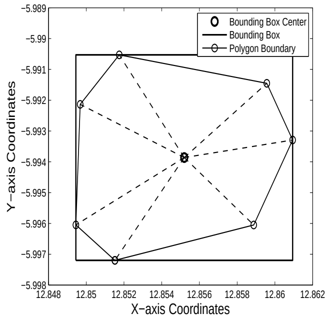


Figure 5.5: Distances from bounding box center to the vertices of polygon

as shown in Algorithm 2, to cluster the bounding box centers in order to determine the best positions for embedding the watermark. The PAM method assigns seeds, i.e initial representative objects, for the given polygons' centers. These seeds are replaced by other representative objects, called medoids, through a number of iterations until the resulting medoids can not be improved or changed. Polygons' centers are clustered into k-clusters and the resulting medoids are kept as a secret key ($key1$). The polygons corresponding to the medoids resulted from clustering are then used for watermark insertion.

Watermark Insertion The concept of zero watermarking (Wang et al., 2012b) is utilized in the proposed watermark embedding process. Zero watermarking aims to exploit some of the host GIS data characteristics in order to generate a more robust watermark. In this case, the topological characteristic of the host GIS data that is used, is the mean-distance length of polygons. This is calculated for the polygons identified through the clustering process.

The watermark is constructed by adding or subtracting a bit value of 1 from the mean-distance length of polygons. The mean-distance length of each polygon is defined by the average value of distance lengths from that polygon's vertices to its center (Huo et al., 2011b; Xun et al., 2012), where the center is calculated as described in Equation(5.1) and Equation(5.2). This is illustrated in Fig. 5.5, while Equation(5.3) demonstrates the way of calculating the mean-distance length of selected polygons.

$$L_c = \frac{1}{n-1} \sum_{v=1}^{n-1} \sqrt{(x_c - x_v)^2 + (y_c - y_v)^2} \tag{5.3}$$

where: L_c is the mean distance length; n is the number of vertices in a polygon; x_c and y_c are the center coordinates in x and y axes, respectively; x_v and y_v are the vertex coordinates in x and y axes, respectively.

The values of mean-distance lengths are stored as a secret key ($key2$) and they represent the selective positions for embedding the watermark. These are based on the polygons whose bounding box centers were selected as final medoids by the k-medoids clustering method. The watermark is embedded by applying odd-even indexing (Huo et al., 2011b; Baiyan et al., 2008a), as outlined in Equation (5.4).

$$W_i = \begin{cases} T-1, & \text{if } OES(I) = odd \\ T+1, & \text{if } OES(I) = even \end{cases} \tag{5.4}$$

where: W_i is the ith bit value of the watermark; OES stands for Odd-Even Status; I is the order index of the mean-distance length value in the matrix; T is the value of the 4th digit of the mean-distance length value, after the decimal point (Huo et al., 2011b).

The index of each mean-distance value is used in this approach, instead of using an additional random sequence proposed by Huo et al. (2011b), to get more consistent positions for embedding the watermark. This consistency sums up both: (a) the indexing as a vital role in the clustering process, and (b) maintaining the security of the watermark position by storing the index values as a key instead of utilizing a random sequence that is not relevant to the used data. This also offers the ability to control the watermark capacity in order to preserve the map fidelity, whereas the use of a random sequence (Huo et al., 2011b) will limit that choice of control.

As shown in Equation (5.4), the watermark is embedded by comparing the OES (Odd-Even Status) of the I and T variables. The conditions are set based on two scenarios as in the following:

- If the OES of I is odd, 1 will be subtracted from the value of T
- In contrast, if the OES of I is even, 1 will be added to the value of T.

After applying the OES to change the values of L_c, the new values of mean-distance length will be represented by L_c^*. The indexes of new mean-distance length values are stored as another secret key ($key3$), to secure the positions in which the watermark is embedded. The change rate α_c is calculated as depicted in Equation (5.5):

$$\alpha_c = \frac{L_c^*}{L_c} \tag{5.5}$$

The change rate α_c is used to change all vertices of polygons identified through clustering on the basis of the embedding condition, as given in equations 5.6 and 5.7:

$$v_x^* = \alpha_c v_x + x_c(1 - \alpha_c) \tag{5.6}$$

$$v_y^* = \alpha_c v_y + y_c(1 - \alpha_c) \tag{5.7}$$

where: v_x^* and v_y^* are the new vertices' coordinates after embedding the watermark according to the aforementioned condition in Equation (5.4).

5.3.2 Attack Stage

Robustness reflects the watermark's resistance to a set of attacks or modifications. This chapter addresses geometric attacks such as rotation, translation and scaling due to their relevance to the geometrical properties of polygons in the GIS vector maps context. Also other relevant attacks such as simplification, interpolation and tracing the positions of watermark bits are taken into account.

1. Rotation Attack: rotation means turning the vector map around its center by a specific angle (Lee and Kwon, 2013). Rotation is of crucial importance because it changes spatial locations of the vector map points. In the proposed approach, this problem is tackled by using the mean distance length which is known for its resilience to the rotation process (Abubahia and Cocea, 2014; Huo et al., 2011b).

2. Translation Attack: translation means moving the whole map by a specific distance towards a specific direction (Xun et al., 2012). Translation also has the property of changing the positions of vector map points, but has no effect on the mean distance length because the distances between the vector map points will remain unchanged (Abubahia and Cocea, 2014; Huo et al., 2011b).

3. Scaling Attack: the scaling attack refers to altering the size of the map, in both axes by a specific value (Lee and Kwon, 2013). Although the scaling attack could change the distances between the vector map points, the scaling factor could be computed by dividing the mean-distance values of the scaled map by the mean-distance values of the original map (Abubahia and Cocea, 2014; Huo et al., 2011b). Consequently, the scaled map can be easily retrieved to its original form after it undergoes the scaling attack.

4. Simplification Attack: the simplification attack refers to the process of removing vertices from the map (Jianguo et al., 2013b). If the polygons' centers are calculated as the average of the vertices, removing some vertices, will change that average. The bounding box centers, however, are not affected by the number of vertices in a polygon; consequently, the proposed approach has more robustness to the simplification attack.

5. Interpolation Attack: the interpolation attack refers to the process of adding new vertices to map's borders (Wang and Men, 2013). Similary to the simplification attack, when the centers of polygons are calculated by averaging the vertices, adding more

vertices will change that average. As the bounding box is independent of the number of vertices in a polygon, out approach will lead to more robustness to interpolation attacks.

6. Tracing watermark bits positions: the positions of the embedded watermark are secured by using a set of three different keys, which are kept secret from the attackers, and stored for the use in the extraction stage. These keys are: (a) the values of computed clusters' centers, (b) the values of mean-distance lengths and (c) the indexes of the new mean-distance values.

5.3.3 Extraction Stage

In the literature, the extraction stage is classified into three categories: blind, semi-blind and non-blind approaches (Abbas and Jawad, 2013a). In the blind approach the original map is not needed in the watermark extraction stage. Semi-blind extraction refers to the case in which the original watermark is used instead of the original map in the watermark extraction stage. Non-blind extraction means that the original map is needed in the watermark extraction stage.

The proposed approach is blind extraction and characterized by flexibility, which means that both the watermark embedding and the watermark extraction processes are quite similar. The keys stored in the embedding process are used in the process of extraction. Firstly, the bounding box center of each polygon is recalculated, and then the polygons' centers are divided into k-clusters by using the k-medoids method, in order to compare with the stored $key1$ (Section 5.3.1). The assumption here is that the attacker will not change the bounding box information, which identifies the boundaries of the whole map, as well as each polygon in the map, because such a change will destroy the map's quality and usability. In the next step, the mean-distance length for the watermarked map is calculated in the same way as in the embedding process. By comparing the computed mean-distance to the stored $key2$ and $key3$ (Section 5.3.1), it becomes easy to extract the watermark bits (1 or -1), and restore the original map even when the watermarked GIS vector map has undergone the attacks mentioned in Section 5.3.2.

5.4 Experimental Results and Discussion

A set of experiments was implemented to assess the balance between fidelity and capacity achieved by the proposed approach. These experiments are carried out on GIS vector maps

Table 5.1: The results of bounding box approach versus mean polygon centers using k-medoids

	k-medoids with bounding box centers		k-medoids with mean polygon centers (in Chapter 4)	
Map (proportion of data used)	Capacity (No. of vertices)	Fidelity (PSNR)	Capacity (No. of vertices)	Fidelity (PSNR)
Benin Map (25%)	1428	42.3485	1321	41.1902
Benin Map (33%)	2187	41.9815	1730	40.8308
Benin Map (50%)	3226	39.2617	2661	38.6129
Angola Map (25%)	4334	46.5627	4118	44.6826
Angola Map (33%)	6379	44.2873	5823	43.3034
Angola Map (50%)	10062	43.6553	9936	41.9183
Burkina Faso Map (25%)	15630	41.1364	15350	40.6581
Burkina Faso Map (33%)	21572	41.6359	19044	40.5387
Burkina Faso Map (50%)	31680	36.8983	31277	36.4201

of 222, 501 and 1046 polygons, as shown in Fig. (5.2a), (5.2b) and (5.2c). The capacity and fidelity results are displayed in Table 5.1.

The fidelity metric aims to measure the imperceptibility of the watermark and reflects its degree of invisibility. This metric is significant because it has two crucial effects in the context of GIS vector data: one on the map shape, and another, consequently, on the usability of the GIS vector map. Fidelity is measured by using PSNR (Peak Signal to Noise Ratio), in decibels (Huo et al., 2011b); there is no specific range for PSNR values but a higher PSNR would normally indicate that the data is of higher quality (Huynh-Thu and Ghanbari, 2008). Typical values are considered to be between 30 and 50 dB, in the context of digital images (Hamzaoui and Saupe, 2006). In order to use this metric, we stored the watermarked GIS vector maps in JPEG image format (jpg) for the measurement purpose.

Capacity refers to the number of vertices that carry the watermark bits. The importance of the watermark capacity is specified by its vital implication on increasing the watermark robustness to cropping attacks. Cropping is the process of cutting some parts of the watermarked GIS vector map (Zhao et al., 2013a). Consequently, it is important not only to have high capacity, but also to have the watermark distributed across the entire map (Abubahia and Cocea, 2014), to avoid having areas of the map with no watermark, which can be then cut off and used without being able to identify ownership. In our approach, the distribution across the map is achieved through the clustering process.

Table 5.1 compares the results of the proposed approach described in this chapter, using the bounding box centers, with the results of the previous work in chapter 4, using polygons' mean centers, to investigate how the performance of the two approaches compare in terms of the trade-off between fidelity and capacity.

There are two considerable differences between the proposed approach and the previous one in chapter 4. The first difference is in the way of calculating the polygons' centers, i.e. using the bounding box as explained in Section 5.3.1 versus using the mean of vertices coordinates in the previous approach in chapter 4. Consequently, the given results can be attributed to the use of the bounding box properties. The second difference is the use of GIS vector maps that contain large numbers of polygons in contrary to chapter 4, which was tested only on small number of polygons (27, 53 and 132 polygons). This should indicate if the approach is suitable for maps with large number of polygons.

As shown in Table 5.1, the trade-off between fidelity and capacity is balanced by increasing the watermark capacity (number of vertices) while keeping higher watermark invisibility (PSNR). Three different proportions of map size, i.e. 25%, 33% and 50%, were used to observe the effect of increased capacity and its effect on fidelity. These proportions represent approximately a quarter, a third and (exactly) half of the number of polygons in the used maps.

The relation between the map size proportions and the number of clusters is illustrated in the following for each of the three maps used in the experiments. Thus, for the map of Benin, 25%, 33% and 50% corresponds to 56, 74 and 111 clusters, respectively; for the map of Angola, 25%, 33% and 50% corresponds to 126, 167 and 251 clusters, respectively; and for the map of South Africa, 25%, 33% and 50% corresponds to 262, 349 and 523 clusters, respectively. This shows that the proposed approach is valid for GIS maps that contain large numbers of polygons.

When looking at the results for the 25% sizes of the three maps in Table 5.1, we notice that the capacity values for the approach proposed in this chapter (bounding box-based k-medoids), i.e. 1428, 4334 and 15630, are higher than those from the previous approach in chapter 4, i.e. 1321, 4118 and 15350. At the same time, it is noticeable that the fidelity values are also higher than the approach of in chapter 4, despite the increase in capacity. The same can be observed for the 33% and 50% sizes on all three maps.

As pointed out in the previous section, one key characteristic of using the bounding box centers is that it does not depend on the number of vertices in a polygon, which has an advantages of more robustness to the interpolation and simplification attacks. Therefore, the approach proposed in this chapter improves the previous approach in chapter 4 by achieving robustness to simplification and interpolation attacks, while also increasing the fidelity and capacity metrics, and, at the same time, preserving the balance between the two metrics.

5.5 Summary

The influence of using the bounding box properties for protecting the copyright of GIS vector data was investigated in this chapter. We introduced the use of bounding box centers in the context of watermarking research, and compared the approach in this chapter with previous work in chapter 4.

To assess the effectiveness of the proposed approach, we looked at two important aspects: fidelity and capacity. The experimental results show that the use of the bounding box centers has a significant implication on the trade-off between the fidelity and the capacity metrics, and resulted in higher fidelity as capacity increased.

In addition to the improvement of the trade-off between fidelity and capacity, the use of bounding box centers adds more robustness to the simplification and interpolation attacks due to their independence from the number of vertices in a polygon. By using vector maps with large numbers of polygons, the approach has been shown to be feasible for large maps.

For measuring fidelity, PSNR was used to be consistent with the previous work in this area, including the previous work in chapter 4, which is an improved work of the approach by Huo et al. (2011b). This metric, however, is used in image watermarking and is not necessarily the best metric for GIS vector data (Niu et al., 2006), as it does not exploit the properties of vector data. As there is no current alternative for measuring fidelity, in future work, we will investigate different metrics that would be more suitable for vector map data.

Chapter 6

Clustering Approaches Comparison

Advancements in Geographical Information Systems (GIS) technology, including capabilities of mapping, monitoring, modeling, management and measurement (Burrough et al., 2015; Longley et al., 2011), led to an increased employment of GIS maps in many applications such as government and public services (Kingston, 2007), business and service planning (Angelaccio et al., 2012), logistics and transportation (Camelli et al., 2012), and environmental studies (Busch, 2012).

The production of a GIS map involves a time-consuming process of analysis and the use of well-trained specialists, accurate hardware and licensed software tools. Therefore, given the high cost of producing GIS maps, the producers of these maps are interested in preserving their copyright.

Moreover, these maps are liable to be illegally copied, modified or distributed due to their digital nature. Consequently, there is a compelling need for copyright protection to combat illegal use of GIS maps (Ciptasari and Sakurai, 2013; Wang et al., 2007).

GIS data can be represented in the form of two main models (Bonham-Carter, 2014): raster and vector data. The raster model (image) stores the geographic information into a form of grid cells, where each cell represents the natural corresponding value on the ground (e.g. color scale). The vector data model stores the geographic information into geometrical entities which have properties such as length, a starting point and an ending point (Kennedy, 2013). GIS vector data is defined by a sequence of coordinates, and includes shapes such as points, polylines and polygons (Abbas and Jawad, 2013a). In this Chapter the focus is on the vector format of GIS data.

In response to the copyright protection issue, many digital watermarking methods have been proposed in literature, e.g. (Abubahia and Cocea, 2015a; Peng et al., 2014a; Urvoy et al., 2014). Nevertheless, GIS data received less attention than images, audio, texts and videos in the field of watermarking research, as pointed out in several recent review papers (Abbas and Jawad, 2013a; Bhanuchandar et al., 2013; Wu et al., 2013b).

In addition, although the use of partition clustering methods for GIS map applications has seen an increase in recent years (e.g. (Choi et al., 2014; Croitoru et al., 2013; Han et al., 2009)), little research in GIS map watermarking takes advantage of data mining methods in general, and clustering in particular. Two partitioning clustering methods have been used in this area: k-means (Huo et al., 2011b) and k-medoids (Abubahia and Cocea, 2014, 2015a). In the context of GIS map watermarking, partition clustering methods use the distance between map vertices to divide the map vertices into a set of clusters with the purpose of identifying locations for embedding the watermark, and provides the advantage of ensuring the distribution of the watermark across the entire map.

In previous work (Abubahia and Cocea, 2015a) we showed that the use of a particular property of vector maps, called a bounding box, leads to an increased resistance of the watermarked map to malicious modification called attacks, and in particular, to interpolation (i.e. adding vertices to the map) (Huo et al., 2011b) and simplification (i.e. deleting vertices from the map) (Jianguo et al., 2013b) attacks. Moreover, we argued that the proposed approach maintains a good trade-off between capacity (the number of inserted watermark bits) and fidelity (the quality of the map after watermark insertion). This trade-off is very important for vector map data, as its value stems from its accurate locations properties; therefore, it is important that the watermark does not affect the precision of the locations (i.e. it has good fidelity), while at the same time it provides enough watermark bits to ensure the map's copyright protection (i.e. it has good capacity).

In this Chapter we argue that using the bounding box property of vector maps maintains the resistance to interpolation and simplification attacks even when a different clustering approach is used for identifying locations for embedding the watermark. To assess this claim, we compare the original k-means approach of Huo et al. (Huo et al., 2011b) with a modified k-means approach using the bounding box property.

The two approaches are also compared in terms of the trade-off between fidelity and capacity, to assess the influence of using the bounding box property on this trade-off. Moreover, we argue that the advantages of using the bounding box property would hold regardless of the methods used for identifying the watermark embedding locations in the map.

The rest of this Chapter is organized as described in the following. In Section 6.1, GIS watermarking research terminology and requirements are briefly explained, while Section 6.2 gives a detailed overview of relevant previous work. Section 6.3 presents the k-means approach with the use of the bounding box property and Section 6.4 presents the comparison results of the two clustering approaches, i.e. the modified k-means approach using the bounding box

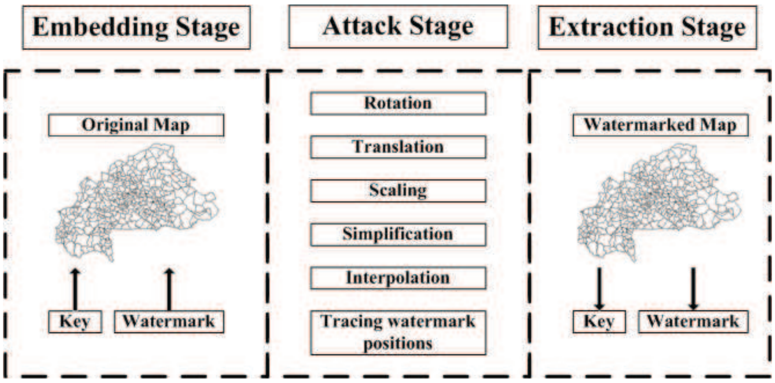


Figure 6.1: Digital GIS Map Watermarking System

property and the k-means approach of Huo et al. (Huo et al., 2011b). Section 6.5 concludes this Chapter.

6.1 Research Background

A GIS map watermarking system includes two main stages: embedding and extraction (Fig. 6.1). The embedding stage aims to insert a watermark (e.g. digital binary sequence) into the GIS vector map points, by using a specific computing approach; the embedding space is often the Cartesian coordinates (Jianguo et al., 2013b), (Niu et al., 2006). The extraction stage refers to obtaining the watermark from the host GIS data in order to retrieve the original map. There are three key requirements for a reliable GIS watermarking system: fidelity, capacity and robustness (Abbas and Jawad, 2013a), (Bhanuchandar et al., 2013).

The fidelity requirement refers to the similarity degree between the watermarked map and the original map in the sense that the watermark insertion process should not noticeably affect the shape and quality of the host map Nin and Ricciardi (2013).

The capacity requirement refers to the amount of inserted watermark bits into the host GIS map. In addition, the watermark bits should be well-distributed over the whole digital map for securing the watermark. The more watermark bits are inserted, the more the host map is changed, which may lead to a decrease in fidelity (Abubahia and Cocea, 2014). Therefore, fidelity and capacity need to be balanced to achieve good security with minimal loss of quality.

The robustness requirement refers to the resilience of the watermark against a potential set of modifications, referred to as attacks, to the host GIS map. Resistance to these attacks is important because they could seriously change the map shape in terms of vertices' coordinates values, and, as a consequence, making the process of watermark extraction more difficult; this, in turn, would jeopardise the identification of the rightful owner of the data. There are several attacks that are relevant for vector map data:

1. rotation attacks, which mean using a specific angle to turn the GIS map around its center (Lee and Kwon, 2013);
2. translation attacks, which involve moving the whole map by a specific distance towards a specific direction (Xun et al., 2012);
3. scaling attacks, which refer to the use of a specific value, in both axes, to alter the size of the GIS map (Lee and Kwon, 2013);
4. simplification attacks, which involve the removal of some vertices from the GIS vector map (Jianguo et al., 2013b);
5. interpolation attacks, which consist of adding new vertices into the GIS vector map (Wang and Men, 2013).

Despite the use of ESRI shapefiles in GIS vector data watermarking research, e.g. (Abubahia and Cocea, 2014; Huo et al., 2011b; Wang et al., 2014), the advantage of the shape bounding box feature in the shapefile header has not been exploited in the watermarking context apart from the previous approach in Chapter 5. As shown in Fig. 6.2, the bounding box properties we are interested in are the minimum and maximum coordinates' values in both horizontal and vertical axes.

6.2 Related Work

There are few published approaches that used partition clustering methods for protecting the copyright of GIS vector maps. In this section, these approaches are reviewed in relation to the trade-off between fidelity and capacity, and in relation to their vulnerability to interpolation and simplification attacks.

Huo et al. (Huo et al., 2011b) presented an approach that used k-means partition clustering for inserting a watermark into a GIS vector map composed of a small number of polygons, based on ESRI shapefile format, according to the polygons' mean centers (i.e. the mean of vertices' coordinates values in the polygon). Although their fidelity achievement is considerably high, the capacity of the watermark was relatively low for the size of the GIS map they used. Therefore, this approach, does not achieve a good trade-off between fidelity and capacity; in addition, this approach is vulnerable to simplification and interpolation attacks.

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

Figure 6.2: The Header of Polygon-based Shapefile, (ESRI, 1998)

To improve the approach of Huo et al. (Huo et al., 2011b), we presented an approach in Chapter 4 that used k-medoids-based partition clustering for inserting watermark bits into a set of GIS vector maps composed of a small number of polygons, and we used mean polygons' centers for identifying the optimum position to insert watermark bits into the GIS maps. Although this approach improved the trade-off between fidelity and capacity, it did not address the vulnerability to simplification and interpolation attacks. Moreover, both approaches did not consider the case of larger maps.

To improve the robustness of the previous approach in Chapter 4, we extended it in Chapter 5 by using k-medoids-based clustering and polygon bounding box information in ESRI shapefiles, for inserting watermark bits into a set of GIS vector maps composed of larger numbers of polygons. In this approach, the polygons bounding boxes centers are used to identify the optimum locations in the GIS map for inserting the watermark bits. This approach achieved: (1) robustness to both simplification and interpolation attacks, (2) a considerable increase in the trade-off between fidelity and capacity and (3) reliability of the approach for GIS vector maps composed of larger number of polygons.

Regardless of the partition method used, It is argued in this Chapter that the use of bounding box property of GIS vector map for locating the watermark bits into polygons' vertices has a significant implication on protecting the GIS vector map copyright, especially in terms of addressing the vulnerability to simplification and interpolation attacks, while preserving a good trade-off between fidelity and capacity. To asses this, we compare a k-means clustering approach based on the bounding box centers of polygons with the k-means approach of Huo et al. (Huo et al., 2011b).

6.3 K-means Clustering with Bounding Boxes Approach

This section presents the proposed approach based on k-means partition clustering using the bounding box information in the ESRI shapefile. We compare the results of this approach with the work of Huo et al. (Huo et al., 2011b), which used the mean centers of polygons. The purpose is to establish the role of the bounding box property in addressing the vulnerability to simplification and interpolation attacks, and to investigate if the trade-off between fidelity and capacity is preserved.

6.3.1 Embedding Locations Identification

The embedding stage involves the identification of locations for embedding the watermark and the insertion of the watermark bits in the identified locations.

The location identification involves three consecutive steps: computing the bounding box centers for each polygon, applying k-means clustering to the polygons' computed centers, and calculating mean distance values (the locations for inserting the watermark bits). Each of these steps is described in the following.

Step 1 : Computing Bounding Box Centers

Each polygon in the GIS vector map has a defined bounding box, which identifies the boundaries of each polygon in the map; the coordinates for the bounding box are available in the shapefile (ESRI, 1998), as illustrated in Fig. 6.2. Polygons' bounding box centers are calculated in both axes, as shown in Equation (6.1) and Equation (6.2), respectively.

$$x_c = \frac{x_{min} + x_{max}}{2} \tag{6.1}$$

$$y_c = \frac{y_{min} + y_{max}}{2} \tag{6.2}$$

where: x_c and y_c are the coordinates of polygon's center in both x and y axes respectively; x_{min} is the minimum vertex coordinate in x-axis; x_{max} is the maximum vertex coordinate in x-axis; y_{min} is the minimum vertex coordinate in y-axis; y_{max} is the maximum vertex coordinate in y-axis. x_{min}, x_{max}, y_{min} and y_{max} are each of 8-byte length (ESRI, 1998).

Unlike the approach to calculating the polygons' centers, based on bounding boxes as explained above, Huo et al. (Huo et al., 2011b) calculate polygons' centers by summing up all vertices coordinates for each polygon and dividing the sum by the number of vertices minus one; the minus one is due to the last vertex coordinates being the same as for the first vertex, according to the polygon shapefile format (ESRI, 1998). These polygons' average centers are quite sensitive to the total number of vertices in a polygon; consequently, adding (interpolation) or removing (simplification) some vertices, will change the average value of the polygons' centers. In contrast, the bounding box centers are independent from the total number of vertices in a polygon; consequently, the use of this property plays a significant role in achieving the required robustness to both simplification and interpolation attacks.

Step 2 : Clustering Polygons' Bounding Boxes Centers

The k-means method is used to cluster the bounding box centers in order to determine the positions for embedding the watermark. The k-means clustering method is relatively simple, easy to implement, and needs a predefined number of clusters (k). We experiment with different numbers of k to explore different values for capacity and the effect they have on fidelity. More specifically, we experiment with values of k that represent approximately 25%, 33% and 50% of the total number of polygons. The resulting centroids are kept as a secret key ($key1$).

Step 3 : Distance Calculation

For each cluster centroid identified at the previous step, unlike the previous approaches (Abubahia and Cocea, 2015a, 2014; Huo et al., 2011b), the distance length is calculated by measuring the distance from the polygon bounding box top right corner to its center, where the center is calculated as described in Step 1. Equation (6.3) illustrates the way of computing the distance length of selected polygons. This approach is adding increased robustness to the simplification and interpolation attacks due to the independence of this distance of the number of vertices in a polygon.

$$L_c = \sqrt{(x_c - x_{max})^2 + (y_c - y_{max})^2} \tag{6.3}$$

where: L_c is the distance length; x_c and y_c are the center coordinates in x and y axes, respectively; x_{max} and y_{max} are the up right bounding box corner coordinates in x and y axes, respectively.

