

Refinement of Existing Road Vector Layers Through High Resolution Satellite Images

M. G. S. Perera



Refinement of Existing Road Vector Layers Through High Resolution Satellite Images

**M. G. S. Perera
Index No: 13000918**

Supervisor: Dr. D. D. Karunaratne

December 2017

Submitted in partial fulfillment of the requirements of the
B.Sc. (Hons) in Computer Science Final Year Project (SCS4124)



Declaration

I certify that this dissertation does not incorporate, without acknowledgement, any material previously submitted for a degree or diploma in any university and to the best of my knowledge and belief, it does not contain any material previously published or written by another person or myself except where due reference is made in the text. I also hereby give consent for my dissertation, if accepted, be made available for photocopying and for interlibrary loans, and for the title and abstract to be made available to outside organizations.

Candidate Name:

.....

Signature of Candidate

Date:

This is to certify that this dissertation is based on the work of

Ms. M. G. S. Perera

under my supervision. The thesis has been prepared according to the format stipulated and is of acceptable standard.

Supervisor Name: Dr. D. D. Karunaratne

.....

Signature of Supervisor

Date:

Supervisor Name: Dr. K. H. E. L. W. Hettiarachchi

.....

Signature of Supervisor

Date:

Abstract

Geospatial data originate from different sources. Integrating multiple geospatial datasets which originate from different sources can provide insights and capabilities that are not possible from a single dataset. However, positional misalignments can occur during this integration process resulting in a requirement of an alignment approach to conflate or integrate different geospatial data types while reducing positional inconsistencies.

In this dissertation, two novel alignment approaches are proposed to address the problem of misalignment between geospatial data in road vector layer and satellite imagery. The first approach, focuses on the distance ratio and angle between the provided control points in the two layers for the alignment process. Whereas, the second approach uses the concept of Bézier curves for alignment. The first three steps of these two approaches are the same. While the two approaches differ in the final step based on the concept used for the alignment process. The initial step is to decompose the road vector layer into a set of points. In the second step, the satellite imagery and the vector layer are transformed to the same coordinate reference system. Third and the most important step is providing control points from the satellite imagery and vector layer. Final step is to carryout the alignment.

A novel evaluation model is proposed to evaluate the accuracy of the two proposed approaches. The proposed approaches are evaluated against the existing Piecewise Linear Rubber sheeting method on the Sri Lankan road vector data on four test scenarios consist of road segments with different misalignments. The research shows that the proposed bézier approach and distance ratio and angle approach has an average misalignment reduction percentage of 75.8% and 73.6% respectively whereas the piecewise linear rubber-sheeting method has an average misalignment reduction percentage of 68.6%.

Preface

Two novel alignment approaches to address the problem of misalignment between road vector layer and satellite imagery are introduced in this dissertation. The first approach focuses on alignment of the road vector layer considering distance ratio and angle between the provided control points. The concept used for the alignment process behind this approach was solely my own work and has not been proposed in any other study related to vector to imagery conflation in the domain of Geographic Information Systems. The second alignment approach introduced in this research is based on the concept of Bézier curves in the domain of Computer Graphics. The concept of Bézier curves were utilized in this research as a road segment can be viewed as a parametric curve. Polynomial interpolation and curve transformation concepts related to Bézier curves were utilized to carryout the alignment process in this approach. However, the integration of these concepts to the alignment process was proposed by myself and has not been proposed in any other study related to the domain of vector to imagery conflation. The evaluation model introduced in this dissertation is a novel evaluation model and has not been proposed in any other work in the domain of Geographic Information Systems.

Acknowledgement

I would like to express my sincere gratitude to my research supervisor, Dr. D. D. Karunaratne, senior lecturer of University of Colombo School of Computing and my research co-supervisor, Dr. K. H. E. L. W. Hettiarachchi, senior lecturer of University of Colombo School of Computing for providing me continuous guidance and supervision throughout the research.

I would also like to extend my sincere gratitude to Dr. K. D. Sandaruwan, senior lecturer of University of Colombo School of Computing and Dr. M. I. E. Wickramasinghe, lecturer of University of Colombo School of Computing for providing feedback on my research proposal and interim evaluation to improve my study. I also take the opportunity to acknowledge the assistance provided by Dr. H. E. M. H. B. Ekanayake as the final year computer science project coordinator.

I appreciate the feedback and motivation provided by my friends to achieve my research goals. This thesis is also dedicated to my loving family who has been an immense support to me throughout this journey of life. It is a great pleasure for me to acknowledge the assistance and contribution of all the people who helped me to successfully complete my research.

Table of Contents

Declaration	i
Abstract	ii
Preface	iii
Acknowledgement	iv
Table of Contents	v
List of Figures	vii
List of Tables	viii
List of Acronyms	ix
Chapter 1 - Introduction	1
1.1. Background to the Research	2
1.2. Research Problem and Research Questions	5
1.3. Research Aim and Objectives	6
1.4. Justification for the Research	6
1.5. Methodology	7
1.6. Outline of the Dissertation	7
1.7. Delimitations of Scope	8
1.8. Summary.....	8
Chapter 2 - Literature Review	9
2.1. Introduction.....	9
2.2. Approaches for Control Point Identification	9
2.3. Traditional Conflation Algorithms	11
2.3.1. Piecewise Linear Rubber-Sheeting	12
2.3.2. Snake Related Algorithms	14
2.4. Summary.....	15
Chapter 3 - Design	16
3.1. Introduction.....	16
3.2. Research Design	16
3.3. Alignment of road vector layer considering distance ratio and angle	18

3.4. Alignment of road vector layer considering Bézier curves	20
3.5. Summary.....	24
Chapter 4 - Implementation.....	25
4.1. Introduction.....	25
4.2. Software Tools.....	25
4.3. Implementation Details - Alignment of road vector layer considering distance ratio and angle	25
4.4. Implementation Details - Alignment of road vector layer considering Bézier curves	27
4.5. Summary.....	31
Chapter 5 - Results and Evaluations.....	32
5.1. Introduction.....	32
5.2. Evaluation Model	32
5.3. Results	33
5.4. Summary.....	48
Chapter 6 - Conclusions.....	49
6.1. Introduction.....	49
6.2 Conclusions about research questions (aims/objectives).....	49
6.3 Conclusions about research problem.....	50
6.4 Limitations	50
6.5 Implications for further research	51
References.....	52
Appendix A: Diagrams.....	55
Appendix B: Code Listings.....	63

List of Figures

Figure 1.1: Misalignment between road vector layer and road layer in satellite imagery..... 5

Figure 1.2: Proposed research methodology..... 7

Figure 3.1: Research Design..... 16

Figure 3.2: Diagram of the refinement process..... 17

Figure 3.3: The geometry representation for any given point P in the vector layer..... 19

Figure 3.4: New location P' for any given point P in the vector layer with respect to R1 and R2.
..... 19

Figure 3.5: Generation of a quadratic bezier curve as illustrated in [24]..... 21

Figure 5.1: Polygon generated between the real road and the aligned road..... 32

Figure 5.2: Main steps of the proposed approach I..... 34, 35

Figure 5.3: Main steps of the proposed approach II considering order two bezier curve..... 36, 37

Figure 5.4: Main steps of the proposed approach II considering order three bezier curve .. 38, 39

Figure 5.5: Average MRP for all alignment approaches..... 47

List of Tables

Table 1.1: Classification of conflation processes as presented in [5, Tab.1]..... 3

Table 5.1: Results obtained for Test Scenario 1..... 40

Table 5.2: Results obtained for Test Scenario 2..... 41

Table 5.3: Results obtained for Test Scenario 3..... 42

Table 5.4: Results obtained for Test Scenario 4..... 43

Table 5.5: MRP for Test Scenario 1..... 44

Table 5.6: MRP for Test Scenario 2..... 44

Table 5.7: MRP for Test Scenario 3..... 45

Table 5.8: MRP for Test Scenario 4..... 45

Table 5.9: Comparison of the proposed approaches with existing approaches..... 47

List of Acronyms

GIS	Geographic Information System
GDB	Geographic Database
DEM	Digital Elevation Model
CSR	Coordinate Reference System
MRP	Misalignment Reduction Percentage
NA	Not Applicable

Chapter 1 - Introduction

Geographic Information System (GIS) is a computer based tool for analyzing and mapping spatial data. It encompasses querying and statistical analysis which are common database operations while providing unique visualization and geographic analysis benefits offered by maps. Data in GIS represent a simplified view of physical entities such as roads, mountains, accident locations or other features which interest the user.

The GIS database encompasses two type of main data models, Spatial data and Attribute Data. Spatial data represents the location of an object while attribute data represents the characteristics of the objects. Generally, spatial data and attribute data are maintained separately but joined during analysis. Considering spatial data models, two basic types of data models have evolved for storing geographic data digitally. Those two data models are known as vector data and raster data.

Raster data models represent the world as a set of cells in the grid pattern. Which means the environment is represented as a matrix of cells (pixels) and the value assigned to the cells will represent information such as temperature, elevation, or spectral data. Satellite images, aerial images, scanned maps fall into category of raster data models.

Vector data models use a set of coordinates and associated attribute data to define discrete objects. Coordinates define spatial locations and shape, attributes record the important nonspatial characteristics. The fundamental concept of vector model in GIS is that all geographic features of the real world are represented through points, lines and polygons.

Important data of base maps are generally stored in raster data. But most of the manipulations are carried out using vector data models.

However, since these above mentioned spatial datasets originate from different sources, positional inconsistencies (misalignments/deviations) can occur when considering the same object in different data formats in GIS (vector data/raster data) [1, 2].

1.1. Background to the Research

The availability of various geospatial data has increased in the current context. Integrating these multiple geospatial datasets can provide insights and capabilities that are not possible with individual datasets [1-3]. Thus, integrating multiple geospatial datasets has become an important aspect in the study of GIS.

In GIS researches, “conflation” is the most often used synonym for integration of multiple geospatial data from different sources. Integration/conflation is the process of combining two or more spatial representations of the same region to produce a superior dataset better than any of the original datasets. Through conflation, individual strengths of different datasets can be aggregated.

However, since these geospatial datasets originate from different sources positional inconsistencies (misalignments/deviations) can occur and they often do not match well to each other [1, 2]. Further, since this positional inconsistency is nonsystematic, carrying out a simple global transformation will not solve this problem. Manual correction of these inconsistencies can be very repetitive, labor intensive, and time consuming which is often not practical [2]. Thus, making the conflation/integration process a challenging task.

Conflation was first introduced in 1988 by Saalfeld [4], and the initial focus of the conflation process was to eliminate the spatial inconsistency between vector maps to improve spatial accuracy. Since then, various conflation techniques have been proposed to achieve the alignments of geospatial datasets. M. Ureña et al. [5] provided a comprehensive classification of the conflation processes, considering several aspects of the conflation processes as shown in Table 1.1.

Table 1.1: Classification of conflation processes as presented in [5, Tab. 1]

<p>According to the matching criteria used</p> <p>Geometric conflation Semantic conflation Topological conflation</p>	<p>According to the representation model used</p> <p>Vector to vector conflation Vector to raster conflation Raster to raster conflation Conflation between an image and a DEM (Digital Elevation Model) Conflation between two DEMs</p>
<p>According to the categorization problem</p> <p>Vertical conflation Horizontal conflation Temporal conflation</p>	<p>According to the automatization level applied</p> <p>Automatic conflation Semiautomatic conflation Manual conflation</p>

According to the criteria used to match geospatial objects, conflation processes are classified as geometric, semantic and topological conflation. Geometric conflation is concerned with minimizing the geometric differences between two Geographic Databases (GDB) when transforming features from one GDB to another. This method of conflation only minimizes the positional inconsistency and does not guarantee the correct matching of the homologous elements of the GDBs. Semantic conflation focuses on minimizing the semantic differences between the two GDBs. However, after removing the positional inconsistencies and semantic differences there is no guarantee of a correct matching between the GDBs. Therefore, topological conflation focuses on preserving topological relationships and optimizing geometric and semantic adjustments [5].

Based on the categorization problem, conflation approaches are classified as vertical, horizontal and temporal. Vertical conflation is related with detecting and eliminating the differences between spatial datasets in the same geographic region. Horizontal conflation is concerned with detecting and eliminating common boundaries of adjacent datasets and temporal conflation is related to eliminating differences between spatial datasets that occupy the same geographical zone at different times [5].

The level of automation applied for the conflation processes is also considered during the classification process. Some conflation processes are fully automatic, while some needs the intervention of humans to provide certain details for the conflation process.

The representation model, or the dealt geospatial data type is an important aspect considered during the classification of conflation techniques. Chen et al. [1] classified the conflation techniques based on the dealt geospatial data types into following three categories:

- Vector to vector data conflation: The integration of two road networks of different accuracy levels.
- Vector to raster data conflation: The integration of road network and imagery.
- Raster to raster conflation: The integration of raster maps and imagery.

Apart from these three categories as presented in Table 1.1, based on the representation model the conflation techniques can be further classified as conflation between an image and a DEM and between two DEMs. Conflation between an image and a DEM focuses on integrating the model capabilities of the DEM with the semantic and interpretive aspect of the image. Whereas, the conflation between two DEMs focuses on removing the gaps and discontinuities on height data to achieve a continuous representation of the land.

Having an accurate vector geographic database is an important aspect in GIS as it can benefit application domains such as crisis management applications, city traffic planning, military intelligence applications, navigation systems, resource management and urban planning [1, 6]. Thus, many GIS researches have been focused on vector to vector conflation in the past few decades [3, 4, 7-10]. However, due to the advancement of the remote sensing technology to capture high resolution imagery, vector to imagery conflation has become a central issue in the study of GIS [1, 2, 11], as this accurate imagery can be used to generate a more accurate vector geographic database.

1.2. Research Problem and Research Questions

When considering the existing road vector layer (vector data) with the actual road layer represented in raster data models (Satellite images), a misalignment of the two data models can be observed. These position misalignments/deviations can occur due to the different projections, different accuracy levels and different formats in geospatial data obtained from different data sources [1]. Figure 1.1 represents the above mentioned misalignment/deviation of existing road vector layers. The figure showcases the misalignment between existing road vector data of the Sri Lanka survey department and the road layer represented by the satellite image of Google maps in a scale of 1: 10,000. Black line represents the vector layer and the white lines in the satellite image represents the actual road layer. The arrows shown in yellow showcase the deviation of the vector layer from the actual road segment. Thus, it is visible that this misalignment is nonsystematic. Therefore, carrying out a global transformation is not possible.

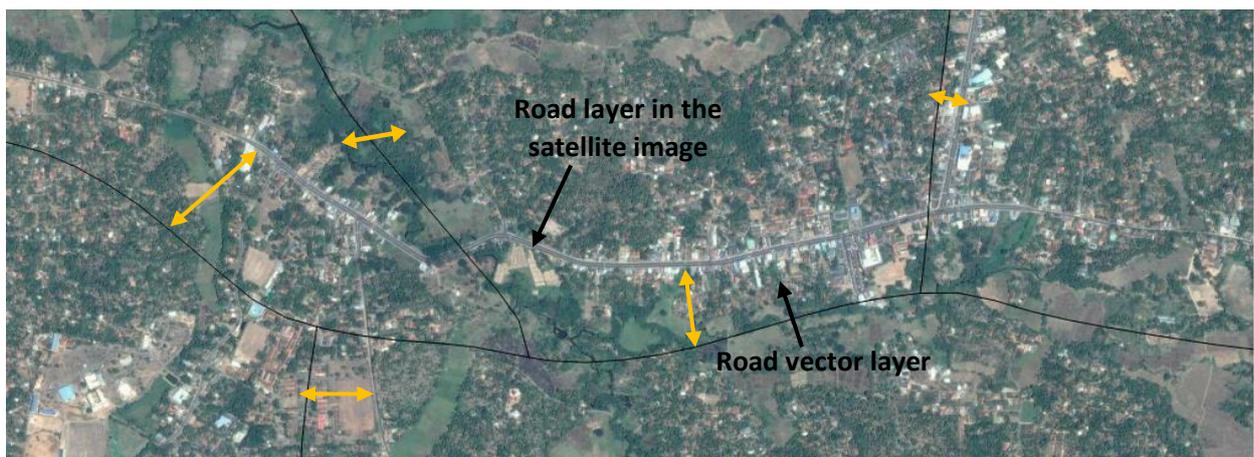


Figure 1.1: Misalignment between road vector layer and road layer in satellite imagery.

Considering this research problem, the generated research questions are as follows:

- What are the drawbacks of the existing approaches to refine the existing road vector layer?
- Which mechanism should be followed to align the road vector layer with the satellite image?
- What is the mechanism to evaluate the accuracy of the refined vector layer?
- Does this method provide required accuracy with respect to refinement?

1.3. Research Aim and Objectives

The main aim of the research is to propose an approach to refine the existing road vector layer to align with the actual road network layer of Google satellite imagery.

The objectives of the research are as follows:

- Analyze the drawbacks and limitations of the existing approaches.
- Devise a mechanism to align the road vector layer with the satellite image.
- Observe and evaluate whether devised mechanism is sufficient to obtain accurate road vector data.
- Identify a suitable evaluation mechanism to measure the success of the refinement process.

1.4. Justification for the Research

Vector to imagery conflation has become one of the central issues in the domain of GIS researches with the advancement of the remote sensing industry to capture high resolution satellite images [1, 2, 11]. In the domain of vector to imagery conflation, conflating road networks presented in vector format with the road networks in the satellite imagery plays an important role as having accurate road vector data benefit application domains such as crisis management applications, city traffic planning, military intelligence applications, navigation systems, resource management and urban planning [1, 6]. This conflation process can be divided into two main steps which are, identifying counterpart elements from the vector layer and satellite imagery using image processing techniques and utilizing an alignment algorithm to align the vector layer and satellite imagery using these counterpart elements. Many GIS researches have focused on the first step of this conflation process which focuses on identifying counterpart elements using image processing techniques. However, few researches have focused on proposing an alignment algorithm for the conflation process. The alignment algorithm plays a vital role with regard to the accuracy of the conflation process. Thus, this research focuses on proposing an alignment approach which will benefit the accuracy of the vector to imagery conflation process.

There is a lack of a generalized evaluation framework to evaluate vector to imagery conflation approaches in the domain of GIS researches. Therefore, this research also focuses on proposing an evaluation model that can be utilized to evaluate vector to imagery conflation approaches and road extraction approaches from satellite imagery.

1.5. Methodology

As the first step of the research approach the existing approaches to refine the road vector layer is analyzed and the key findings of those approaches that can be taken into consideration is identified. Devising a suitable refinement approach is considered as the next step of the proposed approach. The final step is focused on devising a mechanism to evaluate the accuracy of the refinement results. Figure 1.2 represents a high-level diagram of the proposed research methodology.

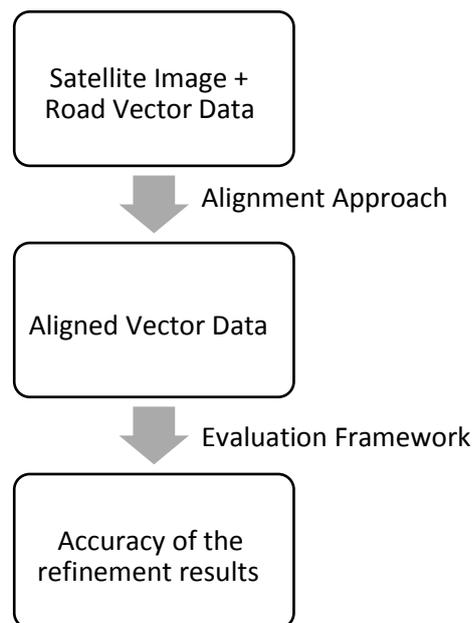


Figure 1.2: Proposed research methodology

1.6. Outline of the Dissertation

The dissertation is structured as follows. Chapter two explores the existing approaches related to the domain of vector to imagery conflation. Chapter three describes the proposed research

design and methodology. Potential ways of addressing the research problem is discussed in this chapter. Chapter four demonstrates the implementation details of the proposed methodology. Chapter five presents the evaluation model and the evaluation results of the proposed approaches. The last chapter, chapter six provides the conclusion of the thesis and outlines the future work.