The values of bounding box distance lengths for all selected polygons are stored as a secret key ($key2$) and they represent the selective positions for embedding the watermark.

6.3.2 Watermark Bits Insertion

The concept of zero watermarking (Wang et al., 2012b) is utilized in the proposed watermark embedding process. Zero watermarking aims to exploit some of the host GIS data characteristics in order to generate a more robust watermark. In this case, the topological characteristic of the host GIS data that is used, is the distance length of polygons.

The watermark is constructed by adding or subtracting a bit value of 1 from the distance length of polygons. The watermark is embedded by applying odd-even indexing (Baiyan et al., 2008a; Huo et al., 2011b), as outlined in Equation (6.4).

$$W_i = \begin{cases} T-1, & \text{if} \quad OES(I) = odd \\ T+1, & \text{if} \quad OES(I) = even \end{cases} \tag{6.4}$$

where: W_i is the ith bit value of the watermark; OES stands for Odd-Even Status; I is the order index of the distance length value in the matrix; T is the value of the 4th digit of the distance length value, after the decimal point (Huo et al., 2011b).

The index of each distance value is used in this approach, instead of using an additional random sequence proposed by Huo et al. (2011b), to get more consistent positions for embedding the watermark. This consistency sum up both: (a) the indexing as a vital role in the clustering process, and (b) maintaining the security of the watermark position by storing the index values as a key instead of utilizing a random sequence that is not relevant to the used data. This also offers the ability to control the watermark capacity in order to preserve the map fidelity, whereas the use of a random sequence (Huo et al., 2011b) will limit that choice of control.

As shown in Equation (6.4), the watermark is embedded by comparing the OES (Odd-Even Status) of the I and T variables. The conditions are set based on two scenarios as in the following:

- If the OES of I is odd, 1 will be subtracted from the value of T.
- In contrast, if the OES of I is even, 1 will be added to the value of T.

After applying the OES to change the values of L_c, the new values of distance length will be represented by L_c^*. This new distance length values are stored as another secret key ($key3$), to secure the positions in which the watermark is embedded. The change rate α_c is calculated as depicted in Equation (6.5):

$$\alpha_c = \frac{L_c^*}{L_c} \tag{6.5}$$

The change rate α_c is used to change all vertices of polygons that belong to each cluster's center on the basis of the embedding condition, as given in Equations (6.6) and (6.7).

$$v_x^* = \alpha_c v_x + x_c(1 - \alpha_c) \tag{6.6}$$

$$v_y^* = \alpha_c v_y + y_c(1 - \alpha_c) \tag{6.7}$$

where: v_x^* and v_y^* are the new vertices' coordinates after embedding the watermark according to the aforementioned condition, in Equation (6.4); v_x and v_y are the original vertices' coordinates before inserting the watermark bits.

Embedding the watermark bits into the distance length values has the advantage of providing robustness to rotation, translation and scaling attacks.

In rotation and translation attacks the entire map is shifted either by turning the map around to a specific angle or by moving the entire map in a specific direction. These modification apply the same shift to all coordinate values of vertices; this consistent change signifies that the distance values will remain the same. Consequently, the distance lengths are not affected by rotation and translation attacks.

Scaling attacks involve a change in the size of the map by a particular scaling factor. This scaling factor can be determined by dividing the distance values of the attacked map by the distance values of the original map (i.e. *key*2). Consequently, the original map can be restored from the attacked map by applying the complementary scaling factor to the attacked map.

6.3.3 Watermark Bits Extraction

The proposed approach is characterized by blindness and flexibility. Blindness means that the original vector map is not needed in the watermark extraction process, while flexibility means that the watermark extraction process can be implemented in similar way as presented in the watermark embedding process.

The bounding box centers are computed for each polygon and the k-means method is used to divide the polygons' computed centers into k-clusters in the same way as illustrated in Step 2 (Section 6.3.1). The results are compared with the stored *key*1 to identify if there have been some modification applied to the vector map; the comparison with the other stored keys (see below) has the same purpose.

The distance length for the watermarked map are calculated in the same way as illustrated in Step 3 (Section 6.3.1). The recalculated distances are compared to the stored *key*2 and *key*3, to ensure that the vector map has the embedded watermark bits (1 or -1) in order to go further for the extraction stage. Both *key*2 and *key*3 help in retrieving the watermarked map to its original form, which maintain the robustness to rotation, translation and scaling attacks, as discussed in Section 6.3.2.

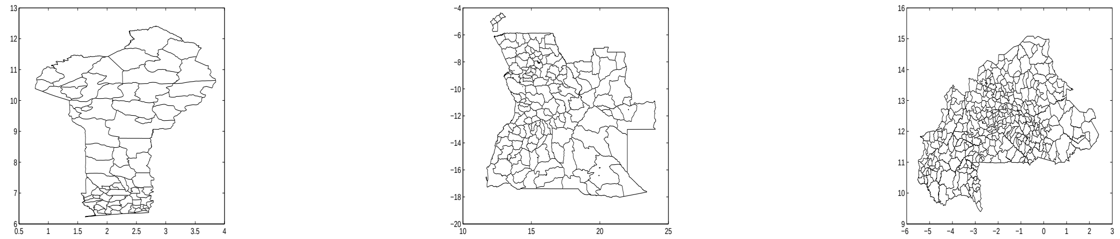
6.4 Experiments and Discussion

Three maps were used to compare the k-means approach using the bounding box property with the k-means approach using the mean centers of polygons. As shown in Fig. (6.3a), (6.3b) and (6.3c), the used GIS maps are polygon-based maps that represent administrative boundaries of 3 countries in Africa: Benin (222 polygons), Angola (501 polygons) and Burkina Faso (1046 polygons). These GIS vector maps are freely available, in ESRI shapefile format, from the Natural Earth website.[1]

ESRI Shapefiles (.shp) are produced by ESRI [2], and considered as a popular format for geographic information system applications (Longley et al., 2011). They have several key features: small storage space, easy reading and writing, fast shape editing, storing both spatial

[1] http://www.mapmakerdata.co.uk.s3-website-eu-west-1.amazonaws.com/library/stacks/Africa/index.htm

[2] http://www.esri.com/



(a) Benin (222 polygons)

(b) Angola (501 polygons)

(c) Burkina Faso (1046 polygons)

Figure 6.3: The GIS vector maps used in the experiments.

Table 6.1: The percentage of added/removed vertices in relation to the total number of vertices in the Burkina Faso map

The total number of vertices in the map	113996
The number of watermarked vertices in the map	39375
The number of removed/added vertices	7875

and attribute information, and supporting point, polyline and polygon geometry types (ESRI, 1998).

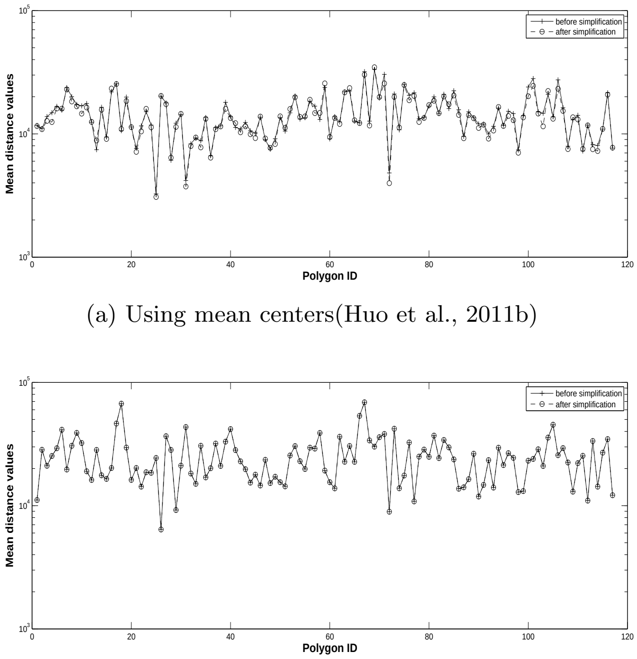
For the watermark embedding and extraction processes, we implemented the two approaches in MATLAB[3] version R2014b (8.4.0.150421) on 64-bits windows-PC.

The effect of simplification and interpolation attacks on the two approaches has been investigated through the following experiment on the map of Burkina Faso, i.e. the map with the largest number of polygons (1046). As shown in Table 6.1, the map of Burkina Faso contains a total of 113996 vertices. The watermark was inserted in a third of the whole map, i.e. 349 polygons were watermarked containing 39375 vertices, referred to as watermarked vertices in Table 6.1. The total number of removed or added vertices is 7875, which represents 6.9% of the map vertices and 20% of the watermarked vertices.

For each watermarked polygon, 20% of the vertices were removed (for the simplification attack) or added (for the interpolation attack), and the changes in the computed distance values were calculated. The differences are illustrated in Fig. (6.4) and Fig. (6.5) and they point out that the interpolation and simplification attacks result in changes when the mean centers approach is used, but have no effect on the bounding box approach. Consequently, the bounding box approach is robust to simplification and interpolation attacks.

Although the changes in the mean distance values when using the mean centers may

[3] http://www.mathworks.co.uk/



(a) Using mean centers(Huo et al., 2011b)

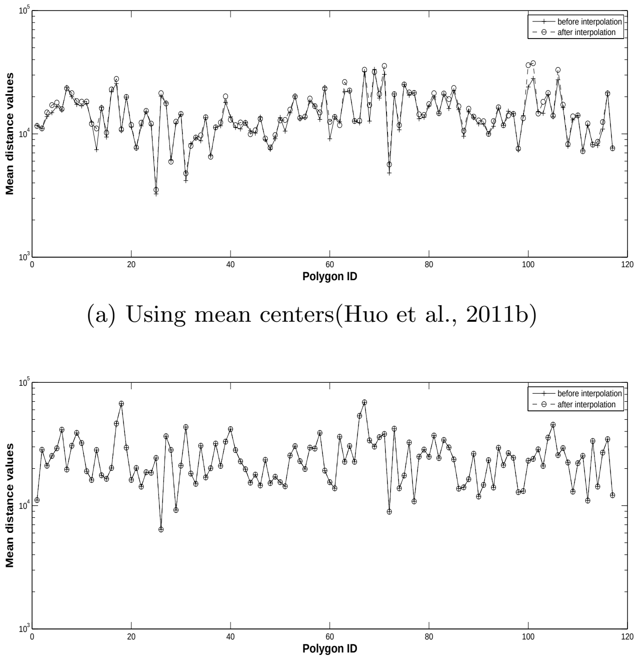
(b) Using Bounding Box centers

Figure 6.4: Changes after the simplification attack.

seem small, they are significant because they distort the watermark, which may lead to the loss of its copyright. Moreover, these small changes may also mean that the quality of the map is still quite high, thus allowing the attackers to use it without the liability of copyright infringement.

The robustness to the simplification and interpolation attacks is ensured by using the bounding box centers due to their independence of the number of vertices in a polygon. In other words, removing or adding new vertices would not affect the main four corners of the bounding box, thus leaving the value of the polygon center unchanged. In contrast, the mean centers approach (Huo et al., 2011b) is vulnerable to these attacks because the center of a polygon is calculated as the average values of vertices' coordinates, thus depending on the number of vertices. Consequently, the removal or addition of vertices will affect the values of the polygons' centers calculated with this approach.

Here, we assume that the attacker will not remove each of bounding box corners' coordinates because removing each of these coordinates will lead to a considerable change in polygon shape and make the map unusable due to the loss of its quality. An attacker would normally be interested in preserving the quality of the map when removing its watermark, so that they can still use it.



(a) Using mean centers(Huo et al., 2011b)

(b) Using Bounding Box centers

Figure 6.5: Changes after the interpolation attack.

To further test the bounding box approach, we investigated the effect of this approach on the capacity and fidelity metrics. Table 6.2 compares the results of the proposed approach using the bounding box information with the results for the approach using mean centers (Huo et al., 2011b), to investigate the advantage of using polygons' bounding box based centers over the traditional polygons' mean centers in achieving a good trade-off between fidelity and capacity. The difference between the compared approaches is in the definition of the centers of polygons, i.e. using the bounding box information as explained in Section 4.1 vs using the mean of polygon vertices coordinates in the approach of Huo et al. (2011b). Consequently, the difference in results can be attributed to a certain degree to the use of the bounding box properties.

The fidelity metric aims to measure the imperceptibility of the watermark and reflects its degree of invisibility. This invisibility is measured by using PSNR (Peak Signal to Noise Ratio), in decibels (Abubahia and Cocea, 2014; Huo et al., 2011b). There is no specific range for PSNR values but a higher PSNR would normally indicate that the data is of higher quality (Huynh-Thu and Ghanbari, 2008). The typical values are considered to be between 30 and 50 dB, in the context of digital images (Hamzaoui and Saupe, 2006). In order to use this metric, we stored the watermarked GIS maps in JPEG image format.

Table 6.2: of C (Capacity) and F (Fidelity) between bounding box approach and mean polygon centers using k-means

	The Proposed Approach		Huo et al. (2011b)	
No. of Clusters (= No. of Polygons)	C	F	C	F
Benin Map (25%)	1278	40.5223	1113	39.7769
Benin Map (33%)	1452	40.1054	1239	40.0029
Benin Map (50%)	2492	37.4146	2363	36.9682
Angola Map (25%)	4009	43.6947	3902	43.6013
Angola Map (33%)	5365	41.9369	5219	41.9134
Angola Map (50%)	9631	39.7193	9507	39.7081
Burkina Faso Map (25%)	15242	40.3900	15171	40.1572
Burkina Faso Map (33%)	19017	39.5909	18930	38.9722
Burkina Faso Map (50%)	31147	36.2769	31012	36.2584

On the other hand, capacity refers to the number of vertices in the host GIS map, which carry the watermark bits. The importance of the watermark capacity is specified by its vital implication on increasing the watermark robustness to cropping attacks. Cropping is the process of cutting some parts of the host GIS map (Zhao et al., 2013a). Consequently, it is important not only to have high capacity, but also to have the watermark distributed across the entire map.

As shown in Table 6.2, the trade-off between capacity and fidelity is achieved by increasing the number of vertices that carry the watermark bits (capacity) while keeping higher watermark invisibility measured by PSNR (fidelity).

In addition, three different proportions of map size, i.e. 25%, 33% and 50%, were used to observe the effect of increased capacity and its effect on fidelity. These proportions represent approximately a quarter, a third and half of the number of polygons in the used maps. The relation between the map size proportions and the number of clusters is illustrated in the following for each of the three maps used in the experiments. Thus, for the map of Benin, 25%, 33% and 50% corresponds to 56, 74 and 111 clusters, respectively; for the map of Angola, 25%, 33% and 50% corresponds to 126, 167 and 251 clusters, respectively; and for the map of Burkina Faso, 25%, 33% and 50% corresponds to 262, 349 and 523 clusters, respectively.

When looking at the results for the 25% sizes of the three maps in Table 6.2, we notice that the capacity values for the approach proposed in this Chapter (bounding box-based k-means), i.e. 1278, 4009 and 15242, are higher than those of Huo et al. (2011b), i.e. 1113, 3902 and 15171. At the same time, we notice that the fidelity values are also higher in the approach using the bounding box compared with the approach using the mean centers, despite the increase in capacity. The same can be observed for the 33% and 50% sizes on all three maps.

The research in this Chapter and in the previous Chapter used clustering for identifying the embedding locations. Clustering has the advantage of ensuring a good distribution of the watermark across the entire map, thus adding resilience to cropping attacks. Other approaches for the identification of locations, however, when used in conjunction with the bounding box property, should still preserve the robustness to simplification and interpolation attacks.

In terms of the trade-off between capacity and fidelity, the experimental results reported in this Chapter, as well as previous results using k-medoids clustering (Abubahia and Cocea, 2015a) indicate that the use of the bounding box centers led to a good trade-off between capacity and fidelity; more specifically, the results indicate an increase in fidelity even when there is an increase in capacity. Further experimentation would be needed to assess the role of the bounding box in achieving this results compared with the role of the two other main factors: the approach for identifying the embedding locations and the watermark insertion approach. Although the results indicate that the bounding box plays a role in the trade-off between capacity and fidelity, we cannot separate this effect from the two factors mentioned above.

6.5 Summary

In this Chapter, we investigated the influence of using the bounding box property for protecting the copyright of digital GIS vector maps, by comparing a k-means clustering method that used the bounding box property with the earlier work of Huo et al. (2011b) using the mean centers of polygons.

Using bounding box centers increases the robustness to the simplification and interpolation attacks due to the independence property of bounding box centers from the number of vertices in a polygon, in contrast to mean centers of polygons, which are dependent on the number of vertices.

The effectiveness of the proposed approach is assessed by looking at both fidelity and capacity aspects. The experiments demonstrate that the computation of bounding box centers has a considerable implication on the trade-off between the fidelity and the capacity metrics, and resulted in higher fidelity as capacity increased.

The PSNR fidelity metric was used for consistency with the work of Huo et al. (2011b). This measurement is often used in image watermarking research, and is not necessarily the best metric for GIS vector map (Niu et al., 2006).

Building on the previous approach based on k-medoids clustering in Chapter 5 which demonstrate the advantages of using bounding box property for promoting the research of GIS map copyright protection, the proposed approach in this Chapter stresses that these advantages are maintained even with a k-means clustering based approach, and can reasonably conclude that they would hold regardless of the clustering method used for identifying the watermark embedding locations in the map.

Chapter 7

Topological Quality Measurement

A key requirement of any watermarking approach is the quality preservation in the watermarked data (Abubahia and Cocea, 2017; Wang et al., 2015b). In the context of vector data, the quality preservation expresses that the original vector map is not affected by the concealed watermark, and is referred to as *fidelity*. Most often this is defined as the perceptual degree of similarity between the original vector map and the watermarked vector map. In the context of images (although used with vector map data as well) it is referred to as *invisibility*. In both cases, the emphasis is on the perceptual perspective (Urvoy et al., 2014) and is measured with error metrics, such as RMSE (Root Mean Squared Error) and PSNR (Peak Signal to Noise Ratio) (which is based on mean squared error). More details about the metrics used for invisibility of vector data can be found in (Abubahia and Cocea, 2017; Peng et al., 2014b; Huynh-Thu and Ghanbari, 2008).

While in the context of image watermarking the invisibility of the watermark can be taken to mean that the original image has preserved its quality (Wang et al., 2015d), in the context of vector data, the quality of the map needs to be assessed in terms of the preservation of its topological properties, i.e. the geometrical shapes have not been distorted in the watermarking process. Although the need for a metric to assess topological quality preservation has been repeatedly highlighted (Abubahia and Cocea, 2017; Niu et al., 2006; Huang et al., 2010; Abubahia and Cocea, 2014), few research works looked into this aspect (Huang et al., 2010; Kim, 2010a,b; Sangita and Venkatachalam, 2012a; Neyman et al., 2014a). These works discussed the importance of topology preservation, and for particular applications looked at the effect of watermarking on some topological properties. A metric for quantifying topological distortion that can be used for assessing watermarked vector map topological quality has not yet been presented.

In this Chapter, a metric based on topological properties of polygon-based maps is proposed. Here, the focus is on three topological rules, stating that the polygons need to be closed, that they should not have gaps between them and that they should not overlap. Consequently, a metric that quantifies to what degree these rules are broken is presented in this Chapter, i.e. how many polygon disclosures, gaps and overlaps are present, in proportion to watermark size. To evaluate the metric, experiments with the two different embedding approaches mentioned above and controlled watermarking capacity (i.e. how much is embedded) were run on maps of various sizes.

The rest of this Chapter is organized as follows: Section 7.1 reviews previous work on topology preservation in the context of digital vector map watermarking. Section 7.2 introduces the proposed metrics for measuring the polygon disclosure, overlap and gap aspects. Section 7.3 describes the experiments, including the data used and the experimental setup for the evaluation of the proposed metric. Section 7.4 discusses the experimental results, while Section 7.5 summarises this Chapter.

7.1 Related Work

In this section, the topological aspects of vector data and the importance of their preservation are briefly outlined. Also an overview of previous work is introduced in relevance to addressing the issue of topological preservation when assessing watermarked vector map quality.

Unlike raster image data, vector map data has to follow topological rules that specify constraints for the shapes, e.g. lines and polygons, used in vector maps. The development of vector maps GIS tools (e.g. ArcGIS) (Maguire, 2015) allows the identification of these errors, which allows them to be fixed. The value of the vector maps is related to the precision of the data, which allows spatial analysis (Maraş et al., 2010). While it is accepted that watermarking without any effect on the precision of vector map data is not possible (Kim, 2010a), it is also clear that measuring the loss of precision only with error metrics, without checking the topology preservation, is not a good way to evaluate watermarked vector map data quality.

A recent review Abubahia and Cocea (2017) outlines that the most used metrics for watermarked vector map fidelity are RMSE and PSNR, which are both error metrics based on the mean square error. The output of error metrics gives an indication of the precise loss caused by the watermarking process. Over the last 10 years, the research community on watermarking vector map data has repeatedly posed that error metrics are not appropriate for the evaluation of watermarked vector map topological quality (Abubahia and Cocea, 2017; Niu et al., 2006; Sangita and Venkatachalam, 2012a).

A limited number of works have discussed topology preservation in the evaluation of watermarked vector maps (Huang et al., 2010; Kim, 2010a,b; Sangita and Venkatachalam, 2012a; Zope-Chaudhari et al., 2015). These works are outlined below. In (Kim, 2010a,b), the authors used what they call an intersection test to verify if modifications occurred in the topology of line-based maps – more specifically, they assessed if lines that intersected previously to watermarking still intersect and if lines that should not intersect still do not intersect after watermarking. They report that they compared the values of the test before and after the watermark embedding, without details of how this was done, and that based on that comparison they concluded that topology was preserved.

In the work of Huang et al. (2010), the authors looked at polygon closure, data topology, error analysis and visual analysis. They also point out that in previous work data quality is mainly assessed through error metrics borrowed from image watermarking. They focused on tools for data inspection of watermarked vector data that allows visual identification of polygon disclosure, self-intersect, self-overlay and overlay for lines.

Like the work of Huang et al. (2010), in the work of Sangita and Venkatachalam (2012a) the authors also focus on the visual inspection of topological issues without proposing a metric to quantify them; however, through this visual inspection, they stress the need for watermarking approaches that retain the topology of vector data and that the error analysis on its own is not an appropriate way of evaluating watermarking vector data approaches. In more recent work Zope-Chaudhari et al. (2015) investigate the data accuracy (i.e. the difference in coordinates values between the original and the watermarked map[1]). The authors talk about the assessment of distortion, but they only look at data accuracy and assess it with error metrics.

In summary, previous work highlighted the importance of topology preservation and proposed visual inspection for identifying distortions after watermarking. In this Chapter, to take this work further, a metric for quantifying topological distortions of polygon-based vector maps is proposed. The next section describes the proposed metric.

7.2 Metric for Topological Distortion

This section presents the proposed metric for judging the topological quality of watermarked GIS vector maps in line with the required standards for spatial data analysis tasks. Such standards are identified by several organisations working with and regulating the use of spatial

[1] some researchers use the term fidelity to mean both data accuracy and invisibility; while other researchers distinguish between these terms, which is also the case for the work discussed in this Chapter

data. Here, this Chapter follows the topological rules defined by the Environmental Systems Research Institute (ESRI), which supports the OCG[2] and ISO/TC211[3] geospatial standards.

ESRI defined a set of polygon-based shapefiles topology rules [4] to ensure the quality of polygon maps for spatial analysis tasks. In relation to the research of digital vector map watermarking, the significant rules are:

- Each polygon must be in the form of closed shape. A polygon is defined by a series of points, with the first point being the same as the last point; if the first and the last point are not the same, the polygon is not closed.
- Polygons must not overlap each other. This rule specifies that the interior of polygons must not overlap; polygons can only share edges or vertices.
- The map must not have gaps between polygons. This rule specifies that there should be no voids within a polygon or between neighboring polygons, so that all polygons form a continuous surface.

In this Chapter, three metrics are proposed in relation to these rules by quantifying the number of times the rules are broken proportionately to the size of the watermark. Also an overall metric as an average of the three metrics is defined, which can be used to compare topological problems across different watermarking approaches and map sizes. The metrics and the way they are calculated are described in the following subsections.

7.2.1 Polygon Disclosure

The polygon shape is formed by a sequence of vertices where the coordinates of the first point and the last point must be the same. Polygon disclosure occurs when this constraint is not met, i.e. the coordinates of the first and the last point are different.

In the watermarking process, there is a potential of having the polygon disclosure issue since the process of inserting the watermark is modifying the redundant bits of data, and the modification of different points may be done in different ways. For example, adding a watermark bit of 1 to the first point, while adding a watermark bit of -1 to the last point, would lead to disclosure.

Consequently, it is important to assess whether the polygon closure has been affected by the watermarking process. For this purpose, the condition used is that the coordinate value

[2] `http://www.opengeospatial.org/docs/is`

[3] `http://www.isotc211.org/`

[4] `http://help.arcgis.com/en/arcgisdesktop/10.0/help/001t/pdf/topology_rules_poster.pdf`

pair of the first point and the coordinate value pair of the last point must be the same, as shown in Equations (7.1) and (7.2).

$$F_x = L_x \tag{7.1}$$

and

$$F_y = L_y \tag{7.2}$$

where F_x is the x-coordinate of the first point, L_x is the x-coordinate of the last point, F_y is the y-coordinate of the first point and L_y is the y-coordinate of the last point.

The metric for polygon disclosure in the watermarked map is defined in Equation (7.3) as the proportion of disclosed polygons from all watermarked polygons:

$$M_1 = \frac{\sum_{i=1}^{n_w} d_i}{n_w} \tag{7.3}$$

where M_1 represents the disclosure metric, n_w represents the number of watermarked polygons and d_i is defined as in Equation (7.4):

$$d_i = \begin{cases} 1, & \text{if } F_x \neq L_x \\ 1, & \text{if } F_y \neq L_y \\ 0, & \text{otherwise} \end{cases} \tag{7.4}$$

for each polygon i, where i takes values from 1 to n_w.

7.2.2 Overlap and Gap Identification

The overlap within the map polygons is a potential issue after inserting the watermark bits. This affects the map topology against the rule that the interior of polygons must not overlap, which means that an area cannot be shared by two or more polygons, i.e. polygons can only share edges or vertices. For example, the satisfaction of this topology rule is important for modeling administrative boundaries, such as voting districts, postal codes or land cover type.

The gaps between the map polygons could also be a consequence of the watermark insertion process, which has the effect of creating voids between adjacent polygons, while the topology rule requires that all polygons must form a continuous surface. This rule is significant in the context of spatial data analysis because it changes the perimeter of the surface. For example, when polygons define the type of soil in a particular area, there should be no gaps between polygons, i.e. the entire area needs to be defined in terms of the soil type; a gap would mean that the soil type (for the surface defined by this gap) is not known.