1.7. Delimitations of Scope

An approach to refine the existing road vector data will be introduced using satellite images of Google maps. In order to carry out the refinement process the road layer presented in the satellite image will be considered to be accurate. The vector data from the Sri Lanka survey department will be used for this process. Several misalignment/deviation scenarios can be observed in the vector layer and a selected set of scenarios will be considered in this research. The best possible resolution (image scale) of the satellite image necessary for the refinement will be considered during the refinement process.

1.8. Summary

Integration or conflation of geospatial datasets that originate from different sources is an important aspect in GIS as it can provide capabilities that are not possible from a single dataset. However, since these geospatial data originate from different sources, misalignments/deviations can occur during the conflation process resulting in a requirement of an alignment approach to reduce the positional inconsistencies that occur. Therefore, when considering the existing road vector layer (vector data) with the actual road layer represented in raster data models (Satellite images), a misalignment of the two data models can be observed. Thus, the aim of the research is to propose an alignment approach to overcome this problem. The research objectives mainly focus on devising an alignment approach and finding an evaluation model to determine the accuracy of the approach. For this purpose, the Sri Lankan road vector data and the Google satellite image will be utilized, and the road layer presented in the satellite image will be considered accurate.

Chapter 2 - Literature Review

2.1. Introduction

In this chapter, a review of related work on vector to imagery conflation is provided. Vector to imagery conflation approaches consist of two main steps. The first step is to find accurate control point pairs and the second step involves in applying a conflation algorithm making use of the identified control point pairs. Thus, Section 2.2 focuses on the existing methods for identifying accurate control point pairs and Section 2.3 presents a review of the existing traditional conflation algorithms.

2.2. Approaches for Control Point Identification

To perform vector to imagery conflation, some spatial objects from the imagery should be extracted to serve as control points. Control points can be considered as counterpart elements of the vector dataset and imagery. These are the base points used to carry-out the refinement process. The existing vector to imagery conflation approaches have utilized different methods to extract these counterpart elements.

Hild and Fritsch [12] extracted polygons from the vector data and carried out image segmentation to identify image polygons. Then, a matching procedure is carried out to calculate the polygon statistics from both vector dataset and the segmented satellite imagery. Thus, in this approach the polygonal features are considered as control points. However, to get a successful matching between the two datasets, existence of polygonal features is essential in this approach. Filin and Doytsher [13] introduced a linear transformation algorithm to align vector data with imagery. In this approach, initially all edges from the image are detected and converted to the vector format. Then, vector data is matched against the edges to identify real road edges. The vector data with no matched edges are transformed based on the influence regions formed by the matched edges. However, since the road edges are considered as control points, it suffers from the difficulties of extracting features from the imagery and converting to vector format.

Various GIS researchers and computer vision researchers have shown that, the intersection points on the road networks are good candidates to be identified as an accurate set of control points, as road intersections are salient points to capture the major layouts of the road network and the road shapes around the road intersections are well defined [14]. Thus, several approaches have been carried out to identify road intersections from the imagery and vector dataset for the conflation process.

Thakkar et al. [11] carried out a localized image processing technique (intersection points and the road directions from the vector data are utilized to identify the corresponding intersection points on the satellite image) to identify road intersections. These intersection points and online sources are used as control points for the conflation process. Further, the inaccurate control points are filtered out using a vector median filter. However, road segments in the satellite image near the intersection point of the vector data should be extracted in this technique which is a computationally intensive task. This approach was modified by Knoblock et al. [1] with the introduction of a histogram based classifier to more accurately identify road intersections as control points and improved the localized image processing technique by exploiting road vector direction and widths to generate templates to match against the satellite imagery. This approach can only refine vector layer misalignments/deviations within a limited range as the run time increases when the search area size increases for the template matching. Another main issue with template matching approaches is that only limited shape models can be provided. It is possible for the angle and road branch width to vary significantly even for one type of intersections in an image, and this may cause matching issues. Also, partial occlusion near intersection points will cause problems for the matching process. However, this template matching approach is more efficient than extracting the road segments from the imagery to identify intersections.

Keller et al. [2] utilizes road intersections and road termination points as control points for conflation by automatically detecting and classifying different types of road intersections and terminations on spatial contextual measures and matching them with the vector layer using a relaxation labeling algorithm. However, this spatial contextual extraction process utilizes a considerable amount of computation time.

Fernando [20] proposed an approach to refine the road vector layers using the blue band feature for road feature extraction. The blue band feature is taken into consideration to remove the

vegetation covers from the road structure. Further, erosion is carried out to remove noise from the image and connected component labeling to identify the road segment from the image. In this approach, points intersecting with the extracted road layer and the vector layer are considered as control points. This approach also focuses on extracting the road segment from the image which is associated with a high computation cost.

Silva [21] updates the existing road vector layers by initially identifying the road pixels using a pixel grouping mechanism. After identification of the road segment, Harris Corner detection approach is used to extract the coordinate points of the identified road segment. In this approach to remove noise they have considered that the noise in the image is spread in a similar manner. However, Harris Corner detection mechanism is computationally very expensive thus requires considerable amount of time for control point identification. However, using the Harris Corner detection and identifying coordinates of the road is not necessary if the existing vector layer information has been utilized in an efficient manner, as the objective of the research was to refine the existing vector layer and not introduce a completely new vector dataset.

Rupasinghe [22] proposed an approach to align the road vector layer initially by identifying intersecting points of the road using Hough transformation based techniques. After the identification of intersecting points in the imagery, a localized template matching is carried out to identify the corresponding road intersect point in the raster layer. However, carrying out Hough transformation for a satellite image may result in a lower precision, as there can be a high number of false points in the imagery identified as actual road intersections. Further, the approach also suffers from the same limitations associated with template matching mechanisms as mentioned above.

2.3. Traditional Conflation Algorithms

The main step of the vector to imagery conflation techniques is the refinement process using a conflation algorithm. However, only two main algorithms such as Piecewise Linear Rubber-Sheeting algorithm and Snakes algorithm have been exploited in most of the vector to imagery conflation approaches. Thus far, the main difference between vector to imagery conflation approaches have been the different mechanisms proposed for the identification of control

points. In this section, a review on these two main algorithms will be presented along with the modifications proposed to these algorithms by different researchers.

2.3.1. Piecewise Linear Rubber-Sheeting

The piecewise linear rubber-sheeting technique for map transformation was introduced by White and Griffin [15]. This algorithm has been utilized by many researchers to align the vector dataset with the imagery after the control point identification process [1, 2, 11, 13, 16]. The algorithm assumes the map to be transformed as a rubber-sheet which is stretched to coincide with the stable map. This process consists of two main steps. The first step is the triangulation process which divides the rubber-sheet map (map to be transformed) into triangles based on the provided control points. In this triangulation process it is important to avoid long and narrow triangles. Thus, a quadrilateral test is carried out to ensure the triangulation maximizes the minimum triangle height. The second step is the transformation process which maps the individual triangles in the rubber-sheet map onto its corresponding triangle on the stable map using only an affine transformation. The affine transformation is carried out for each triangle j by generating a transformation matrix T_j . For a triangle j , if the vertices of the rubber-sheet map are u_k, u_l, u_m and vertices of the stable map are v_k, v_l, v_m the transformation matrix T_j is generated as shown below.

$$v_k = T_j u_k \quad (1)$$

$$v_l = T_j u_l \quad (2)$$

$$v_m = T_j u_m \quad (3)$$

The vector notation of the equations (1) to (3) are expanded by computing,

$$\begin{bmatrix} v_x \\ v_y \\ 1 \end{bmatrix} = \begin{bmatrix} t_{11} & t_{12} & t_{13} \\ t_{21} & t_{22} & t_{23} \\ t_{31} & t_{32} & t_{33} \end{bmatrix} * \begin{bmatrix} u_x \\ u_y \\ 1 \end{bmatrix} \quad (4)$$

Where v_x, u_x are the x coordinates of the vertices and v_y, u_y are the y coordinates. By expanding the equation (4), the following (5), (6), (7) equations can be derived for each vertex of the triangle.

Thus, resulting nine equations for one triangle. By solving the nine equations the transformation matrix T_j will be derived.

$$v_x = t_{11} * u_x + t_{12} * u_y + t_{13} \quad (5)$$

$$v_y = t_{21} * u_x + t_{22} * u_y + t_{23} \quad (6)$$

$$1 = t_{31} * u_x + t_{32} * u_y + t_{33} \quad (7)$$

Finally, any element x inside triangle j in the rubber-sheet map is transformed based on equation (9) where x' is the new location of element x .

$$x' = T_j * x \quad (9)$$

Pieceswise rubber-sheeting method based on triangles with extremely small angles (long and thin triangles) results in contorted conflation. Thus, Thakkar et al. [11] and Knoblock et al. [1] performed Delaunay triangulation [17] to avoid triangles with extremely small angles. A Delaunay triangulation is a triangulation of the points with the property that no point falls in the interior circumcircle of any triangle (the circle passing through the three triangle vertices). This property is known as the "INCIRCLE" property. Thus, Delaunay triangulation maximizes the minimum angle of all angles in the triangulation and avoiding triangles with extremely small angles.

In small regions of the selected control points, the results of the rubber-sheeting approach will be quite good, because it forces those points to coincide, but at places far from the control points, misalignments/deviations may remain. These misalignments/deviations can also be refined by adding more control points with the cost of making the process more complicated. Further, this approach is limited in coping with distortions between sets, since only the seed points are matched and the relative disagreements between the linear features shapes remain unresolved. However, since this approach is considered with generating triangles based on the control points, it is necessary to have a sufficient number of control points to generate the triangles. The computation cost of the algorithm is associated with the triangulation and transformation processes. The triangulation process can build in $O(n \log n)$ worst case time complexity, where n is the number of control points. Thus, it is visible that the time complexity increases with the number of control points.

2.3.2. Snake Related Algorithms

The active contour model [18] (Snakes-related algorithm) is another approach utilized to align the vector dataset (road vector layer) to the corresponding features (road edges) in the imagery. Many vector to imagery conflation researches [2, 19, 22] has used the active contour model for the conflation process.

Snakes (active contour model) is a parametric curve and this curve is modeled by linking the identified multiple control points. Snakes evolve their shape by shifting the control points towards the image features (road edges) and maintaining the smoothness at the same time. The concept behind the process of evolution is the principle of energy (internal and external energy) minimization. The internal energy is concerned with geometric constraints (length and the smoothness of the snakes) while the external energy is concerned with pushing the snakes towards the image features. Thus, by minimizing the internal and external energy the geometric properties are preserved while moving the snakes towards the image features. This method requires some seed points close to the real road segments to start the evolution.

The following weaknesses can be observed in the active contour models:

1. This is a greedy algorithm. Thus, requires a lot of calculations when trying all local solutions to find the optimal solution.
2. It is possible for the snakes to attached to noisy pixels, which will prevent the snakes from converging to real road edges.
3. The initial seed points play a vital part because if the initial points are not accurate the snakes will diverge.
4. The evolution of the snakes will be harmed if the original vector layer data are inconsistent with the real roads.
5. The active contour models are also possible to deform the road shape of the vector layer.

Keller et al. [2] combined the rubber-sheeting approach and the snakes method to minimize the above mentioned weaknesses of the snakes approach. Initially, the misaligned vector layer is transformed/aligned using the rubber-sheeting algorithm. Control points of the vector layer are required to be close to the road segment to obtain a desired alignment in the snakes method. Thus, carrying the rubber-sheeting transformation before the snakes method will solve this problem.

However, as stated above these two conflation algorithms consists of both advantages and disadvantages that directly impacts the vector to imagery conflation process. Thus, the main focus of this research will be on proposing a conflation algorithm taking into consideration the strengths and limitations of the existing traditional conflation algorithms.

2.4. Summary

This chapter mainly focused on providing an extensive review on the existing vector to imagery conflation approaches regarding road networks. Initially, it discussed the two main steps associated with vector to imagery conflation approaches, which is identifying a set of counterpart elements from the vector layer and the imagery and applying a conflation algorithm based on these counterpart elements. Different approaches proposed by GIS researchers on these two main steps were discussed along with the advantages and disadvantages of the proposed approaches.

Chapter 3 - Design

3.1. Introduction

This chapter elaborates the proposed solutions to the research problem. It consists of three major sections namely research design, alignment of road vector layer considering distance ratio and angle and alignment of road vector layer considering Bézier curves.

3.2. Research Design

The research design encompasses three main steps: Coordinate Reference System (CRS) Transformation, Region Selection, and Refinement. Figure 3.1 represents a high-level diagram of the proposed research design.

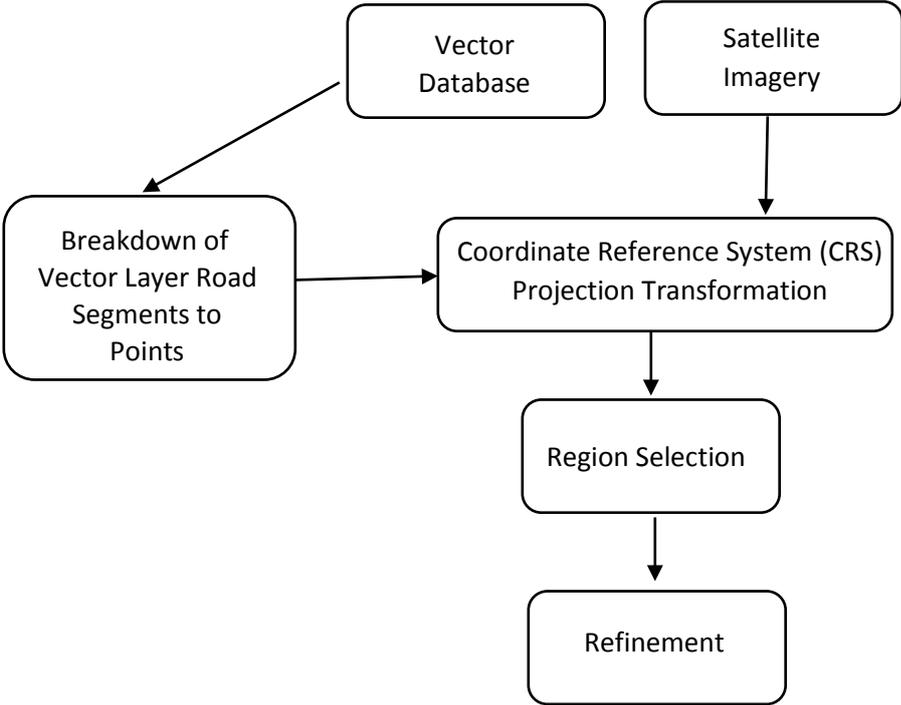


Figure 3.1: Research Design

Initially, the existing vector layer database of the Sri Lankan road data will be decomposed into a set of points (a set of points will represent each road segment in the vector database). Therefore, transforming these points will result in a transformation of the road layer. Secondly, these data

points and the satellite image will be transformed to a single CRS as this transformation is important to see the misalignments/deviations between the two geospatial datasets. The focus is to carryout a local adjustment/refinement for each region. Thus, a segmented region will be considered for the refinement process. Figure 3.2 represents a high level diagram of the main components of the refinement process.

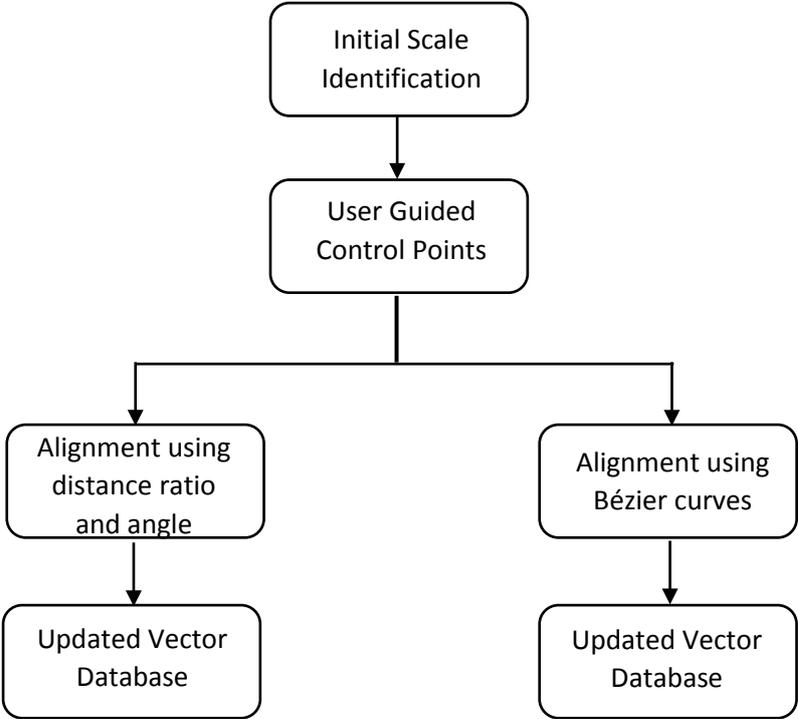


Figure 3.2: Diagram of the refinement process

The refinement process encompasses several important steps. The first step would be the identification of a suitable scale to start the refinement. Scale identification is an essential aspect of the refinement process, as when the scale reduces the misalignment/deviation of the vector layer can be clearly visible. After identification of the initial scale, a set of control points from the vector layer and the satellite image will be provided. The control points from the vector layer and the satellite image can be junction points or road termination (end) points. Based on the provided control points the refinement for the road vector layer can be carried out in two different approaches, considering the distance ratio and angle or considering Bézier curves. The final output after the refinement/alignment will be an updated road vector database.

3.3. Alignment of road vector layer considering distance ratio and angle

Alignment of the road vector layer considering distance ratio and angle is carried out with the use of four control points (two control points from the vector layer and the corresponding two points of the real road in the satellite image). These control points can be road junction points or road end points. Each road layer will be represented with a set of vector points once the vector layer is broken down to points as the second step of the research design. The objective will be transforming these vector points based on the provided control points to coincide with the real road in the satellite image.

The transformation of each point in a road vector layer is carried out as follows:

Consider $V1 (x_1, y_1)$, $V2 (x_2, y_2)$ as the provided vector layer control points and $R1 (x'_1, y'_1)$, $R2 (x'_2, y'_2)$ as the corresponding control points in the satellite image for a road segment to be refined.

For every point P in the road segment:

- 1) Calculate the angle (α) between the three points P, V1, V2.

The angle can be calculated from equations (1) to (3) considering the triangles V1 T1 P and T2 V1 V2 shown in Figure 3.3. (T1 is the intersecting point of the vertical line going through V1 and the horizontal line going through P. Similarly, T2 is the intersecting point of the vertical line going through V1 and the horizontal line going through V2. Angles β and θ are measured in the anti-clockwise direction from negative y-axis)

$$\tan\theta = \frac{(x_2 - x_1)}{(y_1 - y_2)} \quad (1)$$

$$\tan\beta = \frac{(x - x_1)}{(y_1 - y)} \quad (2)$$

$$\alpha = \theta - \beta \quad (3)$$

- 2) Calculate the new distance (r') from R1 (shown in Figure 3.4) as follows:

$$r' = \frac{D * r}{d} \quad (4)$$

Where D is the distance between R1 and R2, d is the distance between V1 and V2 and r is the distance between V1 and P.