Algorithm 3 shows how the number of overlaps and gaps are identified. The *inpolygon* function in Matlab is used for this purpose, which establishes if a point is in or on the edge of a polygon. Thus, for all watermarked vertices, this function is applied with reference to the original polygon. If the watermarked vertex is within the original polygon, a gap is created, while if the watermarked vertex is outside the original polygon, an overlap is created.

Algorithm 3 Overlap_Gap_Calculation

```
Input  : The original and watermarked maps: $M_o$, $M_w$
Output: $Gaps$, $Overlaps$
$sum_1 = 0$
$sum_2 = 0$
$sum_3 = 0$
for each watermarked polygon $P_w$ in the watermarked map $M_w$ do
    $[in, on] = inpolygon(x_{P_w}, y_{P_w}, x_{P_o}, y_{P_o})$
    // $x_{P_w}$ and $y_{P_w}$ are vectors holding the $x$ and $y$ coordinates values of the
       watermarked polygon $P_w$; $x_{P_o}$ and $y_{P_o}$ are vectors holding the $x$ and $y$
       coordinates values of the corresponding original polygon $P_o$
    // $in$ indicates if the points are inside or on the edge of the polygon; $on$
       indicates if the points are on the edge of the polygon
    $sum_1 = sum_1 + numel(x_{P_w}[in])$
    // the number of points inside or on the edge of the polygon
    $sum_2 = sum_2 + numel(x_{P_w}[on])$
    // the number of points on the edge of the polygon
    $sum_3 = sum_3 + numel(x_{P_w}[\sim in])$
    // the number of points outside the edge of the polygon
end
$Gaps = sum_1 - sum_2$
// the number of points inside the original polygons for the whole map
$Overlaps = sum_3$
// the number of points outside the original polygons for the whole map
return $Gaps$, $Overlaps$
```

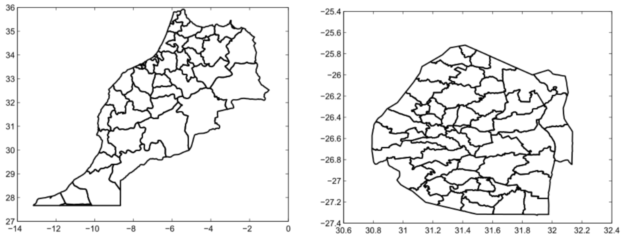
The quantified measure for the overlap issue in the watermarked map is defined in Equation (7.5) as the proportion of overlapping polygons from all watermarked polygons:

$$M_2 = \frac{\sum_{i=1}^{V_w} V_{oi}}{V_w} \tag{7.5}$$

where M_2 represents the overlap metric, V_w represents the number of watermarked vertices and V_o represents the number of vertices placed outside their original polygon after watermarking, thus leading to overlaps.

The quantified measure for the gap issue in the watermarked map is defined in Equation (7.6) as the proportion of gaps between polygons from all watermarked polygons:

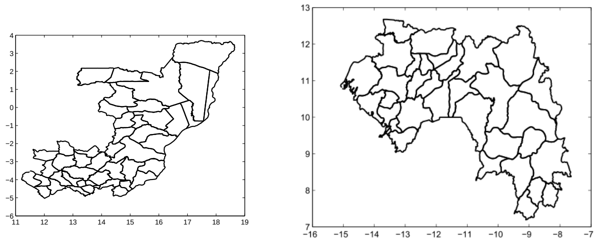
$$M_3 = \frac{\sum_{i=1}^{V_w} V_{gi}}{V_w} \tag{7.6}$$



(a) Map of Morocco (47 polygons, 7523 vertices)

(b) Map of Swaziland (53 polygons, 7678 vertices)

Figure 7.1: Dataset 1.



(a) Map of Congo-Brazzaville (46 polygons, 12511 vertices)

(b) Map of Guinea (56 polygons, 21304 vertices)

Figure 7.2: Dataset 2.

where M_3 represents the gap metric, V_w represents the number of watermarked vertices and V_g represents the number of vertices placed within their original polygon after watermarking, thus leading to gaps.

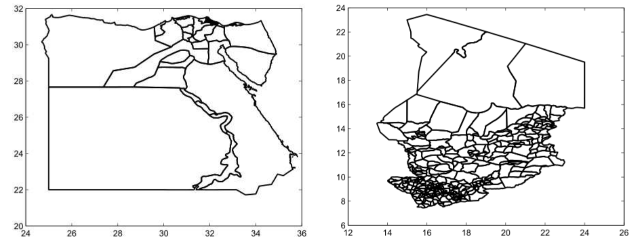
7.2.3 The Overall Metric

The overall metric is defined as the average of disclosure, overlap and gap measurements that were described in the previous subsections – see Equation (7.7).

$$M = \frac{\sum_{i=1}^{3} M_i}{3} \tag{7.7}$$

where M represents the overall fidelity metric, M_1 represents the disclosure metric, M_2 represents the overlap metric and M_3 represents the gap metric.

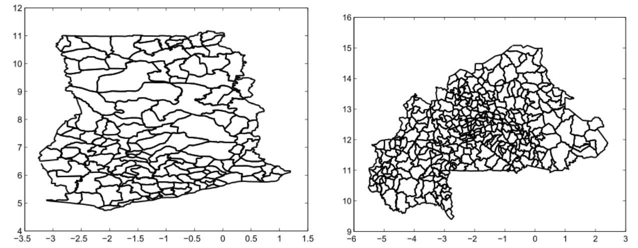
For all metrics, the values are between 0 and 1, where a value of 0 indicates no topology problems, and 1 indicates the maximum number of topology problems. For example, for the overall metric a value on 1 means that all watermarked polygons are disclosed and that overlaps and gaps take place for all watermarked vertices.



(a) Map of Egypt (129 polygons, 5992 vertices)

(b) Map of Chad (347 polygons, 19542 vertices)

Figure 7.3: Dataset 3.



(a) Map of the Ghana (138 polygons, 243329 vertices)

(b) Map of Burkina Faso (351 polygons, 113996 vertices)

Figure 7.4: Dataset 4.

7.3 Experiments

This section describes the experiments that are conducted for the evaluation of the proposed metrics, including the data used and the way of controlling the embedding of the watermark to assess the comparability of the results across maps and watermarks of different sizes.

7.3.1 Data Description and Experimental Setup

To evaluate if the metrics allow comparisons for maps of different sizes in terms of number of polygons and number of vertices, four datasets (of two maps each) combining high and low numbers of polygons and vertices were used, respectively:

- Dataset 1 includes maps with small number of polygons and small number of vertices.
- Dataset 2 includes maps with small number of polygons and large number of vertices.
- Dataset 3 includes maps with large number of polygons and small number of vertices.
- Dataset 4 includes maps with large number of polygons and large number of vertices.

Within each dataset, the two maps are chosen to represent opposite ratios of number of polygons to number of vertices, i.e. one map has on average a smaller number of vertices per polygon compared with the other map in the same dataset.

Also, the size of the watermark is controlled, i.e. 25%, 33% and 50% of the original map, to show that the metrics can be used to compare watermarked maps not only of variable map size, but also variable watermark size.

Table 7.1 lists the maps of the four datasets, their number of polygons and vertices, the average number of vertices per polygon, as well as the number of polygons that correspond to the proportions of 25%, 33% and 50%, which are used when embedding the watermark. Figures 1 to 4 illustrates the eight maps of the four datasets.

Table 7.1: The datasets (D) with corresponding number of polygons (#P), vertices (#V) and number of polygons for proportions of map size.

D	Map	#P	#V	Avg	Proportions		
					25%	33%	50%
1	Morocco (MOR)	47	7523	160	12	16	24
	Swaziland (SWA)	53	7678	144	14	18	27
2	Congo-Brazzaville (CNG)	46	12511	271	12	16	23
	Guinea (GIN)	56	21304	380	14	19	28
3	Egypt (EGY)	129	5992	46	33	43	65
	Chad (CHA)	347	19542	56	87	116	174
4	Ghana (GHA)	138	243329	1763	35	46	69
	Burkina Faso (BUF)	351	113996	324	88	117	176

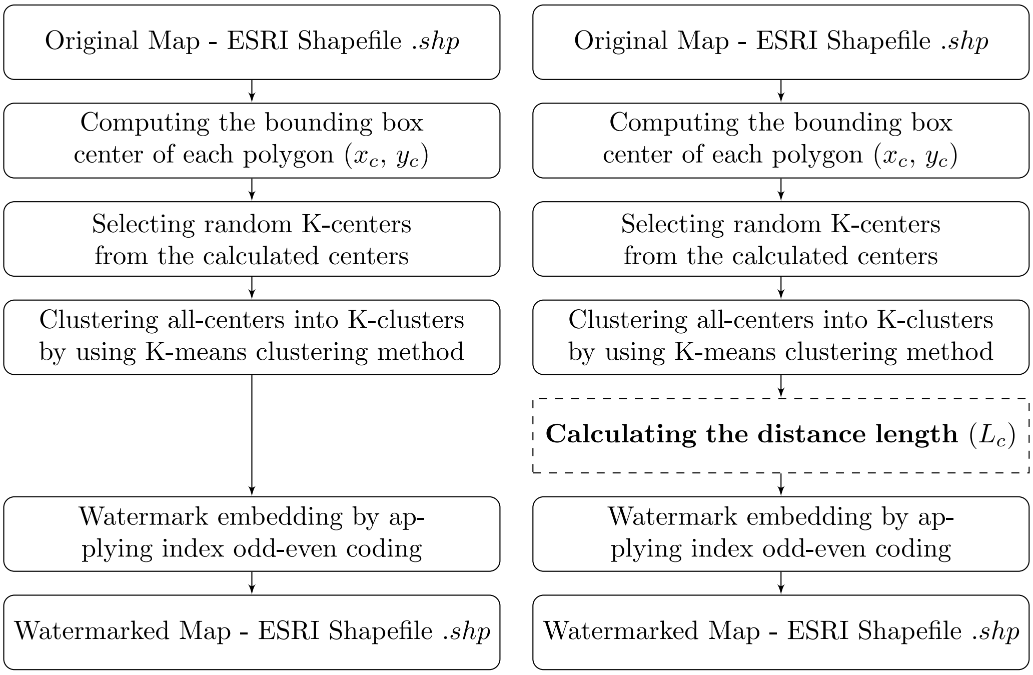
The proposed metrics are defined in relation to the watermark size to allow comparison across maps and watermarks of different sizes. This relativity to the watermark size should results in the experiments in similar metrics values for all the maps within the same dataset, as well as across all datasets. In other words, the experiments were set up to show that regardless of map size, comparisons on the distortions introduced by watermarking still can be made.

The maps used in the experiments are freely available, in ESRI shapefile format, from the map maker website[5]. Maps that are freely available Were used to facilitate the development of benchmarks in the context of vector data, as one of the important aspects of bringing research in this area forward, by making it possible to compare different developments.

ESRI Shapefiles (.shp) are produced by ESRI[6], and considered as a popular format for geographic information system applications (Burrough et al., 2013). They have several key features: small storage space, easy reading and writing, fast shape editing, storing both spatial and attribute information, and supporting point, polyline and polygon geometry types (ESRI, 1998).

[5] http://www.mapmakerdata.co.uk

[6] http://www.esri.com/


Original Map - ESRI Shapefile .shp
Computing the bounding box center of each polygon (x_c, y_c)
Selecting random K-centers from the calculated centers
Clustering all-centers into K-clusters by using K-means clustering method
Watermark embedding by applying index odd-even coding
Watermarked Map - ESRI Shapefile .shp
Original Map - ESRI Shapefile .shp
Computing the bounding box center of each polygon (x_c, y_c)
Selecting random K-centers from the calculated centers
Clustering all-centers into K-clusters by using K-means clustering method
Calculating the distance length (L_c)
Watermark embedding by applying index odd-even coding
Watermarked Map - ESRI Shapefile .shp

(a) The coordinates based approach

(b) The distance based approach

Figure 7.5: Two different watermark insertion approaches

The two most-known watermark embedding approaches were implemented in MATLAB version R2014b (8.4.0.150421) on a 64-bits Windows-PC. The way watermarks of different sizes were embedded, is explained in the following section.

7.3.2 Watermark Insertion Process

For the watermark embedding process, two main prevalent approaches were used and compared: (1) a coordinate-based approach (shown in Fig.7.5a) and (2) a distance based approach (shown in Fig.7.5b). These approaches have shown, practically, a better resilience to map changes/attacks such as: rotation, translation, scaling, simplification and interpolation (Abubahia and Cocea, 2015b,a). In both approaches, clustering is used to control the size of the watermark in relation to map size, as well as distribute the watermark throughout the map. Clustering is used to identify locations in the map for embedding the watermark (Abubahia and Cocea, 2014).

Both approaches mentioned above uses the bounding box property in ESRI shapefiles, which identifies the boundaries of each polygon in the map (ESRI, 1998). Polygons' bounding box centers are calculated in both axes, as shown in Equation 7.8:

$$x_c = \frac{x_{min} + x_{max}}{2} \quad \& \quad y_c = \frac{y_{min} + y_{max}}{2} \tag{7.8}$$

where x_c and y_c are the coordinates of a polygon's center in x and y axes respectively; x_{min} is the minimum vertex coordinate in the x-axis; x_{max} is the maximum vertex coordinate in the x-axis; y_{min} is the minimum vertex coordinate in the y-axis; y_{max} is the maximum vertex coordinate in the y-axis; x_{min}, x_{max}, y_{min} and y_{max} are each of 8-byte length (ESRI, 1998).

The k-means clustering method is used to cluster the bounding box centers, as the polygons' representatives, in order to determine the positions for embedding the watermark. More precisely, through this process, a number of polygons are identified as locations for embedding the watermark. The k-means method is relatively simple, easy to implement, and needs a predefined number of clusters (k) – see reference Abubahia and Cocea (2015b) for more detail. The experiments were set up with values of k that represent approximately 25%, 33% and 50% of the total number of polygons. In this way, the size of the watermark is controlled, which allows evaluating the proposed metrics for different watermark sizes.

The watermark is constructed by adding or subtracting a bit value of 1 from either x and y vertex coordinate values (coordinate-based approach) or distance length values (distance-based approach) within the selected polygons (identified by k-means clustering).

The watermark is embedded by applying odd-even indexing, which is one of the most popular embedding approaches (Baiyan et al., 2008a; Huo et al., 2011b; Abubahia and Cocea, 2015a,b, 2014). This approach is formally represented as in Equation (7.9).

$$W_i = \begin{cases} T-1, & \text{if OES(I)=odd} \\ T+1, & \text{if OES(I)=even} \end{cases} \tag{7.9}$$

where W_i is the ith bit value of the watermark; OES stands for Odd-Even Status; I is the order index of the watermark embedding position value; T is the value of the 4th digit of the embedding position value, after the decimal point. The following two subsections detail the embedding procedure for the coordinate-based and distance-based approaches.

Coordinates-based Embedding

In this approach, the embedding space is the x and y vertex coordinate values. The watermark is embedded by comparing the OES (Odd-Even Status) of I which represents the sequential order of the vertex within the set of polygon's vertices. As shown in Equation (7.10), the conditions are set based on two scenarios: (a) if the OES of I is odd, 1 will be subtracted from the value of T, which represents the 4th bit after the decimal point of the x and y vertex coordinate values; (b) if the OES of I is even, 1 will be added to the value of T.

$$v_x^* = v_x \pm 0.0001 \quad \& \quad v_y^* = v_y \pm 0.0001 \tag{7.10}$$

where v_x^* and v_y^* are the new vertices' coordinates after embedding the watermark according to the aforementioned condition, in Equation (7.9); v_x and v_y are the original vertices' coordinates before inserting the watermark bits.

Distance-based Embedding

In this approach, the embedding space is the mean distance length values. The distance length is calculated by measuring the distance from the polygon bounding box top right corner to its center, as illustrated in Equation (7.11).

$$L_c = \sqrt{(x_c - x_{max})^2 + (y_c - y_{max})^2} \tag{7.11}$$

where L_c is the distance length; x_c and y_c are the center coordinates in x and y axes, respectively; x_{max} and y_{max} are the top right bounding box corner coordinates in the x and y axes, respectively.

As shown in Equation (7.9), the watermark is embedded by comparing the OES (Odd-Even Status) of the I variable, which represents the order index of the mean-distance length values. Similarly to the coordinate-based approach, the conditions are set based on two scenarios: (a) if the OES of I is odd, 1 will be subtracted from the value of T; (b) if the OES of I is even, 1 will be added to the value of T.

After applying the OES to change the values of L_c, the new values of distance length will be represented by L_c^*. The change rate α_c is calculated as depicted in Equation (7.12):

$$\alpha_c = \frac{L_c^*}{L_c} \tag{7.12}$$

The change rate α_c is used to change all vertices of polygons that belong to each cluster's center on the basis of the embedding condition, as given in Equation (7.13).

$$v_x^* = \alpha_c v_x + x_c(1 - \alpha_c) \quad \& \quad v_y^* = \alpha_c v_y + y_c(1 - \alpha_c) \tag{7.13}$$

Both embedding approaches should lead to contrasted readings in overlaps and gaps as the size of the watermark increases; the same should occur for disclosures for the coordinate-based approach (the distance-based approach does not lead to disclosures). In other words, the more watermark bits are included, the more issues with topology will occur. As a metric should allow comparison across different map sizes, as well as watermark size (and not simply penalise bigger watermarks), the metrics are defined as the number of topological issues (disclosures/gaps/overlaps) relative to the watermark size. Consequently, similar metrics were expected across the maps of different size and across the different sizes of watermarks, with some expected variety due to the randomness involved in the selected polygons for embedding (with varying numbers of vertices) and the odd-even status of the embedding locations; these random variations are further discussed in the next section.

Consequently, to show the reliability of the overall metric, the experimental results should show the following:

1. The disclosure metric for the coordinate-based approach will depend on the number of vertices in the watermarked polygons, thus leading to variations unrelated to the map size or watermark size; if all watermarked polygons have an even number of vertices, there will be no disclosures, while if all watermarked polygons have an odd number of vertices, all will have disclosures. The probability for a watermarked polygon to have either an odd or an even number of polygons is 0.5; thus, for higher numbers of watermarked polygons, the $M1$ metric would be expected to have values around 0.5, while for fewer watermarked polygons, a higher variety would be expected in the metrics' values.

2. The gaps and overlaps metrics for both embedding approaches should have very similar values; since all watermarked vertices will lead to either a gap or an overlap, two phenomena are expected: (a) approximately half of the vertices will lead to gaps and half to overlaps, which would results in values of approximately 0.5 for metrics $M1$ and $M2$; (b) when the previous does not happen due to randomness, there will be a complementarity between the number of gaps and overlap, i.e. the more gaps, the fewer overlaps;

3. The overall metric for the coordinate-based approach will follow the variation in the disclosure metric, as it is an average of the disclosure, overlaps and gaps metrics, and the overlaps and gaps metrics should display little variation;

4. The overall metric for the distance-based approach should be very similar for all maps and all watermark sizes, as there are no disclosures for this embedding approach, and the overlaps and gaps metrics should be complementary (i.e. the more gaps, the fewer overlaps).

The next section presents the results and discusses them in terms of the expectations outlined above.

7.4 Results and Discussion

This section presents the results of the experiment in relation to the three metrics corresponding to the three topology rules for polygons, as well as the overall metric. The results are discussed in relation to the experimental setup and the expectations outlined in the previous section.

Table 7.2: The disclosure metric for the coordinate-based embedding method; Notes: n_w = number of watermarked polygons; D = number of disclosures; M_1 = disclosure metric.

Dataset	Map	n_w	Coordinate	
			D	M_1
1	MOR (25%)	12	3	0.25000
	MOR (33%)	16	3	0.18750
	MOR (50%)	24	9	0.37500
	SWA (25%)	14	8	0.57143
	SWA (33%)	18	9	0.50000
	SWA (50%)	27	15	0.55556
2	CNG (25%)	12	4	0.33333
	CNG (33%)	16	6	0.37500
	CNG (50%)	23	11	0.47826
	GIN (25%)	14	9	0.64286
	GIN (33%)	19	11	0.57895
	GIN (50%)	28	17	0.60714
3	EGY (25%)	33	14	0.42424
	EGY (33%)	3	22	0.51163
	EGY (50%)	65	29	0.44615
	CHA (25%)	87	50	0.57471
	CHA (33%)	116	69	0.59483
	CHA (50%)	174	92	0.52874
4	GHA (25%)	35	18	0.51429
	GHA (33%)	46	26	0.56522
	GHA (50%)	69	38	0.55072
	BUF (25%)	88	44	0.50000
	BUF (33%)	117	61	0.52137
	BUF (50%)	176	86	0.48864

The disclosure metrics for all datasets are given in Table 7.2 and Fig. 7.6; this is just for the coordinate-based approach, as for the distance-based approach there are no disclosures due to the embedding process.

As expected, the results show an increase in disclosures proportionate to the watermark size, i.e. the larger the watermarks, the higher the number of disclosures – see the 4th column (D) in Table 7.2. The M_1 metric does not entirely preserve this proportions (see Fig. 7.6) due to the randomness involved in the odd-even status of the number of vertices in a polygon, i.e. if the watermark is added to a polygon with an odd number of vertices, there will be no disclosure, while if the watermark is added to a polygon with an even number of vertices, there will be a disclosure.

When looking at the variations of the M_1 metric for the same map with different water-

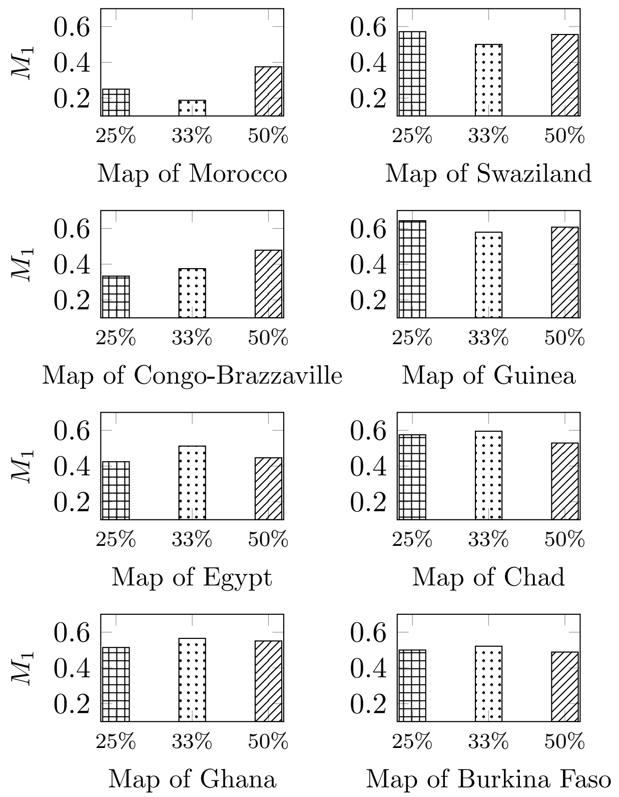


Figure 7.6: Coordinate-based method disclosure metrics (M_1).

mark sizes, it is noticeable that these are relatively small with most differences smaller than 0.09. The biggest variations take place for the MOR (0.19) and CNG (0.15) maps, which is not surprising since these are the maps with the smallest number of polygons (at it is known that the randomness effect stabilizes for larger numbers). Unsurprisingly, the smallest variation occurs for BUF (0.03), which is the map with the highest number of polygons.

The experimental results for the overlap metric (M_2) are displayed in Table 7.3, Fig. 7.7 and Fig. 7.8, for both watermarking approaches.

As expected, the higher the number of watermarked vertices, the higher the number of overlaps (columns 4 and 6 in Table 7.3). The only exception to this is for the Map of Egypt, where the 33% watermark results in fewer watermarked vertices than the 25% watermark. This is due to the embedding procedure in which a number of polygons is selected in which the watermark is inserted, thus, the number of watermarked vertices overall depends on the number of vertices in each polygon selected for embedding. In the case of the Map of Egypt–33%, the polygons selected for the embedding of the watermark had fewer vertices overall than the polygons selected for the Map of Egypt–25%.

As expected, for both embedding approaches, overlaps metrics are very similar regardless

Table 7.3: The overlap metrics for coordinate-based and distance-based embedding methods; Notes: V_w = number of watermarked vertices; O = number of overlaps; M_2 = overlap metric

Dataset	Map	V_w	Coordinate		Distance	
			O	M_2	O	M_2
1	MOR (25%)	2105	1067	0.50689	1094	0.51971
	MOR (33%)	2729	1382	0.50641	1386	0.50788
	MOR (50%)	4275	2165	0.50643	2225	0.52047
	SWA (25%)	1808	922	0.50996	1093	0.60454
	SWA (33%)	2793	1419	0.50806	1559	0.55818
	SWA (50%)	4174	2119	0.50767	2424	0.58074
2	CNG (25%)	3510	1770	0.50427	1860	0.52991
	CNG (33%)	4194	2115	0.50429	1682	0.40105
	CNG (50%)	6036	3043	0.50414	2720	0.45063
	GIN (25%)	6277	3138	0.49992	3115	0.49626
	GIN (33%)	9046	4526	0.50033	4397	0.48607
	GIN (50%)	13887	6947	0.50025	6930	0.49903
3	EGY (25%)	4055	2065	0.50925	2126	0.49824
	EGY (33%)	2855	1478	0.51769	1612	0.56462
	EGY (50%)	4504	2328	0.51687	2467	0.54774
	CHA (25%)	4887	2538	0.51934	2486	0.50870
	CHA (33%)	6933	3595	0.51853	3782	0.54551
	CHA (50%)	10004	5187	0.51849	5082	0.50800
4	GHA (25%)	59299	29417	0.49608	30301	0.51099
	GHA (33%)	94058	46648	0.49595	49442	0.52565
	GHA (50%)	133860	66401	0.49606	70292	0.52513
	BUF (25%)	26270	13206	0.50270	13886	0.52859
	BUF (33%)	36404	18304	0.50280	18677	0.51305
	BUF (50%)	54854	27593	0.50303	29217	0.53263

of map size and watermark size. For the same maps with different watermark sizes, for the coordinate-based approach, the average difference is 0.00109 with a standard deviation of 0.00221. For the distance-based approach, the average is 0.03041 and the standard deviation is 0.03166.

Overall, the overlap metric for all maps ranges between 0.49595 and 0.51934 for the coordinate-based approach and between 0.40105 and 0.60454 for the distance-based approach. Thus, it is noticeable that the coordinate-based approach leads to more similar values than the distance-based approach.

Table 7.4, Fig. 7.9 and Fig. 7.10 displays the gap metrics for both coordinate-based and distance-based approaches. As expected, the more vertices are watermarked, the more gaps occur, with the exception for the Map of Egypt mentioned previously for overlaps - since the gap metric, like the overlap one, is influenced by the total number of vertices in the watermarked polygons, the same effect occurs.

For the same maps with different watermark sizes, for the coordinate-based approach the average difference is 0.00120 and the standard deviation is 0.00235. For the distance-based approach, the average is 0.03108 and the standard deviation is 0.03125.