- 3) Calculate the new coordinates of the point P (new location $P'(x', y')$) as shown in Figure 3.4) with the distance r' and angle α with respect to R1 and R2 as follows (Angle γ is measured in the anti-clockwise direction from negative y-axis):

$$\tan\gamma = \frac{(x'_2 - x'_1)}{(y'_1 - y'_2)} \quad (5)$$

$$x' = x'_1 + r' \sin(\gamma - \alpha) \quad (6)$$

$$y' = y'_1 - r' \cos(\gamma - \alpha) \quad (7)$$

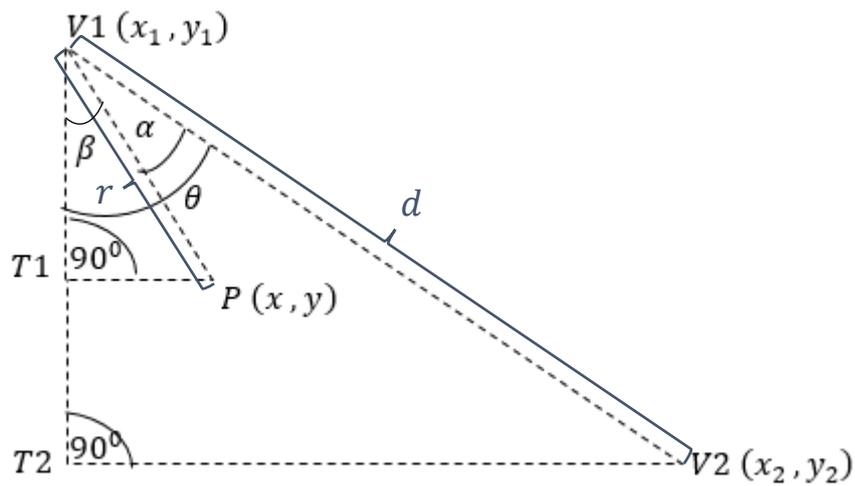


Figure 3.3: The geometry representation for any given point P in the vector layer.

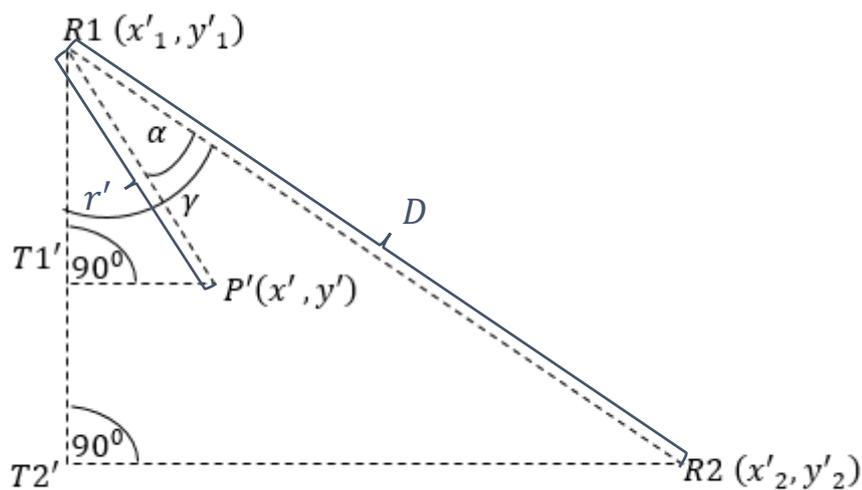


Figure 3.4: New location P' for any given point P in the vector layer with respect to R1 and R2.

3.4. Alignment of road vector layer considering Bézier curves

The next proposed solution for the refinement process of the road vector layer is alignment of the road vector considering Bézier curves. Bézier curves are parametric curves generated using linear interpolation, which are generally used in the domain of computer graphics and related fields [23, 24]. A Bézier curve is defined by a set of control points P_0 to P_n and the number of control points define the order of the curve ($n = 1$ for a linear curve, $n = 2$ for quadratic curve etc.). Thus, an order n bézier curve is defined by $n+1$ control points. The first and the last point control points represent the end points of the curve while the other intermediate control points contribute to shape of the curve. It is important to note that the intermediate control points do not lie on the curve. Figure 3.5 represents the generation of a quadratic bézier curve with the control points $P_0: (70, 250)$, $P_1: (20, 110)$, $P_2: (250, 60)$.

The formula for an order n bézier curve can be expressed as below:

$$B(t) = \sum_{i=0}^n \binom{n}{i} (1-t)^{n-i} t^i P_i \quad \text{where } 0 \leq t \leq 1$$

Here $\binom{n}{i}$ are the binomial coefficients and P_i are the control points.

In this approach, we consider a road segment to be represented as an order two or order three bézier curve. Thus, considering a road segment is controlled by three or four bézier control points.

The transformation of the road vector layer points considering an order two bézier curve will be carried out as follows:

- 1) Consider $V1(x_1, y_1)$, $V2(x_2, y_2)$ as the provided vector layer control points and $R1(x'_1, y'_1)$, $R2(x'_2, y'_2)$ as the corresponding control points in the satellite image of the road segment to be refined.
- 2) Identify the points P_0, P_2 corresponding to $t = 0$ and $t = 1$ which are the start ($V1$) and end points ($V2$) of the road segment (These two points will act as the start and end control points for the bézier curve.).

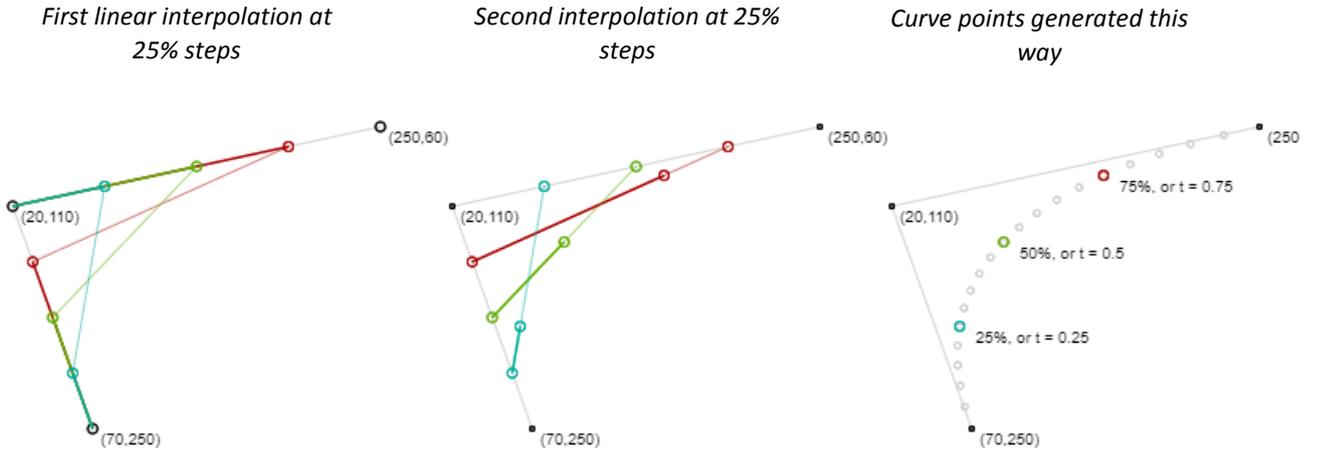


Figure 3.5: Generation of a quadratic bezier curve as illustrated in [24].

- 3) Assign t values to the intermediate points in the road segment assuming that points are uniformly distributed. ($0 < t < 1$)
- 4) Considering the road segment to be a quadratic bézier curve, the following equation (8) will represent the road segment.

$$B(t) = P_0(1 - t)^2 + 2P_1t(1 - t) + P_2t^2 \quad (8)$$

- 5) Next, find the remaining bézier control point P_1 . (coordinates of P_0 and P_2 are already known from step two as these are the vector layer control points). Therefore, from equation (8) we can retrieve the below equation (9):

$$P_1 = \frac{B(t) - P_0(1 - t)^2 - P_2t^2}{2t(1 - t)} \quad (9)$$

Consider any the situation when $t = l$ and $B(l) = \begin{pmatrix} l_x \\ l_y \end{pmatrix}$. From step 3 we have already assigned t values for the vector layer points. Therefore, $B(l) = \begin{pmatrix} l_x \\ l_y \end{pmatrix}$ is already known. (l_x is the x coordinate of the vector layer point when $t = l$ and l_y is the y coordinate of the vector layer point when $t = l$)

Thus, x and y coordinates P_{1x} and P_{1y} of control point P_1 can be derived from equation (10) and (11). Where P_{0x}, P_{0y} are the x and y coordinates of point P_0 . Similarly, P_{2x}, P_{2y} are the x and y coordinates of point P_2 .

$$P_{1x} = \frac{l_x - P_{0x}(1-l)^2 - P_{2x}l^2}{2l(1-l)} \quad (10)$$

$$P_{1y} = \frac{l_y - P_{0y}(1-l)^2 - P_{2y}l^2}{2l(1-l)} \quad (11)$$

- 6) Find the new location P'_1 of P_1 considering relative distance if P_0 is mapped to R_1 and P_2 is mapped to R_2 . This transformation will yield to transforming the road vector layer points.
- 7) Finally, find the new coordinates of the vector layer points considering the assigned t values in step 3 and P'_1, R_1 and R_2 and applying to the below equation (12) which represent the transformed curve (road segment):

$$B(t) = R_1(1-t)^2 + 2P'_1t(1-t) + R_2t^2 \quad (12)$$

The transformation of the road vector layer points considering an order three bézier curve will be carried out as follows:

- 1) Consider $V1(x_1, y_1), V2(x_2, y_2)$ as the provided vector layer control points and $R1(x'_1, y'_1), R2(x'_2, y'_2)$ as the corresponding control points in the satellite image of the road segment to be refined.
- 2) Identify the points P_0, P_3 corresponding to $t = 0$ and $t = 1$ which are the start ($V1$) and end points ($V2$) of the road segment (These two points will act as the start and end control points for the bézier curve.).
- 3) Assign t values to the intermediate points in the road segment assuming that points are uniformly distributed. ($0 < t < 1$)
- 4) Considering the road segment to be a cubic bézier curve, the following equation (13) will represent the road segment.

$$B(t) = P_0(1-t)^3 + 3P_1t(1-t)^2 + 3P_2t^2(1-t) + P_3t^3 \quad (13)$$

5) Next, find the remaining bézier control points P_1, P_2 . (coordinates of P_0 and P_3 are already known from step two as these are the vector layer control points). Consider two situations when $t = t_1$, $t = t_2$ and $B(t_1) = \begin{pmatrix} t_{1x} \\ t_{1y} \end{pmatrix}$, $B(t_2) = \begin{pmatrix} t_{2x} \\ t_{2y} \end{pmatrix}$. From step 3 we have already assigned t values for the vector layer points. Therefore, $B(t_1)$ and $B(t_2)$ is already known. (t_{1x}, t_{2x} are the x coordinates of the vector layer points when $t = t_1$ and $t = t_2$ respectively and t_{1y}, t_{2y} are the y coordinates of the vector layer points when $t = t_1$ and $t = t_2$ respectively.)