Overall, the gap metrics range between 0.48147 and 0.50405 for the coordinate-base approach and between 0.39546 and 0.59895 for the distance-based approach. Similar the over-

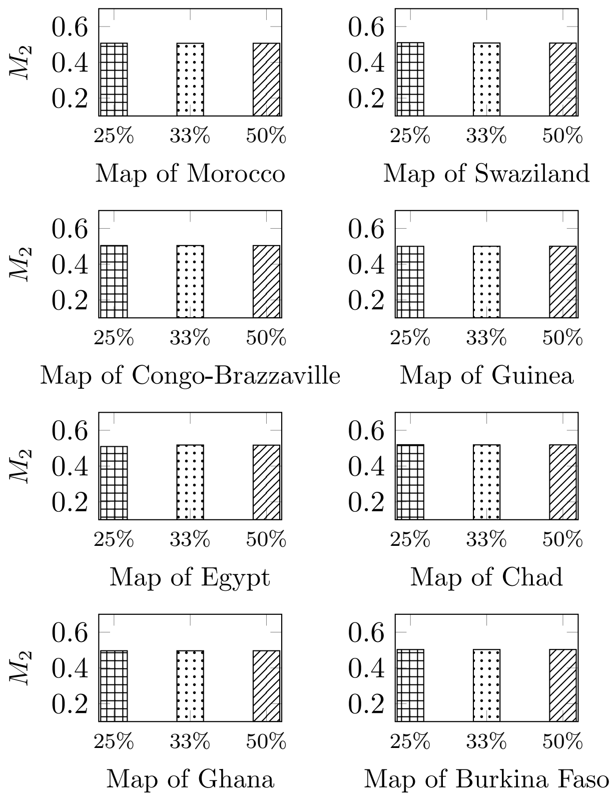


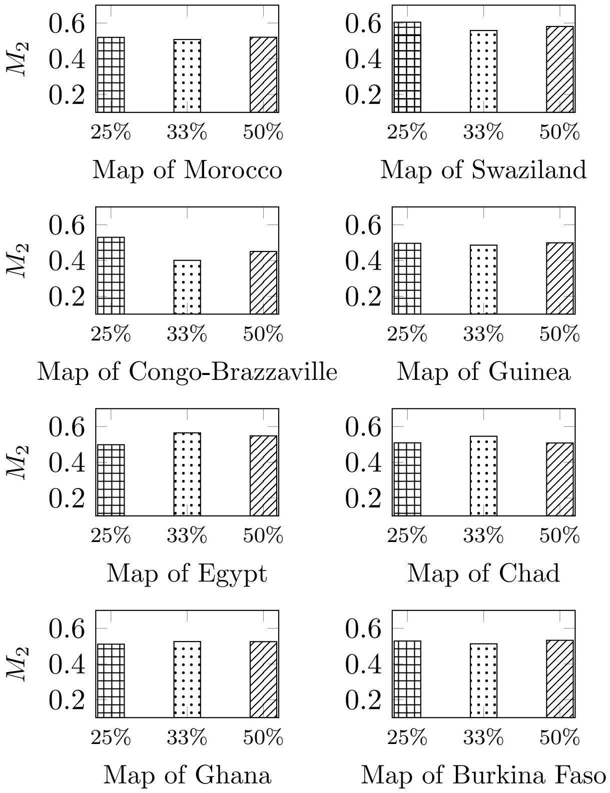
Figure 7.7: Coordinate-based approach overlap metric (M_2).

laps metric, it is noticeable that a smaller range occurs for the coordinate-based approach compared with the distance-based approach.

For the overall metrics, the results are displayed in Table 7.5, Fig. 7.11 and Fig. 7.12. For the coordinate-based approach, the overall metric values are between 0.39583 and 0.54762, while for the distance-based approach the metrics are 0.33333 for all maps and all watermark sizes. For the distance-based approach, the same values are occurring due to the lack of disclosures (thus, the lower value) and the complementarity between gaps and overlaps (i.e. a watermarked vertex will lead to either a gap or an overlap), i.e. when more gaps occur, there are fewer overlaps (as reflected in the $M2$ and $M3$ metrics).

For example, the SWA (25%) map has a large number of overlaps reflected in a high $M2$ metric, i.e. 0.60454, and a lower number of gaps reflected in a low $M3$ metric, i.e. 0.39546 (the two metrics add up to 1); the $M2$ and $M3$ metrics add up to 1 for all maps. As there are no disclosures, and each metric has the same weight, the overall metric becomes 1/3, i.e 0.33333.

The experiments were set up with the purpose of showing that the metrics allow comparisons between maps of different sizes, as well as different watermark sizes. More specifically,



Map of Morocco

Map of Swaziland

Map of Congo-Brazzaville

Map of Guinea

Map of Egypt

Map of Chad

Map of Ghana

Map of Burkina Faso

Figure 7.8: Distance-based approach overlap metric (M_2).

this work looked at a variety of maps grouped into four datasets covering the different combination of number of polygons and number of vertices. Moreover, within the same dataset, maps that had opposite ratios of numbers of vertices per polygon were chosen. The results show that the metrics are comparable across this variation in map size properties, with a few exceptions explained by the randomness involved in the embedding process.

By looking at different watermark sizes, the metrics were tested in terms of their accurate reflection of the number of distortions. As the number of distortions are proportionate to the size of the watermark, an increase in the number of distortions were expected as the size of the watermark increased, which has been shown in the results. Because the metrics are defined as the number of distortions relative to the size of the watermark, it is expected that the metrics for the same map with the different watermark sizes would be very similar, with only small differences in values.

The results showed this consistency in the values of the metrics between the same map with watermarks of different size. The results were more consistent for the overlap and gap metrics than for the disclosure metric for the coordinate-based approach. The higher variability in the disclosure metric could be explained as a consequence of the odd-even

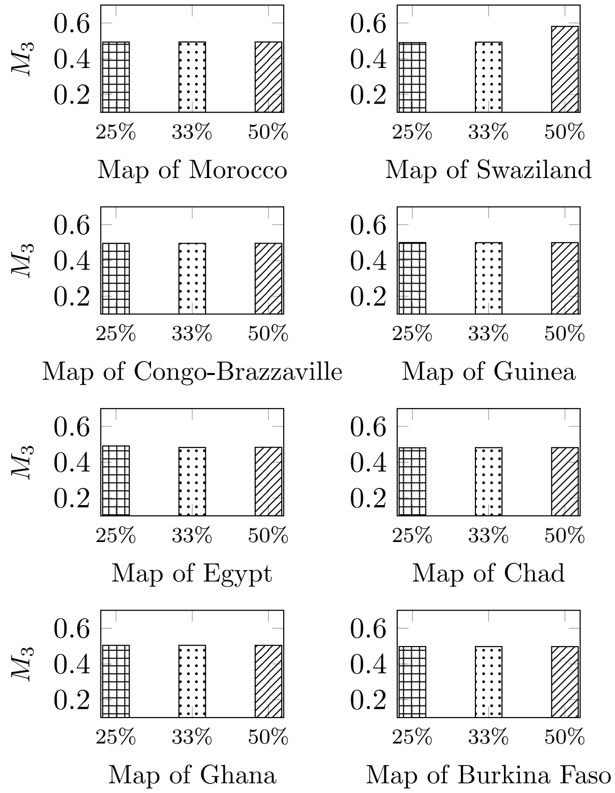


Figure 7.9: Coordinate-based approach gap metric (M_3).

Table 7.4: The gap metrics for coordinate-based and distance-based embedding methods: Notes: V_w = number of watermarked vertices; G = number of gaps; M_3 = gaps metric.

Dataset	Map	V_w	Coordinate (Wang et al., 2007)		Distance (Huo et al., 2011b)	
			G	M_3	G	M_3
1	MOR (25%)	2105	1038	0.49311	1011	0.48029
	MOR (33%)	2729	1347	0.49359	1343	0.49212
	MOR (50%)	4275	2110	0.49357	2050	0.47953
	SWA (25%)	1808	886	0.49004	715	0.39546
	SWA (33%)	2793	1374	0.49194	1234	0.44182
	SWA (50%)	4174	2055	0.49233	1750	0.41926
2	CNG (25%)	3510	1740	0.49573	1650	0.47009
	CNG (33%)	4194	2079	0.49571	2512	0.59895
	CNG (50%)	6036	2993	0.49586	3316	0.54937
	GIN (25%)	6277	3139	0.50008	3162	0.50374
	GIN (33%)	9046	4520	0.49967	4649	0.51393
	GIN (50%)	13887	6940	0.49975	6957	0.50097
3	EGY (25%)	4055	1990	0.49075	2141	0.50176
	EGY (33%)	2855	1377	0.48231	1243	0.43538
	EGY (50%)	4504	2176	0.48313	2037	0.45226
	CHA (25%)	4887	2349	0.48066	2401	0.49130
	CHA (33%)	6933	3338	0.48147	3151	0.45449
	CHA (50%)	10004	4817	0.48151	4922	0.49200
4	GHA (25%)	59299	29882	0.50392	28998	0.48901
	GHA (33%)	94058	47410	0.50405	44616	0.47435
	GHA (50%)	133860	67456	0.50394	63565	0.47487
	BUF (25%)	26270	13064	0.49730	12384	0.47141
	BUF (33%)	36404	18100	0.49720	17727	0.48695
	BUF (50%)	54854	27261	0.49697	25637	0.46737

indexing used in the embedding process. Another aspect related to the higher variability in the disclosure metric is the fact that the disclosure metric is defined in relation to the number of watermarked polygons, while the overlap and gap metrics are defined in relation to the number of vertices. As the number of polygons has a smaller range than the number of vertices, the metrics show more variation for the disclosure metric.

7.5 Summary

In this Chapter, the importance of a metric to assess topological distortions in watermarked vector maps is discussed, and a metric for polygon-based vector maps is proposed. This chapter looked at three distortions that can occur when polygon topology rules are broken in the watermarking process: polygon disclosures, overlaps and gaps.

Maps and watermarks of different sizes were used, as well as two different watermarking approaches to test the metrics; thus, four datasets were used, where each dataset had varying degrees of size in terms of number of polygons and number of vertices. Each dataset contained two maps, which had opposite ratios of number of vertices per polygon. By using k-means clustering to embed the watermark, the size of the watermark is controlled and experimented with three sizes corresponding approximately to 25% (16–117 polygons), 33%

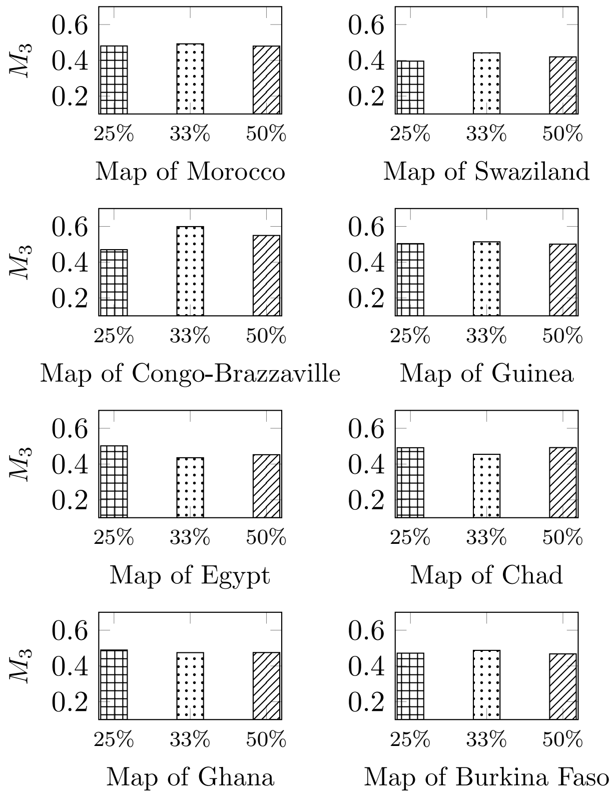


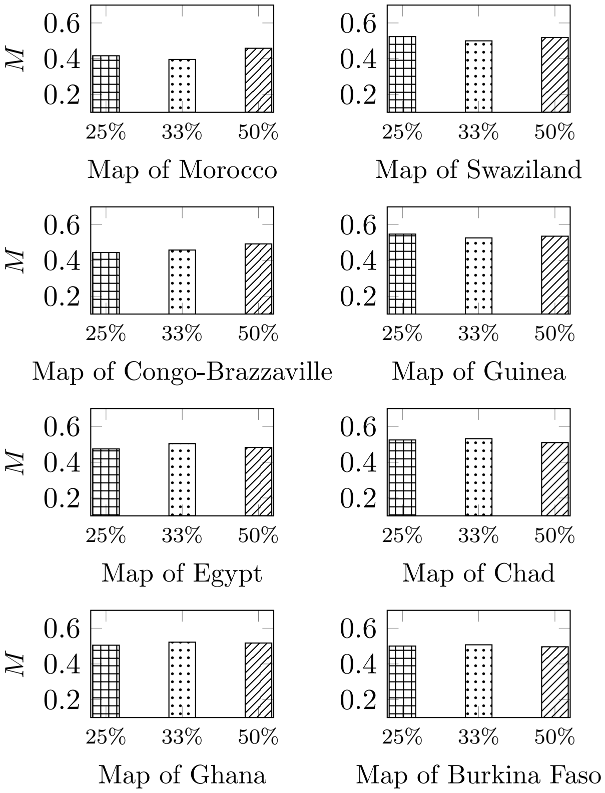
Figure 7.10: Distance-based approach gap metric (M_2).

(12–88 polygons) and 50% (24–176 polygons) of the number of polygons in the original maps. The results indicate that the metrics allow comparisons between watermarked maps of different sizes and of different watermark sizes, and, thus, can be used to asses the quality of watermarked vector maps.

The proposed metric described and tested in this Chapter is a first step towards a standard metric for watermarked vector map quality that assesses topological distortion.

Table 7.5: The overall metric (M) for coordinate-based and distance-based embedding methods.

Dataset	Map	Coordinate	Distance
		M	M
1	MOR (25%)	0.41667	0.33333
	MOR (33%)	0.39583	0.33333
	MOR (50%)	0.45833	0.33333
	SWA (25%)	0.52381	0.33333
	SWA (33%)	0.50000	0.33333
	SWA (50%)	0.51852	0.33333
2	CNG (25%)	0.44444	0.33333
	CNG (33%)	0.45833	0.33333
	CNG (50%)	0.49275	0.33333
	GIN (25%)	0.54762	0.33333
	GIN (33%)	0.52632	0.33333
	GIN (50%)	0.53571	0.33333
3	EGY (25%)	0.47475	0.33333
	EGY (33%)	0.50388	0.33333
	EGY (50%)	0.48205	0.33333
	CHA (25%)	0.52490	0.33333
	CHA (33%)	0.53161	0.33333
	CHA (50%)	0.50958	0.33333
4	GHA (25%)	0.50476	0.33333
	GHA (33%)	0.52174	0.33333
	GHA (50%)	0.51691	0.33333
	BUF (25%)	0.50000	0.33333
	BUF (33%)	0.50712	0.33333
	BUF (50%)	0.49621	0.33333



Map of Morocco

Map of Swaziland

Map of Congo-Brazzaville

Map of Guinea

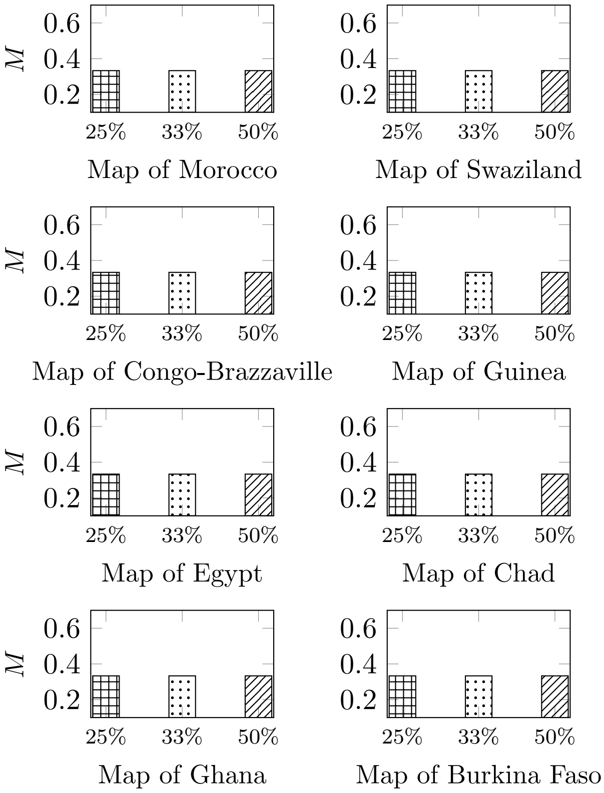
Map of Egypt

Map of Chad

Map of Ghana

Map of Burkina Faso

Figure 7.11: Coordinate-based overall metric (M).



Map of Morocco

Map of Swaziland

Map of Congo-Brazzaville

Map of Guinea

Map of Egypt

Map of Chad

Map of Ghana

Map of Burkina Faso

Figure 7.12: Distance-based overall metric (M).

Chapter 8

Workload Balanced GIS Map Clustering

The growing demand for Geographic Information Systems (GIS) calls for high computation reliability, as the spatial data grow rapidly in volume and require more complex computation. Integrating parallel computing and spatial analysis tasks provides a promising solution to the complexity of GIS data processing (Wei et al., 2015; McKenney and Schneider, 2007; Shuliang et al., 2013; Fu et al., 2011; Wang et al., 2015c; Zhang et al., 2016).

GIS data can be categorized into two main structures[1]: raster data structure and vector data structure. The raster structure (image) stores the geographic information into the form of grid cells, and each cell represents the natural corresponding value on the ground (e.g. color scale). On the other hand, the vector data structure stores the geographic information into geometrical entities which have properties such as length, a starting point and an ending point. GIS vector data is defined by a sequence of coordinates, and includes shapes such as points, polylines and polygons. This Chapter focuses on the vector polygon type of GIS data.

The MapReduce (Dean and Ghemawat, 2004) model implementation for parallel computing shows a considerable efficiency in handling large-scale datasets in general. MapReduce, as its name suggests, refers to two separate tasks: the mapping task and the reduce task. In the map task the whole data set is partitioned into several smaller sub-sets, which are indexed by using tuples on the basis of key/value pairs. In the reduce task, the results of processing the sub-sets are combined by using the key/value pairs to form an output for the whole dataset.

A better parallel computing scheme should ensure balanced workload at different data processors to ensure optimal use of computing resources (i.e. distribute the workload equally

[1] http://www.ordnancesurvey.co.uk/support/understanding-gis/raster-vector.html

to all processors, rather than having some overloaded processors and some idle ones), which poses more challenges with GIS data (Zhao et al., 2016; Wei et al., 2015; Qiu et al., 2015; Gufler et al., 2012; Dean and Ghemawat, 2004). In the context of GIS vector data, the most known examples of MapReduce implementation are Spatial-Hadoop (Eldawy and Mokbel, 2015) and GIS-Hadoop (Aji et al., 2013). They have been introduced as potential solutions for parallel spatial data processing. Although these systems have shown a good performance in terms of spatial data storage and query processing, they still lack the partition strategy that meets the workload balancing requirement.

A particular challenge in most spatial analysis tasks is that a map polygon should be processed as a united structure that consists of a set of vertices. Each vertex can be considered as a tuple in database terminology. The workload balancing, in case of GIS data, could be met by partitioning the GIS map into groups of polygons where these groups should be approximately equal in terms of the total number of vertices, which is an optimization challenge. A common heuristic approach for optimisation is the use of evolutionary computation algorithms, of which the most popular is the Genetic Algorithm (GA) approach.

In this Chapter, we argue that the workload balancing challenge could be solved by applying evolutionary computation to partition large GIS maps into groups of polygons. These groups should be approximately equal in terms of the total number of vertices, and we propose an evolutionary computation based approach that consider both the nature of spatial data and the workload balancing requirement to extend the computation reliability for processing GIS complex data.

The rest of this Chapter is organized as follows: Section 8.1 introduces the proposed evolutionary based approach for balanced workload based GIS vector map partitioning. Section 8.2 describes the experiments, including the data used and the experimental setup for the evaluation of the proposed partitioning approach. Section 8.3 discusses the experimental results, while Section 8.4 concludes this Chapter.

8.1 GIS Map Partitioning

This section outlines the main steps for implementing the proposed partitioning approach, including the map index computation, and applying the genetic algorithm to the problem of workload-balanced partitions in the context of spatial data.

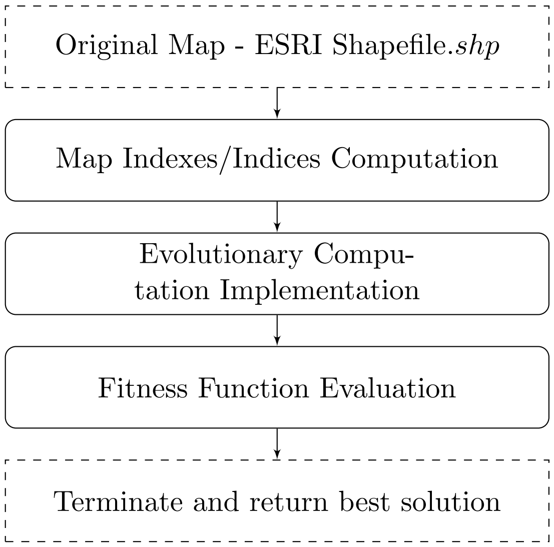

Original Map - ESRI Shapefile.*shp*
Map Indexes/Indices Computation
Evolutionary Computation Implementation
Fitness Function Evaluation
Terminate and return best solution

Figure 8.1: The proposed evolutionary-based approach.

8.1.1 Map Indexes/Indices Computation

In the proposed approach, we argue that the use of polygons' representatives (indexes) will lead to faster processing rather than the use of the whole set of polygons' vertices. For identifying the map spatial features (polygons) indexes, we used the polygon's property of bounding box. Each polygon in the GIS vector map has a defined bounding box, which identifies the boundaries of each polygon in the map; the coordinates for the bounding box are available in the shapefile (ESRI, 1998). The polygons' bounding box centers are calculated in both axes, as shown in Equation (8.1) and Equation (8.2), respectively.

$$x_c = \frac{x_{min} + x_{max}}{2} \tag{8.1}$$

$$y_c = \frac{y_{min} + y_{max}}{2} \tag{8.2}$$

where: x_c and y_c are the coordinates of polygon's center in both x and y axes respectively; x_{min} is the minimum vertex coordinate in x-axis; x_{max} is the maximum vertex coordinate in x-axis; y_{min} is the minimum vertex coordinate in y-axis; y_{max} is the maximum vertex coordinate in y-axis. x_{min}, x_{max}, y_{min} and y_{max} are each of 8-byte length (ESRI, 1998).

8.1.2 Evolutionary Computation Implementation

The Genetic Algorithm (GA) optimization technique is based on random search and has many advantages, such as performing search in complex and large spaces, and providing

Algorithm 4 Genetic Algorithm

```
Data: polygon based map; Seed Population(POP_Size); crossover rate; mutation rate
Result: near-optimal balanced map partitions
START  Initiate $POP$ ($POP_{size}$)  Evaluate $POP$  while $GEN \leq GEN_{size}$ do
    for $i \leftarrow 1$ to $POP_{size} \times crossover_{ratio}$ do
        $Parent_1$ = Tournament Selection ($POP$, $T_{size}$  $Parent_2$ = Tournament Selection
         ($POP$, $T_{size}$  ($Child_1$, $Child_1$) = crossover($Parent_1$ , $Parent_2$)
    end
    $POP_{new} \leftarrow Child_1$  $POP_{new} \leftarrow Child_2$  for $i \leftarrow 1$ to $POP_{size} \times mutation_{ratio}$ do
        $Parent_1$ = Tournament Selection ($POP$, $T_{size}$  $Child_1$ = mutate($Parent_1$)
    end
    $POP_{new} \leftarrow Child_1$  $POP \leftarrow POP_{new}$  Evaluate $POP$  $GEN++$
end
Print best evolution/solution  STOP
```

near-optimal solutions. Unlike other optimization methods, GA is more suitable to this context of discrete variables based optimization problems. As shown in Algorithm 4, GA is a heuristic search algorithm that is based on evolutionary computation, which uses a random search to solve optimization problems. GA involves five main phases: initial population, fitness computation, selection, crossover and mutation. In Algorithm 4, POP refers to the population of individuals; POP_{size} represents the number of populations; GEN_{size} represents the number of generations; T_{size} is the number of tournaments.

The initial population is randomly generated as a set of individuals, called a population, that represent solutions to the map partition problem. The parent selection is the process of selecting the fittest two pairs of individuals (i.e candidate solutions), based on standard deviation value, to be used in producing the next generation. As shown in Fig. 8.2, the crossover operator is the process of mating the selected parents to produce the next generation by identifying a crossover point. One crossover point is selected, and the coordinate values after the crossover point are copied from the second parent to the first child, and from the first parent to the second child. The mutation operator is responsible for maintaining diversity within the population and preventing premature convergence. As shown in Fig. 8.3, the selected coordinate value is inverted to a new coordinate value. The parent selection, crossover and mutation processes are carried for both horizontal lines (X–axis) and vertical lines (Y–axis), as shown in Fig 8.4.

In the proposed approach, the resulted lines are combined to form the partitioning solutions, i.e. a set of three horizontal lines and three vertical lines would lead to sixteen (4 × 4) partitions. A collection of such solutions is called a population. The tournament selection method is used for selecting the parents, which has the advantage of diversifying the individuals set (i.e. candidate solutions). The process of parent selection, crossover and

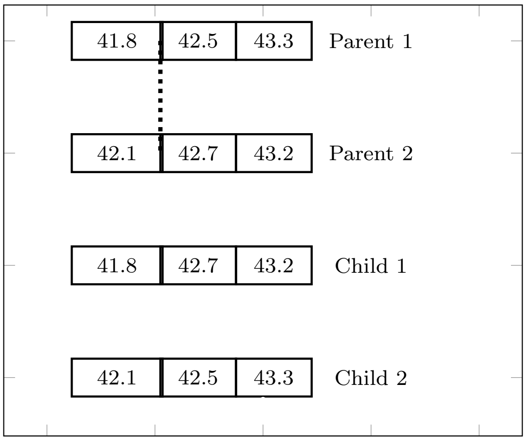


Figure 8.2: Crossover Diagram Example

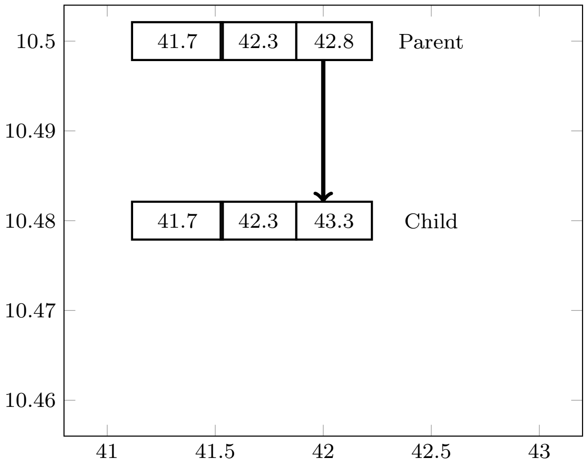


Figure 8.3: Mutation Diagram Example

mutation continues iteratively for a specified number of generations until the fitness function is satisfied.