Thus, x and y coordinates P_{1x} and P_{1y} of control point P_1 and x and y coordinates P_{2x} and P_{2y} of control point P_2 can be derived by solving equations (14) and (15). Where P_{0x}, P_{0y} are the x and y coordinates of point P_0 . Similarly, P_{3x}, P_{3y} are the x and y coordinates of point P_3 .

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ (1-t_1)^3 & 3t_1(1-t_1)^2 & 3t_1^2(1-t_1) & t_1^3 \\ (1-t_2)^3 & 3t_2(1-t_2)^2 & 3t_2^2(1-t_2) & t_2^3 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} P_{0x} \\ P_{1x} \\ P_{2x} \\ P_{3x} \end{bmatrix} = \begin{bmatrix} P_{0x} \\ t_{1x} \\ t_{2x} \\ P_{3x} \end{bmatrix} \quad (14)$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ (1-t_1)^3 & 3t_1(1-t_1)^2 & 3t_1^2(1-t_1) & t_1^3 \\ (1-t_2)^3 & 3t_2(1-t_2)^2 & 3t_2^2(1-t_2) & t_2^3 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} P_{0y} \\ P_{1y} \\ P_{2y} \\ P_{3y} \end{bmatrix} = \begin{bmatrix} P_{0y} \\ t_{1y} \\ t_{2y} \\ P_{3y} \end{bmatrix} \quad (15)$$

6) Find the new location P'_1, P'_2 of P_1, P_2 respectively, considering relative distance if P_0 is mapped to R_1 and P_3 is mapped to R_2 . This transformation will yield to transforming the road vector layer points.

7) Finally, find the new coordinates of the vector layer points considering the assigned t values in step 3 and P'_1, P'_2, R_1 and R_2 and applying to the below equation (16) which represent the transformed curve (road segment):

$$B(t) = R_1(1-t)^3 + 3P'_1t(1-t)^2 + 3P'_2t^2(1-t) + R_2t^3 \quad (16)$$

3.5. Summary

This chapter provided a detailed description on the research design and the two main approaches considered in this study to align the road vector layer. The research design encompasses three main steps which are CRS transformation, region selection and refinement. The refinement process considered two approaches to align the road vector layer. These two approaches were, alignment of road vector layer using distance ratio and angle, and alignment of the road vector layer considering bézier curves. A detailed description of the main steps of these two approaches were provided towards the end of the chapter.

Chapter 4 - Implementation

4.1. Introduction

This chapter provides the implementation details of the proposed solutions. Section 4.2 describes the software tools utilized for implementation, Section 4.3 presents the implementation details of the alignment of road vector layer considering distance ratio and angle and Section 4.4 presents the implementation details of the alignment of the road vector layer considering bézier curves.

4.2. Software Tools

The proposed solutions were implemented using python 2.7, QGIS software and PostgreSQL database management system with PostGIS (a spatial database extender for PostgreSQL). Python 2.7 was used to develop scripts for the transformation of coordinates of the road vector layer points. Displaying the satellite image and road vector layer and retrieving the coordinates of the road vector layer points were carried out using the QGIS software. PostgreSQL database with PostGIS was utilized as it is compatible with QGIS and is also capable of manipulating geospatial data. The database was used to store road vector layer points and to calculate the polygon area (which will be described in Chapter 5) for evaluation purposes.

4.3. Implementation Details - Alignment of road vector layer considering distance ratio and angle

In this approach, as mentioned above in Chapter 3, each vector layer point will be transformed based on the angle and the distance from the control points provided.

As shown below the function 'angle' calculates the angle between the given three points and the function 'find_dis' calculates the distance between two given points.

```

def angle(p1,p2,p3):
    math.atan2(float(p2[1])-float(p1[1]),float(p2[0])-float(p1[0]))
    math.atan2(float(p2[1])-float(p3[1]),float(p2[0])-float(p3[0]))
    angleDegrees = (angle1-angle2)
    print angleDegrees*360/(math.pi*2)
    return angleDegrees

def find_dis(p1,p2):
    math.sqrt((abs(float(p1[0])-float(p2[0])))**2+(abs(float(p1[1])-
float(p2[1])))**2)
    return d

```

The function 'get_point' will find the new (x,y) coordinates of a given vector layer point based on the two control points of the real road in satellite image and the calculated angle and distance.

```

def get_point(p1,p2,theta,r):
    angle1 = math.atan2(
        float(p2[1])-float(p1[1]),float(p2[0]- float(p1[0]))
    )
    print angle1
    print r*math.cos(theta+angle1)
    x = float(p1[0])+r*math.cos(theta+angle1)
    y = float(p1[1])+r*math.sin(theta+angle1)
    return x,y

```

Finally, the function 'new_road_Points' will transform all the road vector layer points based on the control points provided. Here, the parameter vector is an array of all vector layer points (first and last element of this array are the control points of the vector layer) and the parameter points is an array of length two which will have the corresponding control points of the real road segment in the satellite image.

```

def new_road_Points(vector,points):
    for p in vector[2:-1]:
        theta = angle(p,vector[1],vector[-1])
        d1 = find_dis(points[1],points[2])*find_dis(vector[1],p)
        d2 = find_dis(vector[1],vector[-1])
        r = d1/d2
        x,y = get_point(points[1],points[2],theta,r)
        plt.plot(x,y,'go')
        points.append([x,y,'X','Y'])

```

4.4. Implementation Details - Alignment of road vector layer considering Bézier curves

In this approach, as mentioned above in Chapter 3, each vector layer point will be transformed by considering the road vector layer to be an order two or order three Bézier curve. The implementation details for a road segment considering an order two bézier curve is mentioned below.

The following function 'assign_tval' showcase the process of assigning t values for each of the vector layer points. It is assumed that the points are equally distributed along the road segment.

```
def assign_tval(vector):
    tval=[]
    for i in range(0,len(vector)-1):
        val = i * (1.0/(len(vector)-2))
        tval.append(val)
    return tval
```

To find the coordinates of the bézier control point (P_1) of the curve considering the start and end points of the curve (control points of the vector layer) and the coordinates of any point at a given t value, the below function is used. For this purpose, the coordinates and the t value of the middle point of the vector layer points is considered. The function 'bezier_equation' illustrated below will calculate the x, y coordinates of the P_1 control point.

```
def bezier_equation(p0,p2,t1,l):
    array=[]
    plx = (float(t1[0])-(float(p0[0])*(1**2))
           -(float(p2[0])*((1-1)**2)))/(2*1*(1-1))
    ply = (float(t1[1])-(float(p0[1])*(1**2))
           -(float(p2[1])*((1-1)**2)))/(2*1*(1-1))
    array.append(plx)
    array.append(ply)
    return array
```

The below function 'new_Control_Point' generates the new control point considering the relative distance between the current control points of the curve and the real road control points of the satellite image. (Finds the new location of P_1)

```

def new_Control_Point(p,vector,points):
    theta = angle(p,vector[1],vector[-1])
    d1= (find_dis(points[1],points[2])*find_dis(vector[1],p))
    d2 = (find_dis(vector[1],vector[-1]))
    r = d1/d2
    x,y = get_point(points[1],points[2],theta,r)
    plt.plot(x,y,'bo')
    return x,y

```

Finally, the function 'new_Points' calculates the new location of the vector layer points based on the new control points of the curve.

```

def new_Points(tval,p0,p1,p2):
    newpoints = [['X','Y']]
    for t in tval:
        x = (float(p0[0])*(t**2))+(2*t*(1-t)*float(p1[0]))
            +(float(p2[0])*((1-t)**2))
        y = (float(p0[1])*(t**2))+(2*t*(1-t)*float(p1[1]))
            +(float(p2[1])*((1-t)**2))
        plt.plot(x,y,'g+')
        newpoints.append([x,y])
    return newpoints

```

The implementation details for a road segment considering an order three bézier curve is illustrated below.