The fitness function for the problem is the standard deviation, as illustrated in Equation (8.3). The best solution is defined by the minimum standard deviation value.

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N} (x_i - \mu)^2}{N}} \tag{8.3}$$

where: σ is the standard deviation; x_i represents each value in the population; μ is the mean value of the population; and N is the number of values in the population.

The standard deviation is calculated at the level of the set of map partitions, where each set contain a number of polygons. The smaller the value of the standard deviation, the better

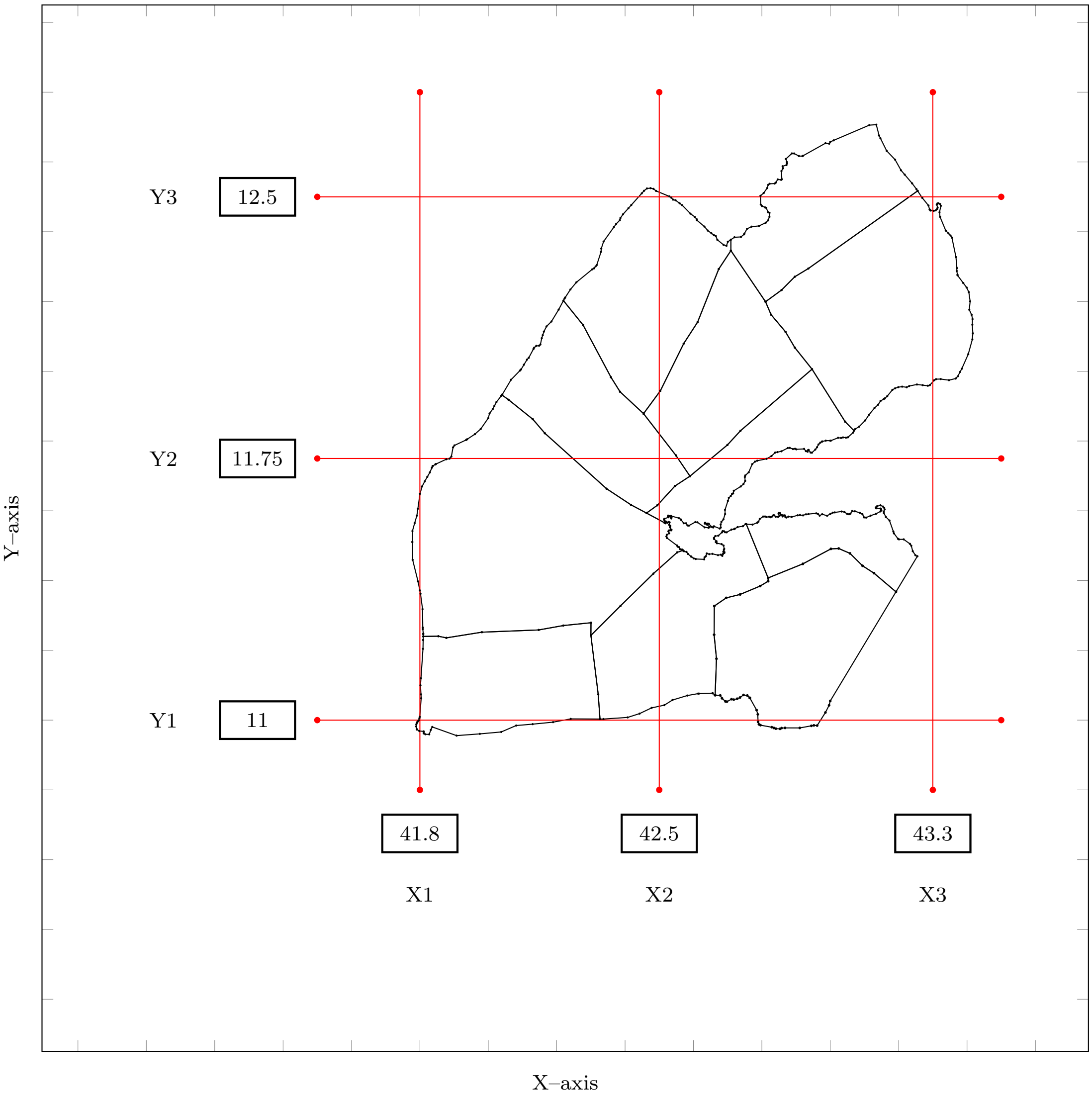


Figure 8.4: Chromosome example of the horizontal and vertical solution lines (3 × 3 lines or 4 × 4 cells)

the balance between the partitions according to the total number of vertices.

8.2 Data and Experiments

This section discusses the experimental setup for evaluating the performance and effectiveness of the proposed approach. Section 8.2.1 describes the data used in two sets of experiments; the initial set of experiments showed that some partitions had no vertices due to the uneven distribution of the data; consequently, an additional step was added to the partitioning approach to deal with these issues and a second set of experiments was carried out. Section 8.2.2 describes the initial experiments, while Section 8.2.3 describes the second set of experiments.

8.2.1 Data and Materials

We implemented the proposed approaches on a PC machine with the following specification: Windows–7 home premium 64-bits operating system, CPU 2.5GHz and RAM 4GB. The programming tasks has been implemented with Java version 8 update 171 in Netbeans integrated development environment.

To allow comparisons for maps of different sizes in terms of number of polygons and number of vertices, four datasets (of two maps each) combining high and low numbers of polygons and vertices were used, respectively:

- Dataset 1 includes maps with small number of polygons and small number of vertices.
- Dataset 2 includes maps with small number of polygons and large number of vertices.
- Dataset 3 includes maps with large number of polygons and small number of vertices.
- Dataset 4 includes maps with large number of polygons and large number of vertices.

Within each dataset, the two maps are chosen to represent opposite ratios of number of polygons to number of vertices, i.e. one map has on average a smaller number of vertices per polygon compared with the other map in the same dataset. As shown in Table 8.1, eight GIS vector maps were used to implement the proposed approach, which are illustrated in Fig. 8.5, Fig. 8.6, Fig. 8.7 and Fig. 8.8.

Table 8.1: The datasets with corresponding number of polygons, vertices and proportions of map size.

Dataset	Map	No. of Polygons	No. of Vertices	Average no. of vertices/polygon
1	Djibouti	11	676	61
	Somalia	88	3175	36
2	Guinea	56	21304	380
	Zimbabwe	81	32382	399
3	Liberia	305	10521	34
	Chad	347	19542	56
4	Burkina Faso	351	113996	324
	Ethiopia	575	261880	455

The used GIS maps are polygon-based maps that represent administrative boundaries of 8 countries in Africa, they are: Djibouti map of 11–polygons and 676–vertices (Fig.8.5a), Somalia map of 88–polygons and 3175–vertices (Fig.8.5b), Guinea map of 56–polygons and, 21304–vertices (Fig.8.6a), Zimbabwe map of 81–polygons and 32382–vertices (Fig.8.6b), Liberia map

12.8
12.6
12.4
12.2
12
11.8
11.6
11.4
11.2
11
10.8
42
42.5
43
43.5
12
10
8
6
4
2
0
−2
40
42
44
46
48
50
52

(a) Djibouti (11 polygons, 676 vertices)

(b) Somalia (88 polygons, 3175 vertices)

Figure 8.5: Data set 1.

13
12
11
10
9
8
7
−16
−15
−14
−13
−12
−11
−10
−9
−8
−7
−15
−16
−17
−18
−19
−20
−21
−22
−23
25
26
27
28
29
30
31
32
33
34

(a) Guinea (56 polygons, 21304 vertices)

(b) Zimbabwe (81 polygons, 32382 vertices)

Figure 8.6: Data set 2.

of 305–polygons and 10521–vertices (Fig.8.7a), Chad map of 347–polygons and 19542–vertices (Fig.8.7b), Burkina Faso map of 351–polygons and 113996–vertices (Fig.8.8a) and Ethiopia map of 575–polygons and 261880–vertices (Fig.8.8b). These vector maps are freely available, in ESRI shapefile format[2], from the Map Library website[3]. ESRI Shapefiles (.shp) are considered as a popular format for geographic information system applications (Abubahia and Cocea, 2017). They have several key features: supporting point, polyline and polygon geometry formats, fast shape editing, easy reading and writing, small storage space, and storing both spatial/attribute information (ESRI, 1998).

[2] http://www.esri.com

[3] http://www.maplibrary.org/library/stacks/Africa/index.htm

9
8.5
8
7.5
7
6.5
6
5.5
5
4.5
4
−11.5 −11 −10.5 −10 −9.5 −9 −8.5 −8 −7.5 −7
24
22
20
18
16
14
12
10
8
6
12 14 16 18 20 22 24 26

(a) Liberia (305 polygons, 10521 vertices)

(b) Chad (347 polygons, 19542 vertices)

Figure 8.7: Data set 3.

16
15
14
13
12
11
10
9
−6 −5 −4 −3 −2 −1 0 1 2 3
16
14
12
10
8
6
4
2
32 34 36 38 40 42 44 46 48

(a) Burkina Faso (351 polygons, 113996 vertices)

(b) Ethiopia (575 polygons, 261880 vertices)

Figure 8.8: Data set 4.

We ran an initial experiment which is described in Section 8.2.2 in which we found that some partitions had no vertices, especially for maps containing concave shapes. To address this issue, we introduce a merging step at each iteration. Consequently, we refer to the fist experiment as ”non-merging” (described in Section 8.2.2) and the second experiment as ”merging” (described in Section 8.2.3).

8.2.2 Non–merging based experiment

The experimental setup were as follows: both population size and generation number parameters were set to 10, 15 or 20, respectively. The crossover parameter was set to 0.8, the mutation parameter was set to 0.1, and the ratio parameter was set to 0.1. Moreover, each of these experiments was ran for 51 times – an odd number was chosen to have a median value as a data point rather than an average of the middle two data points. The standard deviation is used as the fitness function.

In the experiments, the number of grid cells (i.e. the number of partitions) were selected according to the number of polygons. For more clarification, the number of cells was set to 4 × 4 for maps with small number of polygons (i.e maps of Djibouti, Somalia, Guinea and Zimbabwe), while the number of cells was set to 6 × 6 for maps with a large number of polygons (i.e maps of Liberia, Chad, Burkina Faso and Ethiopia). The number of partitions can be user-defined to match the available number of processors in systems like MapReduce.

8.2.3 Merging based experiment

This approach follows the same implementation steps as in the non-merging experiment that were given above

The only difference is that before computing the fitness function, a merging procedure is applied to the partitions based on a threshold value. In this Chapter, the average number of vertices per polygon is used as threshold value. This will be advantageous in avoiding the partitions that contain no vertices, i.e the number of vertices is equal to zero.

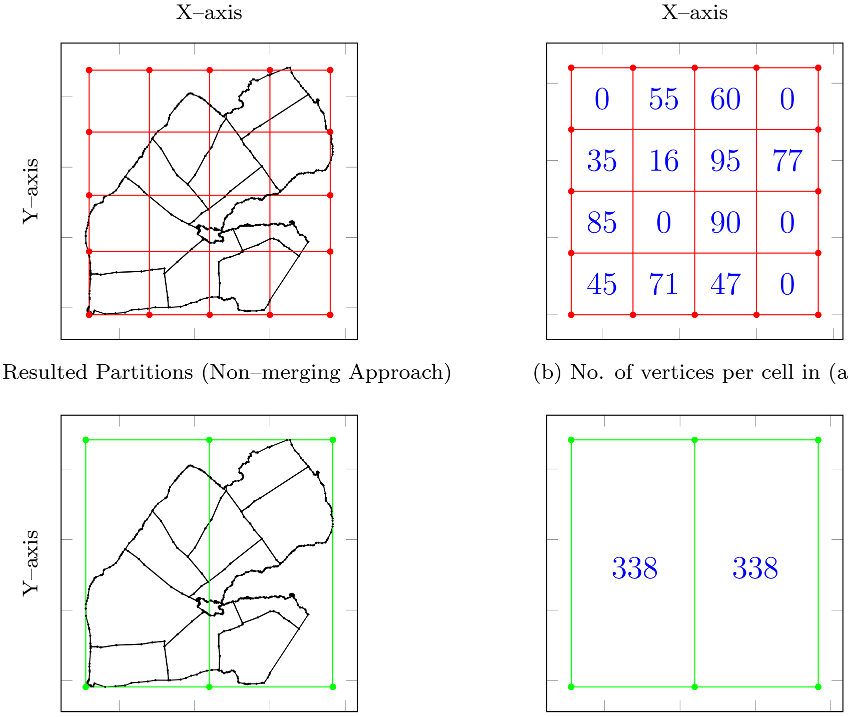
The threshold value (i.e. the number of vertices per cell) were selected according to the average number of vertices per polygon, and then set as follows: Djibouti Map (61 vertices), Somalia Map (36 vertices), Guinea Map (380 vertices), Zimbabwe Map (399 vertices), Liberia Map (34 vertices), Chad Map (56 vertices), Burkina Faso Map (324 vertices), and Ethiopia Map (455 vertices).

8.3 Results and Discussion

In this section the experimental results of the two sets of results are presented and discussed.

Fig.8.9 shows a solution example for the Djibouti map for both the non-merging and merging-based partitioning results for the experimental settings of: 51 run times, population size parameter of 20, generation number parameter of 20, crossover parameter of 0.8, mutation parameter of 0.1, and the ratio parameter of 0.1.

In the non–merging experiment, as shown in the Figure 8.9 (a), there are 16–cells that represent the resulted partitions. Two partitions, i.e. upper-left and upper-right corner cells, contain no vertices. Figure 8.9 (b) illustrates clearly how the number of vertices are distributed over the resulted partition based number of grid cells.



(a) Resulted Partitions (Non–merging Approach)

(b) No. of vertices per cell in (a)

(c) Resulted Partitions (Merging Approach)

(d) No. of vertices per cell in (c)

Figure 8.9: Djibouti Map, examples of the best solution (chromosome)

In the merging experiment, as shown in the Figure 8.9 (c), there are 2 cells that represent the resulted partitions which should contain equal or nearly-equal number of vertices. In the shown figure each cell contains 338 vertices. Figure 8.9 (d) illustrate clearly how the number of vertices are distributed over the resulted partition based number of grid cells.

To compare the results of the two sets of experiments, as well as the influence of the different values for the population size and the number of generations, we present the results in the form of box plots illustrating the range of values for the fitness function, i.e. the standard deviation.

A box plot, as shown in Figure 8.10, is a graphical shape for displaying the statistical range of values. Beginning from the top, the upper whisker represents the highest value in the range. Seventy-five percent (75%) of the resulted values fall below the upper quartile. The median marks the middle-point of the resulted standard deviation values and is shown by the line that divides the box into two parts. Half of the resulted values are greater than or equal to this value and half of the results have values lower than the median. Twenty-five percent (25%) of the resulted standard deviation values fall below the lower quartile. Finally, the lower whisker represents the smallest value in the range.

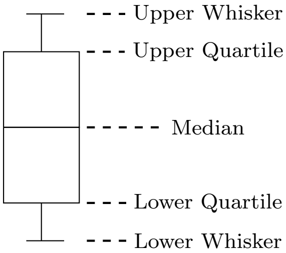
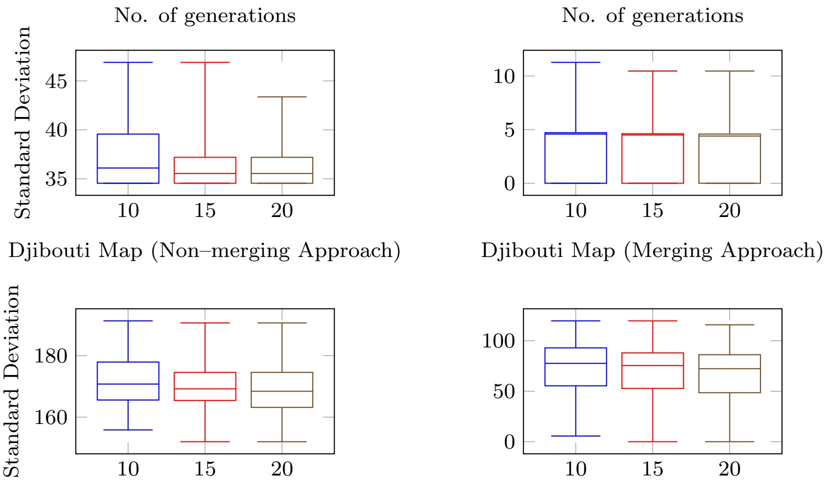


Figure 8.10: Box Plot Diagram

Figures 8.11, 8.12 and 8.13 display the results for Dataset 1 for population sizes of 10, 15 and 20, respectively. Similarly, Figures 8.14, 8.15 and 8.16 display the results for Dataset 2; Figures 8.17, 8.18, 8.19 illustrate the results for Dataset 3, and Figures 8.20, 8.21, 8.22 diaplay the results for Dataset 4.

The experimental results show that an increase in the population size leads to lower values for the standard deviation, which indicates better solutions, i.e. a more even distribution of the vertices among the partitions. Experiments with the Djibouti map, for example, show that when using the population size of 10, the standard deviation values range between 34.5 and 46.8 for the non–merging approach (top left in Figure 8.11), and between 0 and 11.2 for the merging approach (top right in Figure 8.11). For the population size of 15, the standard deviation values range between 34.5 and 40.8 for the non–merging approach (top left in Figure 8.12), and between 0 and 5.1 for to the merging approach (top right in Figure 8.12). For the population size of 20, the standard deviation values range between 34.5 and 37.1 for



Djibouti Map (Non–merging Approach)

Djibouti Map (Merging Approach)

Somalia Map (Non–merging Approach)

Somalia Map (Merging Approach)

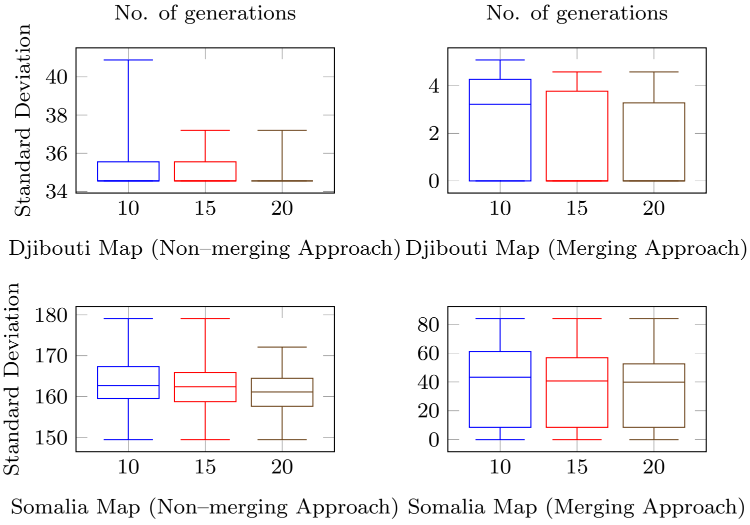
Figure 8.11: Dataset 1, comparison between non-merging approach (left column) and merging approach (right column), No. of runs=51, pop size= 10

the non–merging approach (top left in Figure 8.13), and between 0 and 4.5 for the merging approach (top left in Figure 8.13).

For all population sizes (10, 15 or 20 populations), the results show that the higher the generations number (10, 15 or 20 generations) for the reproduction process, the better the solutions produced, i.e. lower values for the standard deviation. This applies to all datasets regardless of the number of polygons or the number of vertices. Non–merging based experiments with the Liberia map (dataset 3), for example, show that in non–merging based experiments with population size of 20, when reproducing for 10 generations for the non–merging approach, the standard deviation values range between 203 and 318.6; for 15 generations, the standard deviation values are reduced to the range between 203 and 307.6; for 20 generations, the range is further reduced between 203 and 292.3; (top left in Figure 8.19.

While when experimenting with the same map, show that in merging based experiments with population size of 20, when reproducing for 10 generations for the non–merging approach, the standard deviation values range between 116.6 and 284.4; for 15 generations, the standard deviation values are reduced to the range between 116.6 and 257.7; for 20 generations, the range is further reduced between 116.6 and 245.8; (top right in Figure 8.19.

When comparing both experiments with population size of 20 and generation size of 20, it can be seen that the merging based experiment showed better standard deviation values than the non–merging based experiment. for example, for the map of Ethiopia (Figure 22, dataset 4), for the non-merging approach the standard deviation values range between 4895.7 and 9414.4 , while for the merging approach the range of the standard deviation values is



Djibouti Map (Non–merging Approach) Djibouti Map (Merging Approach)

Somalia Map (Non–merging Approach) Somalia Map (Merging Approach)

Figure 8.12: Dataset 1, comparison between non-merging approach (left column) and merging approach (right column), No. of runs=51, pop size= 15

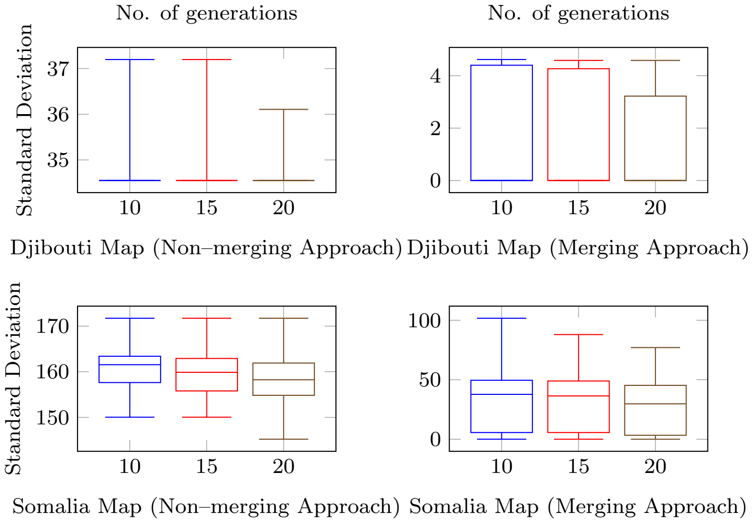


Figure 8.13: Dataset 1, comparison between non-merging approach (left column) and merging approach (right column), No. of runs=51, pop size= 20

between 3766.6 and 7757.7.

All mentioned trends – i.e. the results improve with increasing population size, the results improve with increasing numbers of generations and the results improve in the merging approach compared with the non-merging approach – can be observed for all datasets: Dataset 1 (Figures 8.11, 8.12 and 8.13), Dataset 2 (Figures 8.14, 8.15 and 8.16), Dataset 3 (Figures 8.17, 8.18, 8.19) and Dataset 4 (Figures 8.20, 8.21, 8.22).

8.4 Summary

In this Chapter, we discussed and highlighted the importance of workload balancing for GIS vector map partitioning. To address this problem, we proposed an evolutionary-based approach for GIS map partitioning using the Genetic Algorithm (GA).

The proposed approach considers the nature of spatial data to increase the computation performance for processing GIS polygon-based maps with massive number of vertices and complex shapes. Four datasets were used, where each dataset had varying degrees of size in terms of number of polygons and number of vertices. Each dataset contained two maps, which had opposite ratios of number of vertices per polygon.

A set of experiments on the four datasets were implemented to assess the influence of the evolutionary genetic algorithm parameters including the population size and the number of generations. The results showed the reliability of the proposed partitioning for workload balancing approach in real GIS map based parallel processing scenarios. The use of evolu-

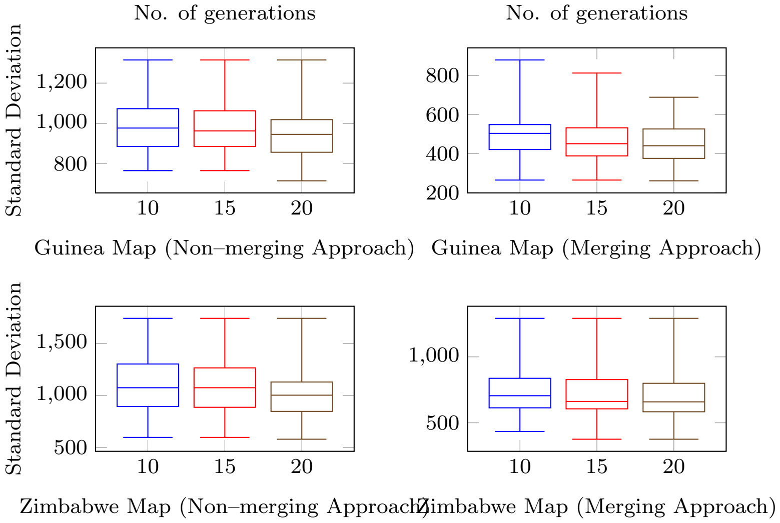


Figure 8.14: Dataset 2, comparison between non-merging approach (left column) and merging approach (right column), No. of runs=51, pop size= 10

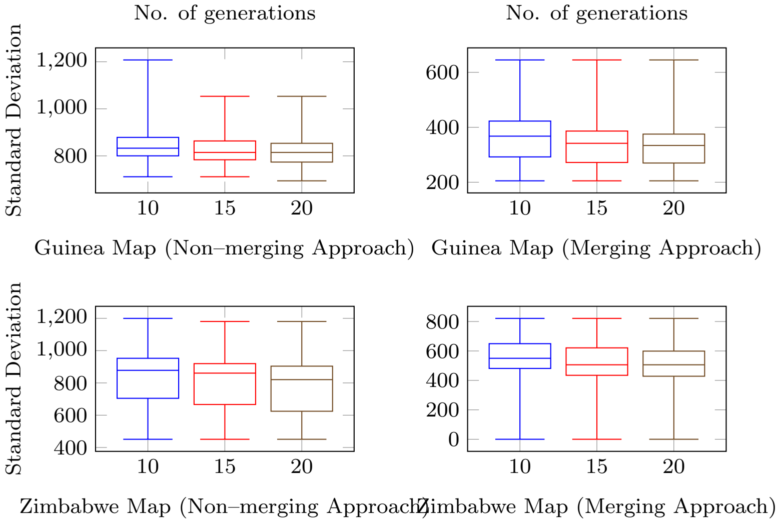


Figure 8.15: Dataset 2, comparison between non-merging approach (left column) and merging approach (right column), No. of runs=51, pop size= 15

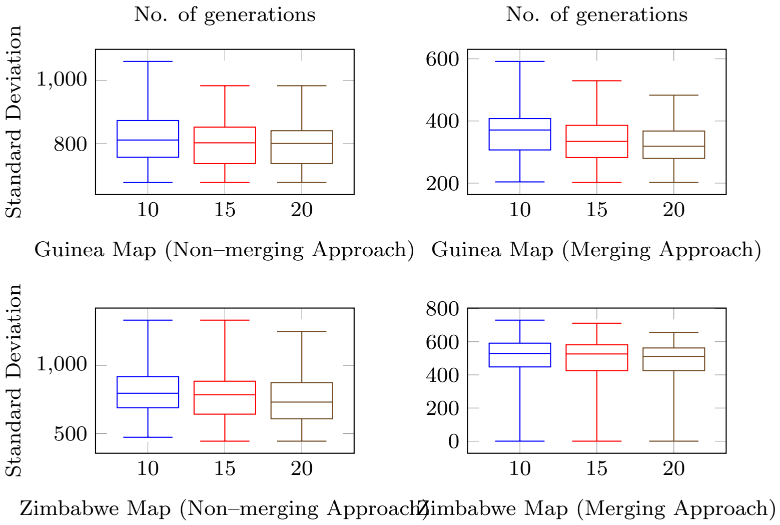


Figure 8.16: Dataset 2, comparison between non-merging approach (left column) and merging approach (right column), No. of runs=51, pop size= 20

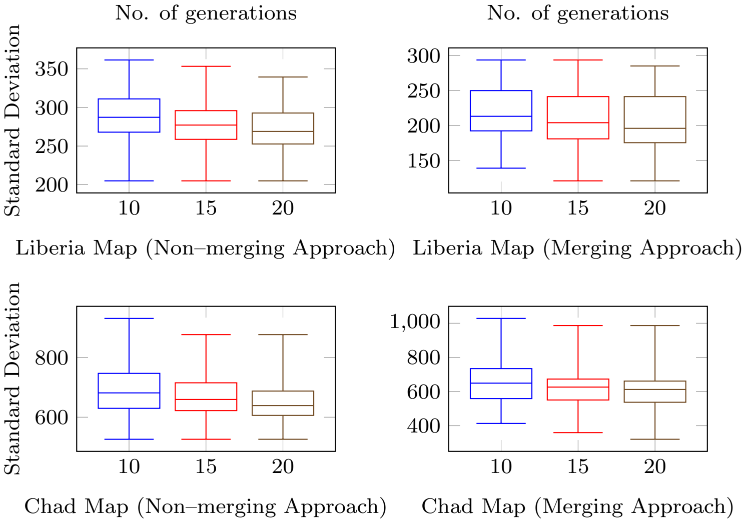
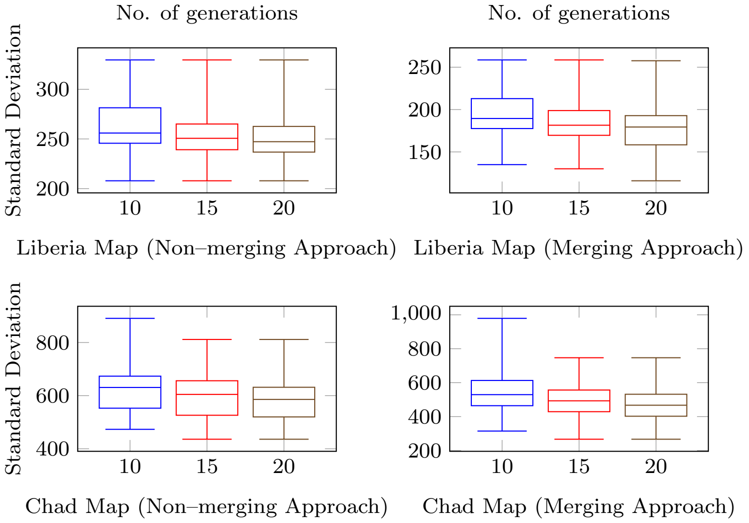


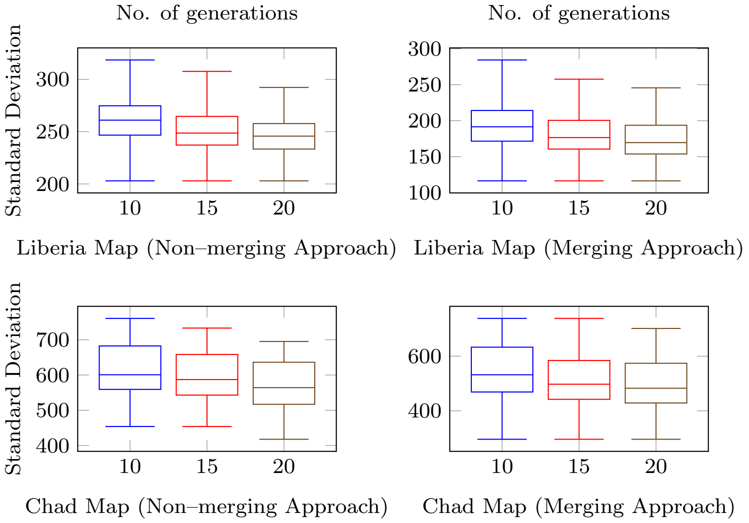
Figure 8.17: Dataset 3, comparison between non-merging approach (left column) and merging approach (right column), No. of runs=51, pop size= 10



Liberia Map (Non–merging Approach) Liberia Map (Merging Approach)

Chad Map (Non–merging Approach) Chad Map (Merging Approach)

Figure 8.18: Dataset 3, comparison between non-merging approach (left column) and merging approach (right column), No. of runs=51, pop size= 15



Liberia Map (Non–merging Approach) Liberia Map (Merging Approach)

Chad Map (Non–merging Approach) Chad Map (Merging Approach)

Figure 8.19: Dataset 3, comparison between non-merging approach (left column) and merging approach (right column), No. of runs=51, pop size= 20

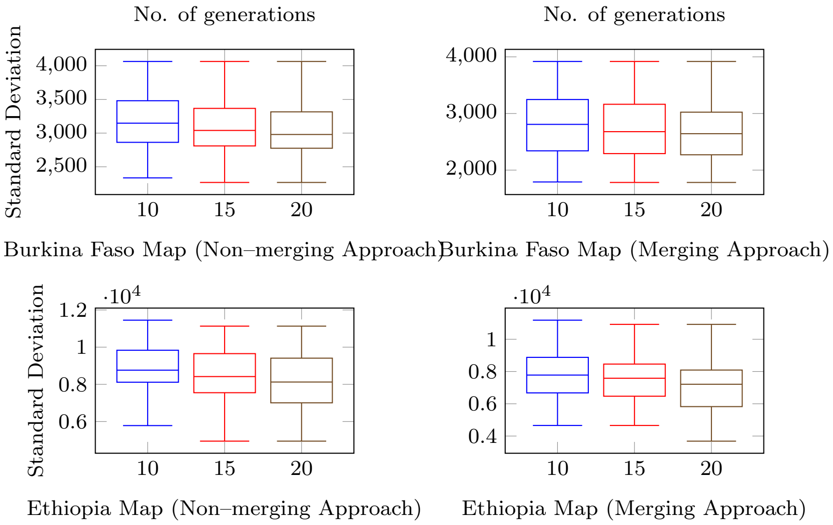


Figure 8.20: Dataset 4, comparison between non-merging approach (left column) and merging approach (right column), No. of runs=51, pop size= 10

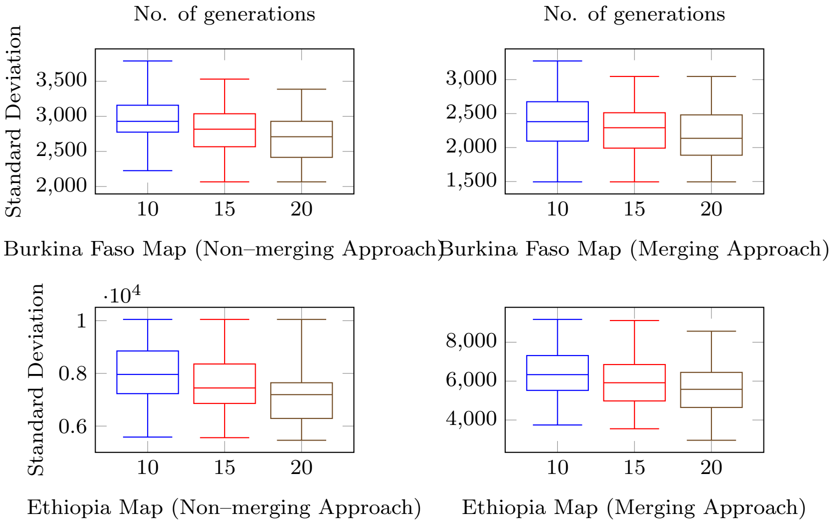
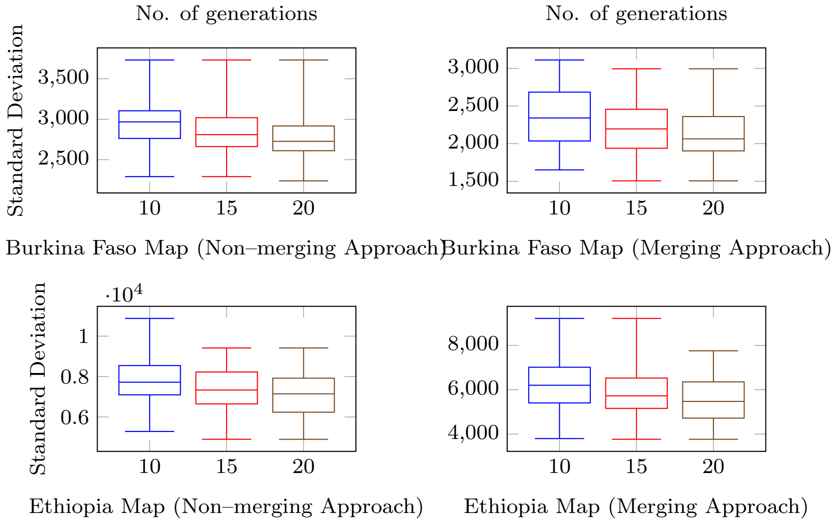


Figure 8.21: Dataset 4, comparison between non-merging approach (left column) and merging approach (right column), No. of runs=51, pop size= 15



Burkina Faso Map (Non–merging Approach) Burkina Faso Map (Merging Approach)

Ethiopia Map (Non–merging Approach) Ethiopia Map (Merging Approach)

Figure 8.22: Dataset 4, comparison between non-merging approach (left column) and merging approach (right column), No. of runs=51, pop size= 20

tionary computation shows a promising potential in partitioning GIS maps into balanced set of adjacent polygons based on the number of vertices.

Chapter 9

Conclusion

This Chapter summarise the main contributions of this thesis, and discusses the important directions of future work.

9.1 Contribution to Knowledge

In conclusion, this thesis contributes to the current knowledge, as follows:

1. The use of k-medoids clustering technique in the process of identifying the location for embedding the watermark has an influence on the security of the watermarked map measured in terms of capacity, fidelity, computational time and robustness. The experimental results show that both k-medoids and kmeans clustering approaches result in high fidelity, while the k-medoids based clustering approach achieves a more balanced trade-off between capacity and fidelity, as well as better computational efficiency due to the k-medoids characteristics.

2. The use of k-medoids clustering technique in combination with the bounding box centers has a significant implication on the trade-off between the fidelity and the capacity metrics, and resulted in higher fidelity as capacity increased. In addition to the improvement of the trade-off between fidelity and capacity, the use of bounding box centers adds more robustness to the simplification and interpolation attacks due to their independence from the number of vertices in a polygon. The experimental results show that both k-medoids and kmeans clustering approaches result in high fidelity, while the k-medoids based clustering approach achieves a more balanced trade-off between capacity and fidelity, as well as better computational efficiency due to the k-medoids characteristics.

3. Regardless of the clustering technique type, merging the use of clustering techniques with the bounding box property of GIS vector map for locating the watermark bits into polygons' vertices has a significant implication on protecting the GIS vector map copyright, especially in terms of addressing the vulnerability to simplification and interpolation attacks, while preserving a good trade-off between fidelity and capacity. The experimental results show that the advantages of using the bounding box property are maintained even with k-means clustering approach, and argue that they would hold regardless of the method used for identifying the watermark embedding locations in the map.

4. The use of partitioning clustering techniques in combination with the vector polygon based map topology rules leads to defining a metric that allow comparisons between watermarked maps of different sizes and of different watermark sizes, and, thus, can be used to assess the quality of watermarked vector maps. The experimental results indicate that the metrics allow comparisons between watermarked maps of different sizes and of different watermark sizes, and, thus, can be used to asses the quality of watermarked vector maps.

5. The use of evolutionary computation technique in combination with GIS map properties advances the implementation of partitioning clustering approach in the context of GIS vector maps. The experimental results show the capability of the proposed clustering approach in addressing the issue of workload balancing in the context of GIS vector map data.

9.2 Future Research Directions

For future research directions, research and experiments will be carried out on computing a fixed set of initial representatives for the proposed k-medoids-based watermarking approach to achieve more predictability and efficiency, to eliminate the randomness involved in the initial selection of the centers involved in the typical PAM-based k-medoids method in Chapter 4. Also, we will experiment with other clustering approaches proposed in the literature, such as grid-based and density-based approaches, to explore the capability of different cluster analysis tools in the research context of GIS vector map data copyright protection.

Moreover, research and experiments will be carried out on addressing the problem of the randomness in the map polygon indexes associated with odd-even coding to further understand the behavior of the proposed metric in Chapter 7 with extreme cases. Also, the possibility of introducing different weights for the different topological aspects will be investigated.

In addition, further research and experiments will be carried out on comparing the proportion-based balancing (e.g. 25%, 33% or 50%) of total number of polygons in comparison with the total number of vertices, and investigating the impact of the vector map shape (concave and convex shapes) on the partitioning experiments in Chapter 8. Also, the possibility of introducing different vertices relevant weights for the different partitioning aspects will be investigated.

Bibliography

Abbas, T., Jawad, M., and Sudirman, S. (2013). Robust watermarking of digital vector maps for copyright protection. In *14th Annual PostGraduate Symposium on The Convergence of Telecommunications, Networking and Broadcasting*, 978-1-902560-27-4, Liverpool.

Abbas, T. A. and Jawad, M. J. (2013a). Digital vector map watermarking: Applications, techniques and attacks. *Oriental Journal of Computer Science and Technology*, 6(3):333–339.

Abbas, T. A. and Jawad, M. J. (2013b). Proposed an intelligent watermarking in GIS environment. *Journal of Earth Science Research*, 1(1):1–5.

Abubahia, A. and Cocea, M. (2014). Partition clustering for GIS map data protection. In *IEEE 26th International Conference on Tools with Artificial Intelligence*, pages 830–837.

Abubahia, A. and Cocea, M. (2015a). A clustering approach for protecting GIS vector data. In *The 27th International Conference on Advanced Information Systems Engineering*, pages 133–147.

Abubahia, A. and Cocea, M. (2015b). Exploiting vector map properties for GIS data copyright protection. In *The 27th International Conference on Tools with Artificial Intelligence*, pages 575 – 581.

Abubahia, A. and Cocea, M. (2017). Advancements in gis map copyright protection schemes - a critical review. *Multimedia Tools and Applications*, 76(10):12205–12231.

Abubahia, A. and Cocea, M. (2018). Evaluating the topological quality of watermarked vector maps. *Applied Soft Computing*, 71:849 – 860.

Aji, A., Wang, F., Vo, H., Lee, R., Liu, Q., Zhang, X., and Saltz, J. (2013). Hadoop-GIS: A high performance spatial data warehousing system over mapreduce. In *The 39th International Conference on Very Large Data Bases*, volume 6, pages 1009 – 1020.

Angelaccio, M., Buttarazzi, B., Basili, A., and Liguori, W. (2012). Using geo-business intelligence to improve quality of life. In *2012 IEEE First AESS European Conference on Satellite Telecommunications (ESTEL)*, pages 1–6.

Araujo Neto, A. C., Coelho da Silva, T. L., de Farias, V. A. E., Macêdo, J. A. F., and de Castro Machado, J. (2015). G2P: A partitioning approach for processing dbscan with mapreduce. In *Web and Wireless Geographical Information Systems*, pages 191–202, Cham. Springer International Publishing.

AutoCAD (2007). DXF reference. Technical report, Autodesk.

Aybet, J., Al-Saedy, H., and Farmer, M. (2009). Watermarking spatial data in geographic information systems. In Jahankhani, H., Hessami, A., and Hsu, F., editors, *Global Security, Safety, and Sustainability*, volume 45 of *Communications in Computer and Information Science*, pages 18–26. Springer Berlin Heidelberg.

Bação, F., Lobo, V., and Painho, M. (2005). Applying genetic algorithms to zone design. *Soft Computing*, 9(5):341–348.

Bainbridge, D. I. (2014). *Information Technology and Intellectual Property Law*. Bloomsbury Professional, 6 edition.

Baiyan, W., Wei, W., and Dandan, M. (2008a). 2D vector map watermarking based on spatial relations. *Proc. SPIE*, 7285:728532–728537.

Baiyan, W., Wei, W., and Dandan, M. (2008b). 2D vector map watermarking based on spatial relations. *SPIE*, 7285:32–37.

Bansal, M. and Upadhyaya, A. (2018). Three-level GIS data security: Conjointly cryptography and digital watermarking. In Bokhari, M. U., Agrawal, N., and Saini, D., editors, *Cyber Security*, pages 241–247, Singapore. Springer Singapore.

Barni, M. and Bartolini, F. (2004). *Watermarking Systems Engineering: Enabling Digital Assets Security and Other Applications*. Signal Processing and Communications. Taylor & Francis.

Barua, H. B., kumar Das, D., and Sarmah, S. (2012). A density based clustering technique for large spatial data using polygon approach. *Journal of Computer Engineering*, 3(6):1–9.

Bazin, C., Le Bars, J.-M., and Madelaine, J. (2007). A blind, fast and robust method for geographical data watermarking. In *2nd ACM Symposium on Information, Computer and Communications Security*, pages 265–272.

Bhanuchandar, P., Prasad, M., and Srinivas, K. (2013). A survey on various watermarking methods for GIS vector data. *International Journal of Computer and Electronics Research*, 2(3).

Bird, S., Bellman, C., and Van Schyndel, R. (2009). A shape-based vector watermark for digital mapping. In *The Conference on Digital Image Computing: Techniques and Applications*, pages 454–461.

Bisher, M., Wytzisk, A., and Morales, J. (2007). Geodrm: Towards digital management of intellectual property rights for spatial data infrastructures. *Research and Theory in Advancing Spatial Data Infrastructure Concepts*, pages 245 – 260.

Bong-Joo, J., Suk-Hwan, L., Sanghun, L., and Ki-Ryong, K. (2014). Progressive vector compression for high-accuracy vector map data. *International journal of Geographic Information Science*, 28(4):763–779.

Bonham-Carter, G. F. (2014). *Geographic Information Systems for Geoscientists: modelling with GIS*. Pergamon.

Boobalan, M. P., Lopez, D., and Gao, X. (2016). Graph clustering using k-neighbourhood attribute structural similarity. *Applied Soft Computing*, 47:216 – 223.

Burrough, P., McDonnell, R., and Lloyd, C. (2013). *Principles of Geographical Information Systems*. Oxford University Press.

Burrough, P. A., McDonnell, R., McDonnell, R. A., and Lloyd, C. D. (2015). *Principles of Geographical Information Systems*. Oxford University Press.

Busch, G. (2012). GIS-based tools for regional assessments and planning processes regarding potential environmental effects of poplar src. *BioEnergy Research*, 5(3):584–605.

Calagna, M. and Mancini, L. (2007). Information hiding for spatial and geographical data. In Belussi, A., Catania, B., Clementini, E., and Ferrari, E., editors, *Spatial Data on the Web*, pages 235–258. Springer Berlin.

Camelli, F., Lien, J.-M., Shen, D., Wong, D. W., Rice, M., Löhner, R., and Yang, C. (2012). Generating seamless surfaces for transport and dispersion modeling in GIS. *GeoInformatica*, 16(2):307–327.

Cao, J., Li, A., and Lv, G. (2010a). Study on multiple watermarking scheme for GIS vector data. In *18th International Conference on Geoinformatics*, pages 1–6.

Cao, L., Men, C., and Gao, Y. (2013a). A recursive embedding algorithm towards lossless 2D vector map watermarking. *Digital Signal Processing*, 23(3):912–918.

Cao, L., Men, C., and Ji, R. (2013b). Nonlinear scrambling-based reversible watermarking for 2D-vector maps. *The Visual Computer*, 29(3):231–237.

Cao, L., Men, C., and Ji, R. (2014). High-capacity reversible watermarking scheme of 2D-vector data. *Signal, Image and Video Processing*, pages 1–8.

Cao, L., Men, C., and Ji, R. (2015). High-capacity reversible watermarking scheme of 2D-vector data. *Signal, Image and Video Processing*, 9:1387–1394.

Cao, L., Men, C., and Li, X. (2010b). Iterative embedding-based reversible watermarking for 2d-vector maps. In *17th IEEE International Conference on Image Processing*, pages 3685–3688.

Cao, L., Men, C., and Sun, J. (2011). A double zero-watermarking algorithm for 2d vector maps. *Journal of Harbin Engineering University*, 3.

Cao, Z., Wang, S., Forestier, G., Puissant, A., and Eick, C. F. (2013c). Analyzing the composition of cities using spatial clustering. In *Proceedings of the 2Nd ACM SIGKDD International Workshop on Urban Computing*, pages 141–148.

Chang, H.-H., Chen, T., and Kan, K.-S. (2003). Watermarking 2D/3D graphics for copyright protection. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 4, pages 720–723.

Chang, K.-T. (2012). *Introduction to Geographic Information Systems.* McGraw-Hill.

Che, S. and Deng, S.-j. (2008). Watermarking arithmetic of 2D vector maps based on two-tier grids. *Hydrographic Surveying and Charting*, 1.

Chen, Y.-F., Sun, C., Shen, J.-Q., Huang, Z.-D., and Chen, L.-P. (2016). Semi-fragile watermarking for automatic detection of engineering drawing modifications in collaborative design. *Computer-Aided Design and Applications*, 13(5):571–586.

Cheng, F.-j., Yin, H., Zhang, X.-p., and Zhang, D.-x. (2010). A digital watermarking algorithm for vector map. In *International Conference on Challenges in Environmental Science and Computer Engineering*, volume 2, pages 101–103.

Choi, J., Lee, D., and Jung, H. (2014). *Knowledge Discovery and Integration: A Case Study of Housing Planning Support System.* Springer Berlin Heidelberg.

Chuanjian, W., Bin, L., Qingzhan, Z., Zuqi, Q., Yuwei, P., and Liang, Y. (2009). A 2D vector map watermarking algorithm resistant to simplication attack. *SPIE*, 7651:4–8.

Ciptasari, R. and Sakurai, K. (2013). *Multimedia Information Hiding Technologies and Methodologies for Controlling Data*, chapter Multimedia Copyright Protection Scheme Based on the Direct Feature-Based Method, pages 412–439. IGI Global.

Cox, I., Miller, M., Bloom, J., Fridrich, J., and Kalker, T. (2007). *Digital Watermarking and Steganography*. The Morgan Kaufmann Series in Multimedia Information and Systems. Elsevier Science.

Croitoru, A., Crooks, A., Radzikowski, J., and Stefanidis, A. (2013). Geosocial gauge: A system prototype for knowledge discovery from social media. *International Journal of Geographical Information Science*, 27(12):2483–2508.

Cui, H., Zhu, C., Ren, N., and Wang, D. (2013). A multiple watermarking algorithm for vector geographic data based on watermarking information segmentation. *Journal of Geomatics Science and Technology*, 2.

Da, Q., Sun, J., Zhang, L., Kou, L., Wang, W., Han, Q., and Zhou, R. (2018). A novel hybrid information security scheme for 2d vector map. *Mobile Networks and Applications*, 23(4):734–742.

Dakroury, Y., El-ghafar, I. A., and Tammam, A. (2010). Protecting GIS data using cryptography and digital watermarking. *International Journal of Computer Science and Network Security*, 10(1):75 – 84.

Davydov, A., Kovalev, A., and Izyurov, K. (2011). Distortion measure of watermarking 2D vector maps in the mesh-spectral domain. In *17th International Conference on Digital Signal Processing*, pages 1–6.

Dean, J. and Ghemawat, S. (2004). Mapreduce: Simplified data processing on large clusters. In *the 6th conference on Symposium on Opearting Systems Design & Implementation*, pages 1–13. Google, Inc.

Deng, L.-p. and Xiao, H. (2010). A lossless watermarking algorithm for vector graphics based on wavelet transform. *Computer Knowledge and Technology*, 4.

Dollner, J. (2005). Geospatial digital rights management in geovisualization. *The Cartographic Journal*, 42(1):27–34.

Doncel, V., Nikolaidis, N., and Pitas, I. (2007). An optimal detector structure for the fourier descriptors domain watermarking of 2D vector graphics. *IEEE Transactions on Visualization and Computer Graphics*, 13(5):851–863.

Du, Q. and Peng, F. (2008). A zero-watermark algorithm with real-mean for 2d engineering graphic. In *International Symposium on Electronic Commerce and Security*, pages 890–893.

Eldawy, A., Alarabi, L., and Mokbel, M. F. (2015). Spatial partitioning techniques in SpatialHadoop. *Proc. VLDB Endow.*, 8(12):1602–1605.

Eldawy, A. and Mokbel, M. F. (2015). Spatialhadoop: A mapreduce framework for spatial data. In *The 31st International Conference on Data Engineering*, pages 1352–1363.

Elhami, S., Saalfeld, A., and Kang, H. (2001). Using shape analyses for placement of polygon labels. In *Esri International User Conference*, San Diego, CA.

Ericsson, A. and WCDMA, R. (2011). Clustering and polygon merging algorithms for fingerprinting positioning in LTE. In *5th International Conference on Signal Processing and Communication Systems (ICSPCS)*.

ESRI (1998). ESRI shapefile technical description. Technical report, Environmental Systems Research Institute, Inc., 380 New York Street, Redlands, CA 92373-8100 USA.

Fei, P., Li, C., and Min, L. (2013). A reversible watermark scheme for 2d vector map based on reversible contrast mapping. *Security and Communication Networks*, 6(9):1117–1125.

Fenwick, T. and Locks, I. (2010). *Copyright in The Digital Age: Industry IssIss and Impacts.* Wildy, Simmonds and Hill Publishing, 1 edition.

Fu, H., Zhu, C., Yuan, J., and Xu, H. (2013). A new watermarking algorithm for geo-spatial data. In Lu, W., Cai, G., Liu, W., and Xing, W., editors, *the International Conference on Information Technology and Software Engineering*, volume 210 of *LNEE*, pages 945–951. Springer Berlin.

Fu, Y. X., Zhao, W. Z., and Ma, H. F. (2011). Research on parallel dbscan algorithm design based on mapreduce. In *Advanced Measurement and Test*, volume 301 of *Advanced Materials Research*, pages 1133–1138. Trans Tech Publications.

Gaata, M. (2018). Robust watermarking scheme for GIS vector maps. *Ibn Al-Haitham J. for Pure & Appl. Sci.*

Geng, M., Yu, P., Han, H., Teng, Z., Hu, J., and Gao, Y. (2012). Reversible watermarking based on invariant sum value for 2D vector maps. In *3rd IEEE International Conference on Network Infrastructure and Digital Content*, pages 521–525.