The below function 'bezier_equation' will calculate the coordinates of the two bézier control points P_1 and P_2 of the curve considering the start and end points of the curve (control points of the vector layer) and the coordinates of two points at two different t values. For this purpose, the coordinates and the corresponding t values of two points are considered. The two points are picked as 1/4 and 3/4 elements of an ordered list of vector layer points of a road segment. 'array1' will return the coordinates of P_1 control point and 'array2' will return the coordinates of P_2 control point of the curve.

```

def bezier_equation(p0,p3,b1,b2,t1,t2):
    array1=[]
    array2=[]

    # define matrix A
    A = np.array([[1, 0, 0, 0],
                  [(1-t1)**3, 3*t1*(1-t1)**2, 3*t1**2*(1-t1),t1**3],
                  [(1-t2)**3, 3*t2*(1-t2)**2, 3*t2**2*(1-t2),t2**3],
                  [0, 0, 0, 1]])

    #define matrix B
    Bx = np.array(
        [float(p0[0]), float(b1[0]), float(b2[0]),float(p3[0])])
    By = np.array(
        [float(p0[1]), float(b1[1]), float(b2[1]),float(p3[1])])

    array1.append(np.linalg.solve(A, Bx ) [1])
    array1.append(np.linalg.solve(A, By ) [1])
    array2.append(np.linalg.solve(A, Bx ) [2])
    array2.append(np.linalg.solve(A, By ) [2])
    return array1,array2

```

The function 'new_Control_Point' generates the new control points considering the relative distance between the current control points of the curve and the real road control points of the satellite image. (Finds the new location of P_1 and P_2)

```

def new_Control_Point(p,vector,points):
    theta = angle(p,vector[1],vector[-1])
    d1 =(find_dis(points[1],points[2])*find_dis(vector[1],p))
    d2 = (find_dis(vector[1],vector[-1]))
    r =d1/d2
    x,y = get_point(points[1],points[2],theta,r)
    plt.plot(x,y,'bo')
    return x,y

```

Finally, the function 'new_Points' calculates the new location of the vector layer points based on the new control points of the curve.

```

def new_Points(tval,p0,p1,p2,p3):
    newpoints = [['X','Y']]
    for t in tval:
        x = (float(p3[0])*(t**3))+(3*t**2*(1-t)*float(p2[0]))
            +(3*t*(1-t)**2*float(p1[0]))+(float(p0[0])*((1-t)**3))
        y = (float(p3[1])*(t**3))+(3*t**2*(1-t)*float(p2[1]))
            +(3*t*(1-t)**2*float(p1[1]))+(float(p0[1])*((1-t)**3))
        plt.plot(x,y,'g+')
        newpoints.append([x,y])
    return newpoints

```

The evaluation model (will be described in Chapter 5) to measure the success of the alignment approaches was implemented as mentioned below.

Initially, a table was created in the PostgreSQL (with PostGIS) database to store the coordinates of the points that will generate a polygon. These points will be the points of the real road in the satellite image (ground truth data) and the points of the vector layer. The below code showcases this process.

```
CREATE TABLE roadArea(  
  id varchar(10),  
  geom geometry(POLYGON,5235)  
)
```

The value 5235 in the above code indicates that the CRS of the points of the polygon is the Sri Lankan CRS.

To calculate the area of the polygon the required points are inserted to the above created table. Shown below is an example of this process for one road segment.

```
INSERT INTO roadArea(id,geom) VALUES  
 ('road1',ST_GeomFromText('POLYGON((402151.4746 477598.2697,  
402153.5357 477647.5966,  
402154.7906 477687.1371,  
402151.7411 477743.5205,  
402146.7995 477842.5821,  
402133.5995 477951.4464,  
402115.6228 478034.4797,  
402084.6743 478135.9376,  
402043.4669 478230.7586,  
401985.7288 478337.3551,  
401955.4919 478396.92,  
401945.1411 478402.6792,  
401990.869 478305.8821,  
402037.8363 478192.8568,  
402079.7698 478077.9169,  
402114.1914 477960.4737,  
402135.3386 477839.9255,  
402143.3278 477717.6287,  
402148.6601 477603.431,  
402151.4746 477598.2697  
))',5235))
```

Finally, to calculate the area of the polygon covering the inserted points is calculated as shown below.

```
SELECT id, ST_Area (geom) from roadArea;
```

4.5. Summary

In this chapter, the software tools utilized to implement the proposed solution was elaborated followed by the important functionalities of the two proposed solutions. The main functionalities of the alignment of vector layer considering distance ratio and angle are finding the angle between points, calculating the distance, calculating the new coordinates for a given point and finally, transforming all the vector layer points given. The main functionalities of the second solution are assigning t values, finding the control points that generates the curve based on the order of the curve, finding the new location of the control points based on relative distance, and finally, transforming all the given vector layer points. The implementation details of the evaluation model (will be described in Chapter 5) was described towards the end of the Chapter.

Chapter 5 - Results and Evaluations

5.1. Introduction

This chapter elaborates how results are evaluated and the success level of the proposed solutions. There is no generalized evaluation model that can be utilized for evaluating GIS researches focused on vector to imagery conflation. Therefore, a new evaluation model which can be utilized for evaluating conflation approaches in the domain of GIS is proposed. Section 5.2 elaborates on this proposed evaluation model. Section 5.3 describes the results obtained for the proposed solutions utilizing the evaluation model.

5.2. Evaluation Model

To evaluate the accuracy of the alignment/refinement approaches it is required to have the ground truth data of the real road layers (Manually marked real road segments). The ground truth data and the aligned road vector layers can be viewed as a polygon when combined as shown in Figure 5.1.

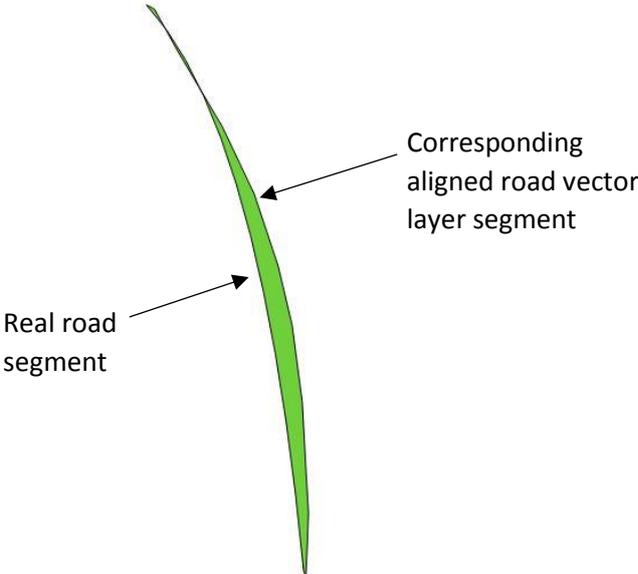
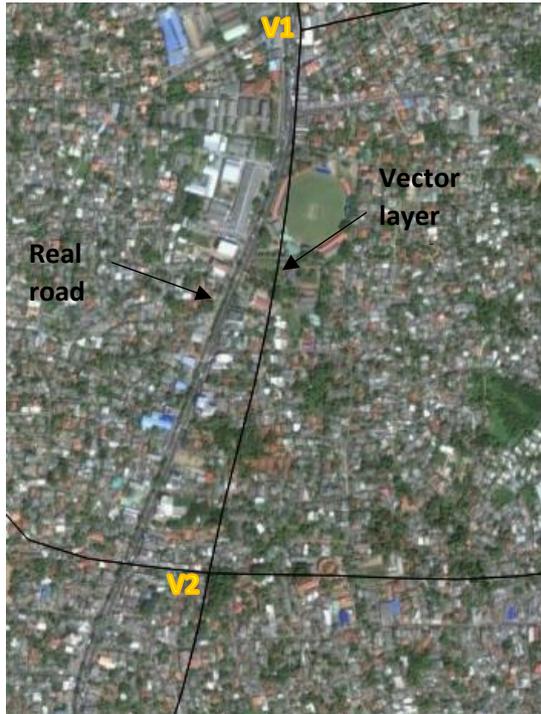


Figure 5.1: Polygon generated between the real road and the aligned road. (The figure is generated using QGIS software)

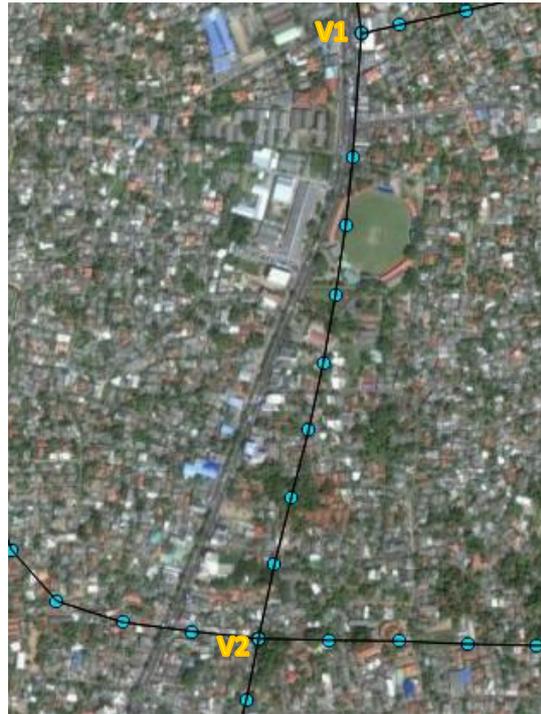
The area of this polygon can be considered to measure the accuracy of the alignment approach. If the area of the polygon is zero it emphasizes the aligned road has coincided with the real road. Thus, showcasing 100% accuracy. Therefore, if the area of the polygon is less it showcases higher accuracy of the refinement while having a high value for the polygon area showcases lower accuracy of the refinement. This evaluation model has been utilized to measure the accuracy of the proposed solutions in this study.

5.3. Results

The road vector layer database of the Sri Lankan survey department and the satellite imagery of Google maps was utilized for evaluation. The real road segments of the satellite image were manually marked in order to prepare the ground truth data. The satellite image and the vector data were transformed to the Sri Lankan CRS. An image scale of 1: 3000 was considered to identify the misalignments between the road vector data and the road layer in the satellite image. Figure 5.2 (a)-(f) showcases the results of the main steps associated with the first proposed approach, alignment of road vector layer considering distance ratio and angle for a single road segment. Similarly, Figure 5.3 (a)-(h) showcases the main steps associated with the second proposed approach, alignment of road vector layer considering bézier curves of order two and Figure 5.4 (a)-(h) showcases the results of the main steps associated with the alignment of road vector layer considering bézier curves of order three.



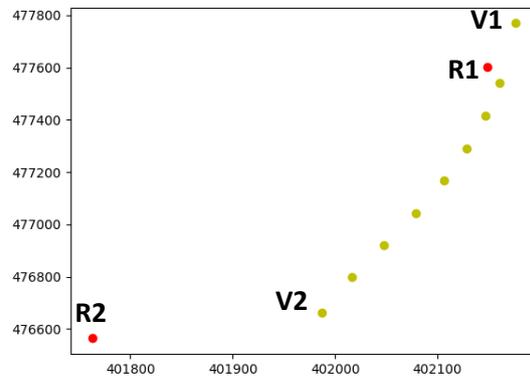
(a) Initial road vector layer V1, V2 with the misalignment.



(b) Initial road vector layer V1, V2 broken down into a set of points.