Giannoula, A., Nikolaidis, N., and Pitas, I. (2002). Watermarking of sets of polygonal lines using fusion techniques. In *IEEE International Conference on Multimedia and Expo*, volume 2, pages 549–552.

Gu, X., Angelov, P. P., and Príncipe, J. C. (2018). A method for autonomous data partitioning. *Information Sciences*, 460-461:65 – 82.

Gufler, B., Augsten, N., Reiser, A., and Kemper, A. (2012). *The Partition Cost Model for Load Balancing in MapReduce*, pages 371–387. Springer New York, New York, NY.

Guo, R.-s. and Peng, F. (2010). Semi-fragile watermarking algorithm for 2D engineering graphics based on improved odd-even quantization. *Journal of Chinese Computer Systems*, 10.

Hamzaoui, R. and Saupe, D. (2006). Fractal image compression. In Barni, M., editor, *Document and Image Compression*, chapter 6, pages 145–177. CRC.

Han, J., Kamber, M., and Pei, J. (2012). *Data Mining: Concept and Techniques*. Morgan Kaufmann, Waltham, 3rd edition.

Han, J., Lee, J.-G., and Kamber, M. (2009). *An Overview of Clustering Methods in Geographic Data Analysis*, chapter 7, pages 150–187. Taylor & Francis Group, LLC, 2 edition.

Haowen, Y. (2011a). Watermarking algorithm for vector point clusters. In *7th International Conference on Wireless Communications, Networking and Mobile Computing*, pages 1–4.

Haowen, Y. (2011b). Watermarking algorithm for vector point clusters. In *7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM)*, pages 1–4.

Hassan, R. and Mohammed, M. (2017). Information hiding using geographic information system (GIS) vector file 2017. *Engineering and Technology Journal*.

He, X., Zhu, C., and Wang, Q. (2009). The blind watermarking model of the vector geospatial data based on DFT of QIM. In *IEEE International Conference on Network Infrastructure and Digital Content*, pages 1039–1044.

Horness, E., Nikolaidis, N., and Pitas, I. (2007). Blind city maps watermarking utilizing road width information. In *15th European Signal Processing Conference*, pages 2291 – 2295, Poland.

Hou, H., Li, J., Qi, J., and Guo, J. (2014). A blind watermarking for 2D-vector engineering graphics. *Information Technology Journal*, pages 869–873.

Hou, X., Min, L., and Yang, H. (2018). A reversible watermarking scheme for vector maps based on multilevel histogram modification. *Symmetry*, 10(9).

Hu, J. and Geng, M. (2013). A reversible watermarking algorithm for 2d vector maps. In *2nd International Symposium on Instrumentation and Measurement, Sensor Network and Automation*, pages 1101–1104.

Huan, Y. and Yufeng, G. (2009). A digital watermarking algorithm for cad two-dimensional graphics. *Electronic Measurement Technology*, 4.

Huang, L., Zhou, W., Jiang, R., and Li, A. (2010). Data quality inspection of watermarked GIS vector map. In *Geoinformatics, 2010 18th International Conference on*, pages 1–5.

Huang, X.-s. and Gu, J.-w. (2006). A non-blind detection watermarking algorithm for 2-dimensional engineering drawings. *Journal of Engineering Graphics*, 4.

Huber, S., Kwitt, R., Meerwald, P., Held, M., and Uhl, A. (2010). Watermarking of 2D vector graphics with distortion constraint. In *IEEE International Conference on Multimedia and Expo*, pages 480–485.

Huo, X.-J., Lee, S.-H., Kwon, S.-G., Moon, K.-S., and Kwon, K.-R. (2011a). A watermarking scheme for shapefile-based GIS digital map using polyline perimeter distribution. *Journal of Korea Multimeadia Society*, 14(5):595 – 606.

Huo, X.-J., Moon, K.-S., Lee, S.-H., Seung, T.-Y., and Kwon, S.-G. (2011b). Protecting GIS vector map using the k-means clustering algorithm and odd-even coding. In *17th Korea-Japan Joint Workshop on Frontiers of Computer Vision*, pages 1–5. IEEE.

Huo, X.-J., Moon, K.-S., Lee, S.-H., Seung, T.-Y., and Kwon, S.-G. (2011c). Protecting GIS vector map using the k-means clustering algorithm and odd-even coding. In *17th Korea-Japan Joint Workshop on Frontiers of Computer Vision*, pages 1–5.

Huo, X.-J., Seung, T.-Y., Jang, B.-J., Lee, S.-H., and Kwon, S.-G. (2010). A watermarking scheme using polyline and polygon characteristic of shapefile. In *3rd International Conference on Intelligent Networks and Intelligent Systems*, pages 649–652.

Huynh-Thu, Q. and Ghanbari, M. (2008). Scope of validity of PSNR in image/video quality assessment. *Electronics Letters*, 44(13):800–801.

Im, D.-H., Lee, H.-Y., Ryu, S.-J., and Lee, H.-K. (2008). Vector watermarking robust to both global and local geometrical distortions. *IEEE Signal Processing Letters*, 15:789–792.

Jang, B.-J., Lee, S.-H., Lee, E.-J., Lim, S., and Kwon, K.-R. (2016). A crypto-marking method for secure vector map. *Multimedia Tools and Applications*, pages 1–34.

Jasim, M. and Asadi, T. A. (2012). New graph mining algorithm for vector GIS systems. In *8th International Conference on Computing Technology and Information Management (NCM and ICNIT)*, volume 1, pages 335–338.

Ji, G. and Zhang, L. (2009). A spatial polygon objects clustering algorithm based on topological relations for GML data. In *2009 International Conference on Information Engineering and Computer Science*, pages 1–4.

Jia, P., Chen, Y., Ma, J., and Zhu, D. (2006). Digital watermark-based security technology for geo-spatial graphics data. *Chinese Geographical Science*, 16(3):276–281.

Jian-Guo, S., Guo-Yin, Z., Ai-Hong, Y., and Jun-Peng, W. (2014). A reversible digital watermarking algorithm for vector maps. *International Journal of Network Security*, 16(1):40–45.

Jiang, K., Zhu, K. Q., Huang, Y., and Ma, X. (2013). Watermarking road maps against crop and merge attacks. In *1st ACM Workshop on Information Hiding and Multimedia Security*, IHMMSec '13, pages 221–230, New York, NY, USA.

Jianguo, S., Chonghui, Z., and Di, G. (2014). Lossless digital watermarking scheme for image maps. *Communications, China*, 11(8):125–130.

Jianguo, S., Liang, K., and Songzhu, X. (2013a). Research of lossless digital watermarking technology. *Applied Mechanics and Materials*, 333:1219–1223.

Jianguo, S., Liang, K., and Songzhu, X. (2013b). Research of lossless digital watermarking technology. *Applied Mechanics and Materials*, 333:1219–1223.

Jianguo, S., Songzhu, X., and Guoyin, Z. (2012). Vector map watermarking evaluation based on certainty factor. *Journal of Theoretical and Applied Information Technology*, 46(1):67 – 72.

Joshi, D., Samal, A., and Soh, L.-K. (2009a). A dissimilarity function for clustering geospatial polygons. In *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 384–387, New York, NY, USA. ACM.

Joshi, D., Samal, A. K., and Soh, L. K. (2009b). Density-based clustering of polygons. In *2009 IEEE Symposium on Computational Intelligence and Data Mining*, pages 171–178.

Joshi, D., Soh, L. K., and Samal, A. (2009c). Redistricting using heuristic-based polygonal clustering. In *2009 Ninth IEEE International Conference on Data Mining*, pages 830–835.

Joshi, D., Soh, L.-K., and Samal, A. (2012). Redistricting using constrained polygonal clustering. *IEEE Transactions on Knowledge and Data Engineering*, 24(11):2065 – 2079.

Junfeng, Z. and Bing, X. (2011). Research on digital watermarking algorithms for 2D graphics. In *IEEE 3rd International Conference on Communication Software and Networks*, pages 179–183.

Kan, Y.-h., Yang, C.-s., Cui, H.-c., Wang, Y.-y., and Liu, R. (2010). High-fidelity digital watermarking algorithm for vector geospatial data. *Journal of Geomatics Science and Technology*, 2.

Kang, H., Kim, K., and Choi, J. (2001a). A vector watermarking using the generalized square mask. In *International Conference on Information Technology: Coding and Computing*, pages 234–236.

Kang, H., Kim, K., and Choi, J. (2002). Map data watermarking using generalised square mask. *Electronics Letters*, 38:1645–1646.

Kang, H. I., Kim, K. I., and Choi, J. U. (2001b). A map data watermarking using the generalized square mask. In *IEEE International Symposium on Industrial Electronics*, volume 3, pages 1956–1958.

Kang, J.-j. and Zhang, H.-l. (2009). Blind watermarking algorithm for 2d engineering graphics based on fractional fourier transform. *Journal of Computer Applications*, 6.

Katzenbeisser, S. and Petitcolas, F. (2000). *Information Hiding Techniques for Steganography and Digital Watermarking*. Computer Security Series. Artech House.

Kennedy, M. (2013). *Introducing Geographic Information Systems with ArcGIS*. John Wiley and Sons.

Kim, J. (2010a). Robust vector digital watermarking using angles and a random table. *Advances in Information Sciences and Service Sciences*, 2(4):79–90.

Kim, J. (2010b). Vector map digital watermarking using angles. In *6th International Conference on Networked Computing and Advanced Information Management*, pages 417–423.

Kim, J. and Hong, S. (2009). Development of digital watermarking technology to protect cadastral map information. In *2nd International Conference on Interaction Sciences: Information Technology, Culture and Human*, pages 923–929.

Kim, J., Won, S., Zeng, W., and Park, S. (2011). Copyright protection of vector map using digital watermarking in the spatial domain. In *7th International Conference on Digital Content, Multimedia Technology and its Applications*, pages 154–159.

Kingston, R. (2007). Public participation in local policy decision-making: The role of web-based mapping. *The Cartographic Journal*, 44(2):138–144.

Kisore, N. R. and Koteswaraiah, C. B. (2017). Improving atm coverage area using density based clustering algorithm and voronoi diagrams. *Information Sciences*, 376:1 – 20.

Kitamura, I., Kanai, S., and Kishinami, T. (2001). Copyright protection of vector map using digital watermarking method based on discrete fourier transform. In *International Symposium on Geoscience and Remote Sensing*, volume 3, pages 1191–1193.

Kolatch, E. (2001). Clustering algorithms for spatial databases: A survey.

Lafaye, J., Béguec, J., Gross-Amblard, D., and Ruas, A. (2007a). Geographical database watermarking by polygon elongation. Technical report, HAL.

Lafaye, J., Béguec, J., Gross-Amblard, D., and Ruas, A. (2007b). Invisible graffiti on your buildings: Blind and squaring-proof watermarking of geographical databases. In Papadias, D., Zhang, D., and Kollios, G., editors, *Advances in Spatial and Temporal Databases*, volume 4605 of *LNCS*, pages 312–329. Springer Berlin.

Lafaye, J., Béguec, J., Gross-Amblard, D., and Ruas, A. (2012). Blind and squaring-resistant watermarking of vectorial building layers. *GeoInformatica*, 16(2):245–279.

Lan, H. and Peng, Y. (2017). Reversible fragile watermarking for fine-grained tamper localization in spatial data. In *ADC*.

Lee, S.-H., Hwang, W.-J., and Kwon, K.-R. (2014). Polyline curvatures based robust vector data hashing. *Multimedia Tools and Applications*, 73(3):1913–1942.

Lee, S.-H. and Kwon, K.-R. (2010). CAD drawing watermarking scheme. *Digital Signal Processing*, 20(5):1379 – 1399.

Lee, S.-H. and Kwon, K.-R. (2013). Vector watermarking scheme for GIS vector map management. *Multimedia Tools and Applications*, 63(3):757–790.

Lele, W., Wei, L., and Yinghong, D. (2013). A good anti-robust algorithm with map watermarking. In Zhang, W., editor, *Advanced Technology in Teaching*, volume 163 of *Advances in Intelligent and Soft Computing*, pages 789–794. Springer Berlin.

Li, A., Chen, Y., Lin, B., Zhou, W., and Lu, G. (2008a). Review on copyright marking techniques of GIS vector data. In *International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pages 989–993.

Li, A., Lin, B.-x., Chen, Y., and Lu, G. (2008b). Study on copyright authentication of GIS vector data based on zero-watermarking. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XXXVII:1783–1786.

Li, A., Lv, G., Zhou, L., Lin, B., and Gu, Z. (2009). Real-time copyright protection for spatial data files. *Journal of Geo-Information Science*, 1.

Li, A., Zhou, W., Lin, B., and Chen, Y. (2008c). Copyright protection for gis vector data production. In *Geoinformatics 2008 and Joint Conference on GIS and Built Environment: Geo-Simulation and Virtual GIS Environments*, volume 7143, pages 71432–71439.

Li, A.-B., Li, S.-S., and Lv, G.-N. (2012a). Disguise and reduction methods of GIS vector data based on difference expansion principle. *Procedia Engineering*, 29(0):1344 – 1350.

Li, B., Zhang, X., Jiang, X., and Ai, Q. (2014a). A selective authentication watermarking algorithm for 2D CAD engineering drawings based on entity localization. In *Innovative Design and Manufacturing (ICIDM), Proceedings of the 2014 International Conference on*, pages 82–87.

Li, Q., Min, L.-q., Wang, F., Yang, Y.-q., and He, H.-z. (2011). A watermarking algorithm of anti douglas compression for vector map data. *Science of Surveying and Mapping*, 3.

Li, Q., Min, L.-q., Wu, B., and Yang, Y.-q. (2010). A practical blind digital watermarking scheme for vector map data. *Engineering of Surveying and Mapping*, 4.

Li, S.-S., Zhou, W., and Li, A.-B. (2012b). Image watermark similarity calculation of GIS vector data. *Procedia Engineering*, 29:1331–1337.

Li, X., Li, W., Anselin, L., Rey, S., and Koschinsky, J. (2014b). A MapReduce algorithm to create contiguity weights for spatial analysis of big data. In *Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Analytics for Big Geospatial Data*, pages 50–53, New York, NY, USA. ACM.

Li, Y. and Xu, L. (2003). A blind watermarking of vector graphics images. In *5th International Conference on Computational Intelligence and Multimedia Applications*, pages 424–429.

Li, Y. and Xu, L. (2004). Vector graphical objects watermarking scheme in wavelet domain. *Acta Photonica Sinica*, 1.

Liang, B., Rong, J., and Wang, C. (2010). A vector maps watermarking algorithm based on DCT domain. In *The Canadian Geomatics Conference and Symposium of Commission*, volume XXXVIII.

Lin, B. and Li, A. (2010). Study on benchmark system for copyright marking algorithms of GIS vector data. In *18th International Conference on Geoinformatics*, pages 1–5.

Lin, Z., Peng, F., and Long, M. (2018). A low-distortion reversible watermarking for 2d engineering graphics based on region nesting. *IEEE Transactions on Information Forensics and Security*, 13(9):2372–2382.

Lin, Z.-x., Peng, F., and Long, M. (2017). A reversible watermarking for authenticating 2d vector graphics based on bionic spider web. *Image Commun.*, 57(C):134–146.

Ling, Y., Lin, C.-F., and Zhang, Z.-y. (2012). A zero-watermarking algorithm for digital map based on DWT domain. In He, X., Hua, E., Lin, Y., and Liu, X., editors, *Computer, Informatics, Cybernetics and Applications*, volume 107 of *LNEE*, pages 513–521. Springer Netherlands.

Liu, R., Wang, H., and Yu, X. (2018). Shared-nearest-neighbor-based clustering by fast search and find of density peaks. *Information Sciences*, 450:200 – 226.

Liu, Y., Yang, F., Gao, K., Dong, W., and Song, J. (2017). A zero-watermarking scheme with embedding timestamp in vector maps for big data computing. *Cluster Computing*, 20(4):3667–3675.

Longley, P. A., Goodchild, M., Maguire, D. J., and Rhind, D. W. (2011). *Geographic Information Systems and Science.* John Wiley and Sons, 3 edition.

Longley, P. A., Goodchild, M. F., Maguire, D. J., and Rhind, D. W. (2005). *Geographic Information Systems and Science.* John Wiley & Sons, Ltd, 2nd ed. edition.

Lopez, C. (2002). Watermarking of digital geospatial datasets: a review of technical, legal and copyright issues. *International Journal of Geographical Information Science*, 16(6):589–607.

Lucchese, C., Vlachos, M., Rajan, D., and Yu, P. (2010). Rights protection of trajectory datasets with nearest-neighbor preservation. *The VLDB Journal*, 19(4):531–556.

Magalhaes, K. and Dahab, R. (2009). SB-RAWVEC - a semi-blind watermarking method for vector maps. In *IEEE International Conference on Communications*, pages 1–6.

Maguire, D. J. (2015). *ArcGIS: General-Purpose GIS Software*, pages 1–8. Springer International Publishing, Cham.

Maraş, S. S., Maraş, H. H., Aktuğ, B., Maraş, E. E., and Yildiz, F. (2010). Topological error correction of GIS vector data. *International Journal of Physical Sciences*, 5(5):476–483.

Marques, D., Magalhaes, K., and Dahab, R. (2007). Rawvec: A method for watermarking vector maps. In *VII Brazilian Symposium of Information Security and Computer Systems*.

Matheus, A. (2005). Authorization for digital rights management in the geospatial domain. In *the 5th ACM Workshop on Digital Rights Management*, pages 55–64.

McKenney, M. and Schneider, M. (2007). Spatial partition graphs: A graph theoretic model of maps. In Papadias, D., Zhang, D., and Kollios, G., editors, *Advances in Spatial and Temporal Databases*, pages 167–184, Berlin, Heidelberg. Springer Berlin Heidelberg.

Men, C., Cao, L., and Li, X. (2010a). Perception-based reversible watermarking for 2d vector maps. *SPIE*, 7744:34–38.

Men, C., Cao, L., Li, X., and Wang, N. (2010b). Global characteristic-based lossless watermarking for 2D-vector maps. In *International Conference on Mechatronics and Automation*, pages 276–281.

Men, C., Cao, L., and Sun, J. (2010c). A perception-based reversible watermarking algorithm for 2d-vector maps. *Chinese High Technology Letters*, 4.

Men, C.-g., Cao, L.-j., and Sun, J.-g. (2009). Reversible watermarking for 2D-vector maps based on graph spectral domain. *Journal of Harbin Institute of Technology*, 12.

Miller, H. J. and Han, J. (2009). *Geographic Data Mining and Knowledge Discovery*. CRC Press.

Min, L.-q. (2007). The digital watermark of vector geo-data. *Bulletin of Surveying and Mapping*, 1.

Min, L.-q., Li, Q., Yang, Y.-b., and Yu, Q.-h. (2009). A survey of watermarking techniques for vector map data. *Journal of Geomatics Science and Technology*, 2(110162).

Min, L.-q., Zhu, X.-z., and Li, Q. (2012). A robust blind watermarking of vector map. In Zhang, T., editor, *Instrumentation, Measurement, Circuits and Systems*, volume 127 of *Advances in Intelligent and Soft Computing*, pages 51–59. Springer Berlin.

Mohammed, G. N., Yasin, A., and Zeki, A. M. (2014). Robust image watermarking based on dual intermediate significant bit (DISB). In *6th International Conference on Computer Science and Information Technology (CSIT)*, pages 18–22.

Mouhamed, M., Rashad, A. M., and ella Hassanien, A. (2012). Blind 2D vector data watermarking approach using random table and polar coordinates. In *2nd International Conference on Uncertainty Reasoning and Knowledge Engineering*, pages 67–70.

Murti, K. C. S. and Tadimeti, V. R. (2011). A simplified geodrm model for SDI services. In *International ACM Conference on Communication, Computing and Security*, ICCCS '11, pages 545–548.

Mustafa, A. S. (2011). Copyright protection for GIS vector map based on wavelet transform. In *The European Conference on Information Management.*

Muttoo, S. K. and Kumar, V. (2012). Watermarking digital vector map using graph theoretic approach. *Annals of GIS*, 18(2).

Neyman, S., Sitohang, B., and Cahyono, F. (2013a). An improvement technique of fragile watermarking to assurance the data integrity on vector maps. In *International Conference on Computer, Control, Informatics and Its Applications*, pages 179–184.

Neyman, S., Wijaya, Y., and Sitohang, B. (2014a). A new scheme to hide the data integrity marker on vector maps using a feature-based fragile watermarking algorithm. In *Data and Software Engineering (ICODSE), 2014 International Conference on*, pages 1–6.

Neyman, S. N., Pradnyana, I. N. P., and Sitohang, B. (2014b). A new copyright protection for vector map using fft-based watermarking. *TELKOMNIKA Telecommunication, Computing, Electronics and Control*, 12(2):367–378.

Neyman, S. N., Sitohang, B., and Sutisna, S. (2013b). Reversible fragile watermarking based on difference expansion using manhattan distances for 2D vector map. *Procedia Technology*, 11:614 – 620.

Ng, R. T. and Han, J. (2002). CLARANS: a method for clustering objects for spatial data mining. *IEEE Transactions on Knowledge and Data Engineering*, 14(5):1003–1016.

Nin, J. and Ricciardi, S. (2013). Digital watermarking techniques and security issues in the information and communication society. In *27th International Conference on Advanced Information Networking and Applications Workshops*, pages 1553–1558.

Niu, X., Shao, C., and Wang, X. (2006). A survey of digital vector map watermarking. *International Journal of Innovative Computing, Information and Control*, 2(6):1301 – 1316.

Niu, X.-M., Shao, C.-Y., and Wang, X.-T. (2007). GIS watermarking: Hiding data in 2D vector maps. In Pan, J.-S., Huang, H.-C., Jain, L., and Fang, W.-C., editors, *Intelligent Multimedia Data Hiding*, volume 58 of *Studies in Computational Intelligence*, pages 123–155. Springer Berlin.

Ohbuchi, R., Ueda, H., and Endoh, S. (2002). Robust watermarking of vector digital maps. In *IEEE International Conference on Multimedia and Expo*, volume 1, pages 577–580.

Ohbuchi, R., Ueda, H., and Endoh, S. (2003). Watermarking 2D vector maps in the mesh-spectral domain. In *International Conference on Shape Modeling*, pages 216–225.

Okabe, A. (2016). *GIS-based Studies in the Humanities and Social Sciences*. CRC Press.

Pan, J., Zheng, J., and Zhao, G. (2013). Blind watermarking of NURBS curves and surfaces. *Computer-Aided Design*, 45(2):144 – 153.

Park, K., Kim, K., Kang, H., and Han, S. (2002). Digital geographical map watermarking using polyline interpolation. In Chen, Y.-C., Chang, L.-W., and Hsu, C.-T., editors, *Advances in Multimedia Information Processing*, volume 2532 of *LNCS*, pages 58–65. Springer Berlin.

Peng, F., Guo, R.-S., Li, C.-T., and Long, M. (2010). A semi-fragile watermarking algorithm for authenticating 2D CAD engineering graphics based on log-polar transformation. *Computer-Aided Design*, 42(12):1207–1216.

Peng, F., Lei, Y., and Sun, X. (2011). Reversible watermarking algorithm in wavelet domain for 2D CAD engineering graphics. *Journal of Image and Graphics*, 7.

Peng, F., Liu, Y., and Long, M. (2014a). Reversible watermarking for 2D CAD engineering graphics based on improved histogram shifting. *Computer-Aided Design*, 49:42 – 50.

Peng, F., Long, Q., Lin, Z.-X., and Min, L. (2017a). A reversible watermarking for authenticating 2d cad engineering graphics based on iterative embedding and virtual coordinates. *Multimedia Tools and Applications.*

Peng, F., Yan, Z.-J., and Long, M. (2017b). *A Reversible Watermarking for 2D Vector Map Based on Triple Differences Expansion and Reversible Contrast Mapping*, pages 147–158. Springer.

Peng, H., Jianya, G., and Liang, C. (2006). An improved adaptive watermarking algorithm for vector digital maps. In *IEEE International Conference on Geoscience and Remote Sensing Symposium*, pages 2844–2847.

Peng, Y., Lan, H., Yue, M., and Xue, Y. (2017c). Multipurpose watermarking for vector map protection and authentication. *Multimedia Tools and Applications*, 77:1–21.

Peng, Y., Wang, C., Fang, Y., and Li, W. (2012). Anonymous watermarking protocol for vector spatial data. In *International Conference on Computer Science and Service System*, pages 2095 – 2098.

Peng, Y. and Yue, M. (2015). A zero-watermarking scheme for vector map based on feature vertex distance ratio. *Journal of Electrical and Computer Engineering*, 2015:35:1–35:6.

Peng, Y.-l. (2010). Review of digital watermarking for vector digital map. *Journal of Xiangnan University*, 2.

Peng, Z., Yue, M., Wu, X., and Peng, Y. (2014b). Blind watermarking scheme for polylines in vector geo-spatial data. *Multimedia Tools and Applications*, pages 1–19.

Photis, Y. N. (2012). Redefinition of the Greek electoral districts through the application of a region-building algorithm. MPRA Paper 42398, University Library of Munich, Germany.

Pu, Y.-C., Du, W.-C., and Jou, I.-C. (2006). Toward blind robust watermarking of vector maps. In *18th International Conference on Pattern Recognition*, volume 3, pages 930–933.

Pu, Y.-C., Jou, I.-C., et al. (2009). Blind and robust watermarking for street-network vector maps. *Information Technology Journal*, 8(7):982–989.

Puri, S., Agarwal, D., He, X., and Prasad, S. K. (2013). Mapreduce algorithms for GIS polygonal overlay processing. In *2013 IEEE International Symposium on Parallel Distributed Processing, Workshops and Phd Forum*, pages 1009–1016.

Qiu, Q., Yao, X., Chen, C., Liu, Y., and Fang, J. (2015). A spatial data partitioning and merging method for parallel vector spatial analysis. In *2015 23rd International Conference on Geoinformatics*, pages 1–5.

Qiu, Y., Gu, H., and Sun, J. (2018a). High-payload reversible watermarking scheme of vector maps. *Multimedia Tools Appl.*, 77(5):6385–6403.

Qiu, Y., Gu, H., and Sun, J. (2018b). Reversible watermarking algorithm of vector maps based on ECC. *Multimedia Tools and Applications*, 77.

Raafat, M. M., Zawbaa, H. M., Al-Shammari, E., Hassanien, A. E., and Snasel, V. (2013). Blind watermark approach for map authentication using support vector machine. In *Advances in Security of Information and Communication Networks*, pages 84–97. Springer Berlin Heidelberg.

Ramaswmay, G. and Srinivasarao, V. (2010). A novel approach of cryptography and watermarking using to protect gis data. *Journal of Theoretical and Applied Information Technology*, 16(2):116 – 128.

Ren, N., sheng Wang, Q., and qing Zhu, C. (2014a). Selective authentication algorithm based on semi-fragile watermarking for vector geographical data. In *Geoinformatics (GeoInformatics), 2014 22nd International Conference on*, pages 1–6.

Ren, N., Zhu, C.-q., Ren, S.-j., and Zhu, Y.-s. (2014b). A digital watermark algorithm for tile map stored by indexing mechanism. In Buchroithner, M., Prechtel, N., and Burghardt, D., editors, *Cartography from Pole to Pole*, Lecture Notes in Geoinformation and Cartography, pages 79–86. Springer Berlin Heidelberg.

Sangita, Z.-C. and Venkatachalam, P. (2012a). Evaluation of spatial relations in watermarked geospatial data. In *3rd ACM SIGSPATIAL and International Workshop on GeoStreaming*, IWGS '12, pages 78–83, New York, NY, USA.

Sangita, Z.-C. and Venkatachalam, P. (2012b). Protecting geospatial data using digital watermarking. In *International Conference on Computer and Communication Engineering*, pages 594–598.

Sangita, Z.-C. and Venkatachalam, P. (2012c). Robust watermarking for protection of geospatial data. *Academic Journal*, 29.

Sangita, Z.-C. and Venkatachalam, P. (2013). Conceptual framework for geospatial data security. *International Journal of Database Management Systems*, 5(5):29 – 35.

Schulz, G. and Voigt, M. (2004). A high capacity watermarking system for digital maps. In *the ACM Workshop on Multimedia and Security*, pages 180–186.

Shao, C., Wang, H., Niu, X., and Wang, X. (2005). Shape-preserving algorithm for watermarking 2-D vector map data. In *IEEE 7th Workshop on Multimedia Signal Processing*, pages 1–4.

Shao, C. Y., Wang, H. L., Niu, X., and Wang, X. T. (2006). A shape-preserving method for watermarking 2d vector maps based on statistic detection. *IEICE Transactions on Information and Systems*, 89-D(3):1290 –1293.

Shearer, C. (2000). The CRISP-DM model: The new blueprint for data mining. *Journal of Data Warehousing*, 5(4).

Shi, X.-l. and Yang, W.-n. (2008). Research on digital watermarking of gis small data layer. *Surveying and Mapping of Sichuan*, 4.

Shujun, D., liang, L., Shujun, D., and Sen, C. (2007). Research on a digital watermarking algorithm suitable to vector map. In *IEEE International Conference on Automation and Logistics*, pages 1236–1240.

Shuliang, W., Gangyi, D., and Ming, Z. (2013). Big spatial data mining. In *IEEE International Conference on Big Data*, pages 13–21.

Solachidis, V., Nikolaidis, N., and Pitas, I. (2000a). Fourier descriptors watermarking of vector graphics images. In *International Conference on Image Processing*, volume 3, pages 9–12.

Solachidis, V., Nikolaidis, N., and Pitas, I. (2000b). Watermarking polygonal lines using fourier descriptors. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 1955–1958.

Solachidis, V. and Pitas, I. (2004). Watermarking polygonal lines using fourier descriptors. *IEEE Computer Graphics and Applications*, 24(3):44–51.

Sonnet, H., Isenberg, T., Dittmann, J., and Strothotte, T. (2003). Illustration watermarks for vector graphics. In *11th Pacific Conference on Computer Graphics and Applications*, pages 73–82.

Su, Z., Zhou, L., Mao, Y., Dai, Y., and Tang, W. (2017). A unified framework for authenticating topology integrity of 2d heterogeneous engineering CAD drawings. *Multimedia Tools Appl.*, 76(20):20663–20689.

Suk-Hwan, L., Xiao-Jiao, H., and Ki-Ryong, K. (2014). Vector watermarking method for digital map protection using arc length distribution. *IEICE Transactions on Information and Systems*, pages 34–42.

Sun, G., Shen, Z., and Chen, H. (2009a). Vector polygon blind watermarking based on canonical correlation analysis. In *International Conference on Multimedia Information Networking and Security*, volume 1, pages 544–548.

Sun, J.-g., Men, C.-g., Cao, L.-j., and Li, C.-m. (2010a). Digital watermarking of vector maps based on structure features. *Journal of Central South University(Science and Technology*, 4.

Sun, J.-g., Men, C.-g., Ma, C.-g., and Cao, L.-j. (2010b). Digital watermarking with authentication for vector maps. *Journal of Electronics and Information Technology*, 5.

Sun, J.-g., Men, C.-g., Yu, L.-f., and Cao, L.-j. (2009b). Survey of digital watermarking for the vector maps. *Computer Science*, 9.

Tao, S., Dehe, X., Chengming, L., and Jianguo, S. (2009). Watermarking gis data for digital map copyright protection. In *24th International Cartographic Conference.*

Tian, Z., Chen, G., Zhang, X., Zheng, Y., and Li, G. (2004). Digital watermark: Technique, application and improvement as a copyright-protecting method for RS and cartographic data. In *MTTS/IEEE TECHNO-OCEAN Conference*, volume 2, pages 776–780.

Tie, X., Zou, J., Zhong, W., and Qi, D. (2007). Watermarking polygonal lines using V-descriptors. In *2nd International Conference on Pervasive Computing and Applications*, pages 442–446.

Tulapurkar, H., Choudhary, S. Z., Ansari, R., Arun, P. V., Mohanty, J., Venkatachalam, P., and Buddhiraju, K. M. (2018). *Digital Watermarking of Geospatial Data*, pages 285–304. Springer, Singapore.

Tulapurkar, H., Mohan, B. K., and Bharadi, V. (2017). Invisible watermarking algorithm for GIS data using curvelet transform — comparitive study with wavelet. In *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 3389–3392.

Urvoy, M., Goudia, D., and Autrusseau, F. (2014). Perceptual dft watermarking with improved detection and robustness to geometrical distortions. *IEEE Transactions on Information Forensics and Security*, 9(7):1108–1119.

Vlachos, M., Lucchese, C., Rajan, D., and Yu, P. S. (2008). Ownership protection of shape datasets with geodesic distance preservation. In *11th International Conference on Extending Database Technology: Advances in Database Technology*, EDBT '08, pages 276–286.

Voigt, M. and Busch, C. (2002). Watermarking 2D-vector data for geographical information systems. *SPIE*, 4675:621–628.

Voigt, M. and Busch, C. (2003). Feature-based watermarking of 2D vector data. In *Security and Watermarking of Multimedia Contents*, volume 5020, pages 359–366.

Voigt, M., Yang, B., and Busch, C. (2004). Reversible watermarking of 2D-vector data. In *the 2004 Workshop on Multimedia and Security*, pages 160–165.

Voigt, M., Yang, B., and Busch, C. (2005). High-capacity reversible watermarking for 2D vector data. *SPIE*, 5681:409–417.

Wang, C., Cheng, L., Zhao, Q., Zhang, L., and Guo, L. (2010a). A geographical data copyright protection algorithm based on discrete cosine transform. *Journal of Shihezi University Natural Science*, 4.

Wang, C., Peng, Z., Peng, Y., and Yu, L. (2009a). Watermarking 2D vector maps on spatial topology domain. In *International Conference on Multimedia Information Networking and Security*, volume 2, pages 71–74.

Wang, C., Peng, Z., Peng, Y., Yu, L., Wang, J., and Zhao, Q. (2012a). Watermarking geographical data on spatial topological relations. *Multimedia Tools and Applications*, 57(1):67–89.

Wang, C., Wang, W., Wang, Q., and Qin, Q. (2009b). A watermarking algorithm for vector maps in spatial domain. *Wuhan University Journal of Geomatics and Information Science*, 2.

Wang, C., Wang, W., Wu, B., and Qin, Q. (2009c). A watermarking algorithm for vector data based on spatial domain. In *1st International Conference on Information Science and Engineering*, pages 1959–1962.

Wang, C., Zhang, L., Liang, B., Zheng, H., Du, W., and Peng, Y. (2011). Watermarking vector maps based on minimum encasing rectangle. In *International Conference on Intelligent Computation Technology and Automation*, volume 2, pages 1243–1246.

Wang, C., Zhao, Q., and Zhong, F. (2010b). A shape-preserving and robust watermarking algorithm for vector maps. In *International Conference on Computational and Information Sciences*, pages 590–593.

Wang, D.-m. (2008). Application of digital watermarking in copyright protection of engineering drawing. *Manufacture Information Engineering of China*, 17.

Wang, N. (2017). Reversible watermarking for 2D vector maps based on normalized vertices. *Multimedia Tools and Applications*, 76(20):20935–20953.

Wang, N., Bian, J., and Zhang, H. (2015a). Rst invariant fragile watermarking for 2D vector map authentication. *International Journal of Multimedia and Ubiquitous Engineering*, 10:155–172.

Wang, N. and Kankanhalli, M. (2018). 2D vector map fragile watermarking with region location. *ACM Trans. Spatial Algorithms Syst.*, 4(4):12:1–12:25.

Wang, N. and Men, C. (2012). Reversible fragile watermarking for 2-D vector map authentication with localization. *Computer-Aided Design*, 44(4):320–330.

Wang, N. and Men, C. (2013). Reversible fragile watermarking for locating tampered blocks in 2D vector maps. *Multimedia Tools and Applications*, 67(3):709–739.

Wang, N., Zhang, H., and Men, C. (2014). A high capacity reversible data hiding method for 2d vector maps based on virtual coordinates. *Computer-Aided Design*, 47:108 – 117.

Wang, N. and Zhao, X. (2018a). 2D vector map reversible data hiding with topological relation preservation. In *International Conference on Acoustics, Speech and Signal Processing*, pages 2097–2101.

Wang, N. and Zhao, X. (2018b). 2D vector map reversible data hiding with topological relation preservation. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2097–2101.

Wang, N., Zhao, X., and Xie, C. (2016a). Rst invariant reversible watermarking for 2D vector map. *International Journal of Multimedia and Ubiquitous Engineering*, 11:265–276.

Wang, N., Zhao, X., and Zhang, H. (2015b). Block-based reversible fragile watermarking for 2d vector map authentication. *International Journal of Digital Crime and Forensics*, 7(3):60–80.

Wang, Q., Cheng, G., and Liu, H. (2017). A semi-fragile watermarking algorithm against geometric transformation for vector geographic data. In *Proceedings of the International Conference on Watermarking and Image Processing*, ICWIP 2017, pages 11–15, New York, NY, USA. ACM.

Wang, Q. and Zhu, C. (2012). Fragile watermarking algorithm for vector geographic data exact authentication. *Journal of Geomatics Science and Technology*, 3.

Wang, S., Chen, C.-S., Rinsurongkawong, V., Akdag, F., and Eick, C. F. (2010c). A polygon-based methodology for mining related spatial datasets. In *Proceedings of the 1st ACM SIGSPATIAL International Workshop on Data Mining for Geoinformatics*, pages 1–8.

Wang, S. and Eick, C. F. (2014). A polygon-based clustering and analysis framework for mining spatial datasets. *GeoInformatica*, 18(3):569–594.

Wang, S.-M. and Chiu, C.-S. (2012). A reversible information hiding algorithm for 2d vector maps. In *International Symposium on Intelligent Signal Processing and Communications Systems*, pages 424–429.

Wang, W., Du, S., Guo, Z., and Luo, L. (2015c). Polygonal clustering analysis using multilevel graph-partition. *Transactions in GIS*, 19(5):716–736.

Wang, X., Huang, D., and Zhang, Z. (2012b). A robust zero-watermarking algorithm for vector digital maps based on statistical characteristics. *Journal of Software*, 7(10):2349–2356.

Wang, X., Pang, K., Zhou, X., Zhou, Y., Li, L., and Xue, J. (2015d). A visual model-based perceptual image hash for content authentication. *IEEE Transactions on Information Forensics and Security*, 10(7):1336–1349.

Wang, X., Shao, C., Xu, X., and Niu, X. (2007). Reversible data-hiding scheme for 2-D vector maps based on difference expansion. *IEEE Transactions on Information Forensics and Security*, 2(3):311–320.

Wang, Y., Yang, C., qing Zhu, C., Ren, N., and Chen, P. (2016b). A novel multiple watermarking algorithm based on correlation detection for vector geographic data. In *GRMSE*.

Wang, Y., Yang, C., and Zhu, C. (2018). A multiple watermarking algorithm for vector geographic data based on coordinate mapping and domain subdivision. *Multimedia Tools and Applications*, 77(15):19261–19279.

Wang, Y.-s. and Xu, M.-z. (2003). Scale digital watermarking algorithm based on two-dimensional engineering graphics. *Journal of Nanchang University in Engineering and Technology*, 4.

Wei, H., Du, Y., Liang, F., Zhou, C., Liu, Z., Yi, J., Xu, K., and Wu, D. (2015). A k-d tree-based algorithm to parallelize kriging interpolation of big spatial data. *GIScience & Remote Sensing*, 52(1):40–57.

Wu, B., Wang, W., Peng, Z., and Du, D. (2009a). A new algorithm for watermarking building polygons. In *International Conference on Digital Image Processing*, pages 366–370.

Wu, B., Wang, W., Peng, Z., Du, D., and Wang, C. (2010). Design and implementation of spatial data watermarking service system. *Geo-spatial Information Science*, 13(1):40–48.

Wu, D. (2012). A reversible watermarking scheme for 2d vector maps. In Wu, Y., editor, *Software Engineering and Knowledge Engineering: Theory and Practice*, volume 115 of *Advances in Intelligent and Soft Computing*, pages 197–203. Springer Berlin.

Wu, D., Wang, G., and Gao, X. (2009b). Reversible watermarking of SVG graphics. In *International Conference on Communications and Mobile Computing*, volume 3, pages 385–390.

Wu, D. and Wang, G.-z. (2009). Reversible digital watermarking scheme for 2-D vector maps based on difference expansion and shifting. *Journal of Optoelectronics.Laser*, 7.

Wu, J., Liu, Q., Wang, J., and Gao, L. (2013a). A robust watermarking algorithm for 2D CAD engineering graphics based on DCT and chaos system. In Tan, Y., Shi, Y., and Mo, H., editors, *Advances in Swarm Intelligence*, volume 7929 of *LNCS*, pages 215–223. Springer Berlin.

Wu, J., Yang, F., and Wu, C. (2013b). Review of digital watermarking for 2D-vector map. In *IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing*, pages 2098–2101.

Wu, J., Yang, F., and Wu, C. (2013c). Review of digital watermarking for 2D-vector map. In *IEEE International Conference on Green Computing and Communications, Internet of Things and Cyber, Physical and Social Computing*, pages 2098–2101.

Xu, D.-h., Zhu, C.-q., and Wang, Q.-s. (2007). A survey of the research on digital watermarking for the vector digital map. *Geomatics World*, 6.

Xu, X., Ester, M., Kriegel, H. P., and Sander, J. (1998). A distribution-based clustering algorithm for mining in large spatial databases. In *Proceedings 14th International Conference on Data Engineering*, pages 324–331.

Xun, W., Ding-jun, H., and Zhi-yong, Z. (2012). *A Robust Zero-Watermarking Algorithm for 2D Vector Digital Maps*, volume 107 of *Lecture Notes in Electrical Engineering*, chapter 56, pages 533–541. Springer Netherlands.

Xun, W., Hai, L., and Hujun, B. (2004). A robust watermarking algorithm for vector digital mapping. *Journal of Computer Aided Design and Computer Graphics*, 10.

Yamada, T., Fujii, Y., Tezuka, S., and Komoda, N. (2006). Evaluation of digital watermarking system for vector map content distribution. In *International Conference on Service Systems and Service Management*, volume 2, pages 1637–1642.

Yan, H. and Li, J. (2011). A blind watermarking approach to protecting geospatial data from piracy. *International Journal of Information and Education Technology*, 1(2):94–98.

Yan, H. and Li, J. (2012). Blind watermarking technique for topographic map data. *Applied Geomatics*, 4(4):225–229.

Yan, H., Li, J., and Wen, H. (2011). A key points-based blind watermarking approach for vector geo-spatial data. *Computers, Environment and Urban Systems*, 35(6):485–492.

Yan, H., Zhang, L., and Yang, W. (2017). A normalization-based watermarking scheme for 2d vector map data. *Earth Science Informatics*, 10.

Yang, C.-s. and Zhu, C.-q. (2007). Watermarking algorithm for vector geo-spatial data on wavelet transformation. *Journal of Zhengzhou Institute of Surveying and Mapping*, 1.

Yue, M., Peng, Z., and Peng, Y. (2014). A fragile watermarking scheme for modification type characterization in 2d vector maps. In Chen, L., Jia, Y., Sellis, T., and Liu, G., editors, *Web Technologies and Applications*, volume 8709 of *Lecture Notes in Computer Science*, pages 129–140. Springer International Publishing.

Yun, H., Hongtao, W., Hanyu, Z., Xinxin, N., and Yixian, Y. (2004). A digital watermark scheme in vector data. *Computer Engineering and Applications*, 21.

Zhanchuan, C., Wei, S., Changzhen, X., and Dongxu, Q. (2005). Watermarking of two-dimensional engineering graph based on the orthogonal complete u-system. In *9th International Conference on Computer Aided Design and Computer Graphics*.

Zhang, C., Zhang, X., Zhang, D., and Jiao, Y. (2008a). Digital watermarking of vector map based on vector angle. In *International Conference on Intelligent Computation Technology and Automation*, volume 2, pages 127–130.

Zhang, D., Qian, D., and Han, P. (2007). A new attributes-priority matching watermarking algorithm satisfying topological conformance for vector map. In *3rd International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, volume 2, pages 469–472.

Zhang, G., Da, Q., Zhang, L., Sun, J., Han, Q., Kou, L., and Wang, W. (2018a). *A Fragile Watermarking Scheme of Anti-deleting Features for 2D Vector Map*, pages 360–368. Springer.

Zhang, G., Da, Q., Zhang, L., Sun, J., Han, Q., Kou, L., and Wang, W. (2018b). *A Novel Fragile Watermarking Scheme for 2D Vector Map Authentication*, pages 880–889. Springer.

Zhang, H. and Li, Y. (2009). Toward a 2D vector map with a feature nodes-based watermarking method. *SPIE*, 7146:1–10.

Zhang, H.-l. and Gao, M.-m. (2009). A semi-fragile digital watermarking algorithm for 2d vector graphics tamper localization. In *International Conference on Multimedia Information Networking and Security*, volume 1, pages 549–552.

Zhang, J., Samal, A., and Soh, L. (2005). Polygon-based spatial clustering. In *The 8th International Conference on GeoComputation*, pages 1–5.

Zhang, L., Li, A., Lv, G., and Lin, B. (2008b). Study on adaptive watermark of GIS vector data. *Geo-Information Science*, 6.

Zhang, L., Yan, D., Jiang, S., and Shi, T. (2010). A new robust watermarking algorithm for vector data. *Wuhan University Journal of Natural Sciences*, 15(5):403–407.

Zhang, X., Huang, B., and Tay, R. (2016). Estimating spatial logistic model: A deterministic approach or a heuristic approach? *Information Sciences*, 330:358 – 369. SI Visual Info Communication.

Zhang, Y. and Wang, Q. (2011). Digital watermarking algorithm of vector map based on feature points. In *Cross Strait Quad-Regional Radio Science and Wireless Technology Conference*, volume 2, pages 1430–1433.

Zhang, Z., Wang, Y., and Sun, S. (2009a). An anti-compression watermarking scheme for vector map based on improved douglas-peucker algorithm. In *1st International Workshop on Education Technology and Computer Science*, volume 2, pages 1075–1079.

Zhang, Z.-l. (2010). Anti-compression watermark algorithm for vector map. *Computer Engineering*, 20.

Zhang, Z.-l., Sun, S.-s., Wang, Y.-m., and Zheng, K.-b. (2009b). Zero-watermarking algorithm for 2D vector map. *Computer Engineering and Design*, 6.

Zhao, H., Du, S., and Zhang, D. (2010a). A reversible watermarking scheme for 2d vector drawings based on difference expansion. In *11th International Conference on Computer-Aided Industrial Design Conceptual Design*, volume 2, pages 1441–1446.

Zhao, H., Yuan, W., and Wang, Z. (2008). A new watermarking scheme for cad engineering drawings. In *9th International Conference on Computer-Aided Industrial Design and Conceptual Design*, pages 518–522. IEEE.

Zhao, J., Lin, H., and Wang, F. (2010b). Study on data quality evaluation of vector map watermarking. *Journal of Image and Graphics*, 7.

Zhao, L., Chen, L., Ranjan, R., Choo, K.-K. R., and He, J. (2016). Geographical information system parallelization for spatial big data processing: a review. *Cluster Computing*, 19(1):139–152.

Zhao, Q., Sui, L., Wang, C., and Yin, X. (2013a). Publicly verify the integrity of the geographical data using public watermarking scheme. In *Geo-Informatics in Resource Management and Sustainable Ecosystem*, volume 398, pages 646–652. Springer Berlin Heidelberg.

Zhao, Q., Sui, L., Wang, C., and Yin, X. (2013b). Publicly verify the integrity of the geographical data using public watermarking scheme. In Bian, F., Xie, Y., Cui, X., and Zeng, Y., editors, *Geo-Informatics in Resource Management and Sustainable Ecosystem*, volume 398 of *Communications in Computer and Information Science*, pages 646–652. Springer Berlin.

Zheng, L., Chen, R., and Cheng, X. (2011). Research and implementation of digital rights management model for vector graphics. In *International Conference on Uncertainty Reasoning and Knowledge Engineering*, volume 2, pages 17–20.

Zheng, L., Chen, R., Li, L., and Li, Y. (2010a). Study on digital watermarking for vector graphics. In *2nd International Workshop on Education Technology and Computer Science*, volume 2, pages 535–538.

Zheng, L., Feng, L., Li, Y., and Cheng, X. (2010b). Research on digital rights management model for spatial data files. In *2nd International Conference on Information Engineering and Computer Science*, pages 1–4.

Zheng, L., Jia, Y., and Wang, Q. (2009). Research on vector map digital watermarking technology. In *1st International Workshop on Education Technology and Computer Science*, volume 1, pages 304–307.

Zheng, L., Li, Y., Feng, L., and Liu, H. (2010c). Research and implementation of fragile watermark for vector graphics. In *2nd International Conference on Computer Engineering and Technology*, volume 1, pages 522–525.

Zheng, L., Xie, K., Li, Y., Liu, H., and Li, T. (2010d). A digital watermark scheme for vector graphics. In *International Conference on Image Analysis and Signal Processing*, pages 699–702.

Zheng, L. and You, F. (2009). A fragile digital watermark used to verify the integrity of vector map. In *International Conference on E-Business and Information System Security*, pages 1–4.

Zhong, S., Hu, Y., and Lu, J. (2006). A new geometric-transformation robust and practical embedding scheme for watermarking 2d vector maps in the graph spectral domain. In *International Conference on Communications, Circuits and Systems*, volume 1, pages 24–30.

Zhou, Q., Ren, N., Zhu, C., and Tong, D. (2018). Storage feature-based watermarking algorithm with coordinate values preservation for vector line data. *KSII Transactions on Internet and Information Systems*, 12(7).

Zhou, X. and Bi, D.-y. (2004). Use digital watermarking to protect GIS data by chinese remaindering. *Journal of Image and Graphics*, 5.

Zhou, X. and Pan, X. (2006). Watermark-based scheme to protect copyright of SVG data. In *International Conference on Computational Intelligence and Security*, volume 2, pages 1199–1202.

Zhou, X., Ren, Y., and Pan, X. (2006). Watermark embedded in polygonal line for copyright protection of contour map. *International Journal of Computer Science and Network Security*, 6(7):202–205.

Zhou, Y., Li, A., and Lv, G. (2010). Research of robustness evaluation method for GIS vector data digital watermarking algorithm. In *18th International Conference on Geoinformatics*, pages 1–6.

Zhu, C., Yang, C., and Ren, N. (2010). Application of digital watermarking to geospatial data security. *Bulletin of Surveying and Mapping*, 10.

Zhu, C.-q., Yang, C.-s., and Wang, Q.-S. (2008). A watermarking algorithm for vector geo-spatial data based on integer wavelet transform. In *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVII, pages 15–18.

Zhu, J.-f., Deng, S.-h., and Xu, W.-z. (2011). Integration of a variety of algorithms in high robustness watermarking for vector map data. *Science of Surveying and Mapping*, 2.

Zope-Chaudhari, S., Venkatachalam, P., and Buddhiraju, K. M. (2015). Assessment of distortion in watermarked geospatial vector data using different wavelets. *Geo-spatial Information Science*, 18(2-3):124–133.

Zope-Chaudhari, S., Venkatachalam, P., and Buddhiraju, K. M. (2017). Copyright protection of vector data using vector watermark. In *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 6110–6113.

Zuo, C., Li, A., and Meng, C. (2010). GIS vector data automatic watermark detection based on mobile agent technology. In *Geoinformatics*, pages 1–4.

Appendix A

UPR16 Form – Ethics Review Checklist

FORM UPR16

Research Ethics Review Checklist

Please include this completed form as an appendix to your thesis (see the Research Degrees Operational Handbook for more information

Postgraduate Research Student (PGRS) Information		**Student ID:**	145008
PGRS Name:	Ahmed Abubahia		
Department:	School of Computing	**First Supervisor:**	Mihaela Cocea
Start Date: (or progression date for Prof Doc students)	01/10/2013		

Study Mode and Route:	Part-time □ Full-time ☒	MPhil □ PhD ☒	MD □ Professional Doctorate □			

Title of Thesis:	Spatial Data Mining Approaches for GIS Vector Data Processing
Thesis Word Count: (excluding ancillary data)	44000

If you are unsure about any of the following, please contact the local representative on your Faculty Ethics Committee for advice. Please note that it is your responsibility to follow the University's Ethics Policy and any relevant University, academic or professional guidelines in the conduct of your study

Although the Ethics Committee may have given your study a favourable opinion, the final responsibility for the ethical conduct of this work lies with the researcher(s).

UKRIO Finished Research Checklist:
(If you would like to know more about the checklist, please see your Faculty or Departmental Ethics Committee rep or see the online version of the full checklist at: http://www.ukrio.org/what-we-do/code-of-practice-for-research/)

a) Have all of your research and findings been reported accurately, honestly and within a reasonable time frame?	YES ☒ NO □
b) Have all contributions to knowledge been acknowledged?	YES ☒ NO □
c) Have you complied with all agreements relating to intellectual property, publication and authorship?	YES ☒ NO □
d) Has your research data been retained in a secure and accessible form and will it remain so for the required duration?	YES ☒ NO □
e) Does your research comply with all legal, ethical, and contractual requirements?	YES ☒ NO □

Candidate Statement:

I have considered the ethical dimensions of the above named research project, and have successfully obtained the necessary ethical approval(s)

Ethical review number(s) from Faculty Ethics Committee (or from NRES/SCREC):	0B3A-2067-B577-E09B-28AB-CCAC-C4B2-EE9F

If you have *not* submitted your work for ethical review, and/or you have answered 'No' to one or more of questions a) to e), please explain below why this is so:

Signed *(PGRS)*:	**Date:** 17/08/2018