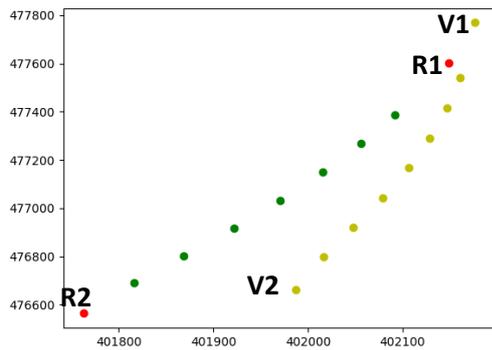


(c) R1, R2 Control points of the real road in the satellite image (shown in red color) and V1, V2 control points of the vector layer (shown in yellow).

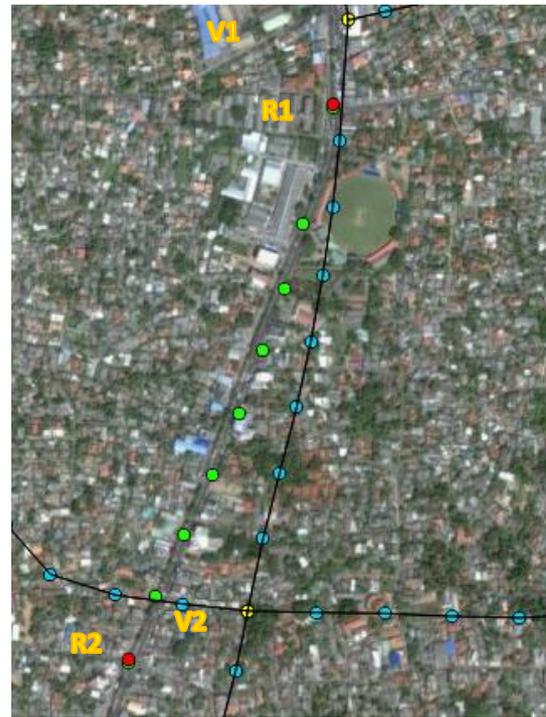


(d) Control points described in (c) presented in a coordinate system. Yellow points represent the initial road vector layer points.

Figure 5.2: Main steps of the proposed approach I.

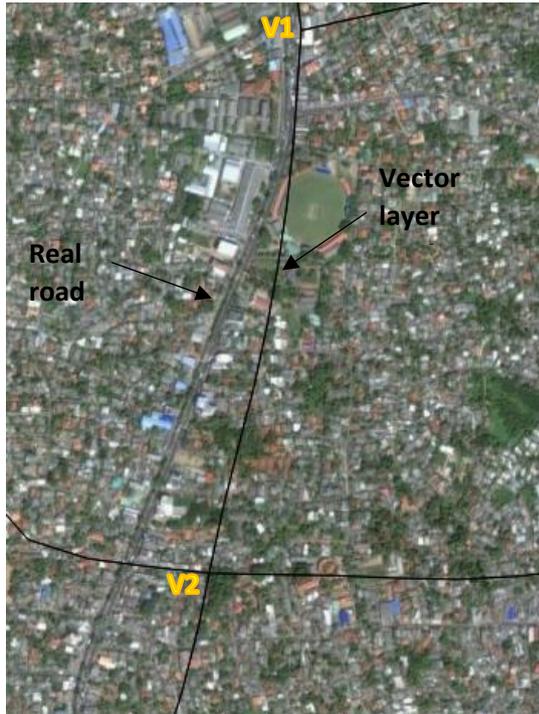


(e) Result after carrying out the alignment approach. The greens points represent the new coordinates of the vector layer points.

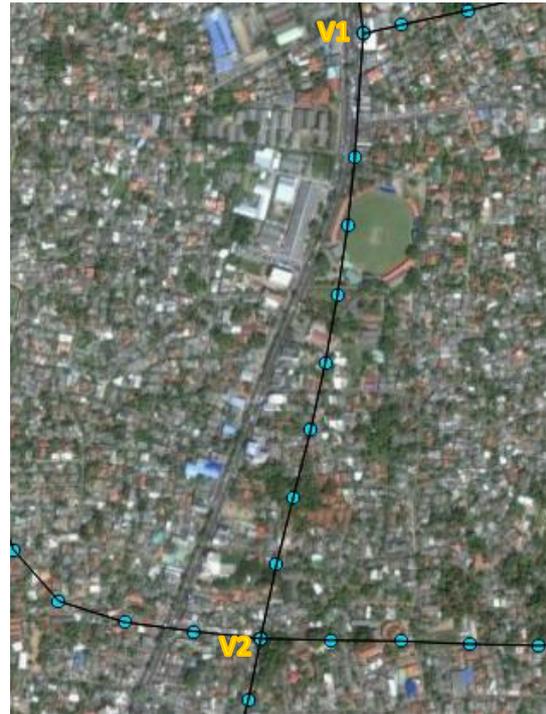


(f) Final aligned road segment. The light green points represent the new aligned road layer.

Figure 5.2: Main steps of the proposed approach I contd.



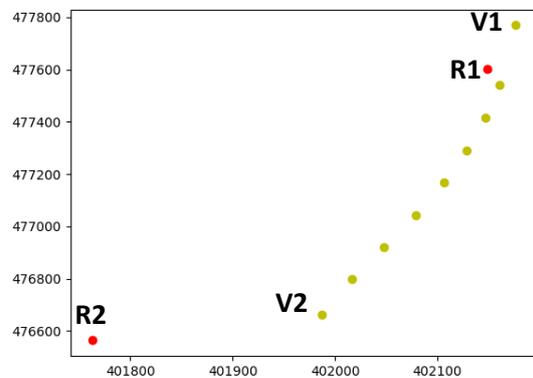
(a) Initial road vector layer V1, V2 with the misalignment.



(b) Initial road vector layer V1, V2 broken down into a set of points.

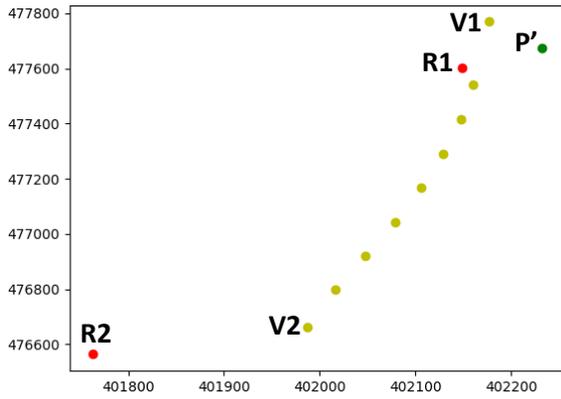


(c) R1, R2 Control points of the real road in the satellite image (shown in red color) and V1, V2 control points of the vector layer (shown in yellow).

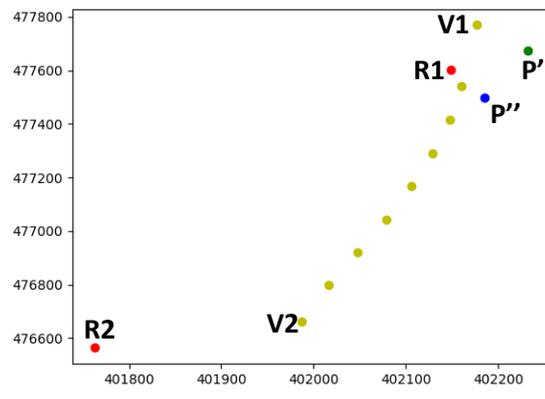


(d) Control points described in (c) presented in a coordinate system. Yellow points represent the initial road vector layer points.

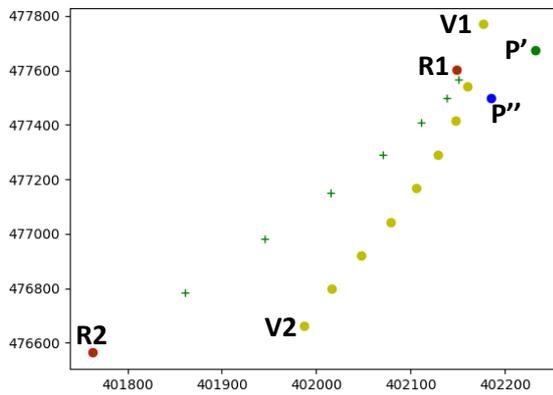
Figure 5.3: Main steps of the proposed approach II considering order two bezier curve



(e) The third control P' (shown in green) found solving the Bezier equation.



(f) The blue point P'' represents the new location of the green control point P' considering relative distance.



(g) The green '+' points represent the new coordinates of the vector layer points shown in yellow color.

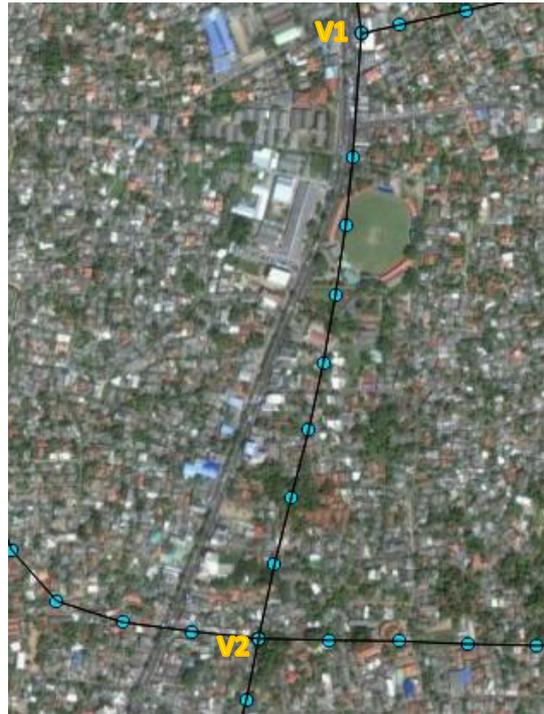


(h) The final aligned road segment (shown in green color points).

Figure 5.3: Main steps of the proposed approach II considering order two bezier curve contd.



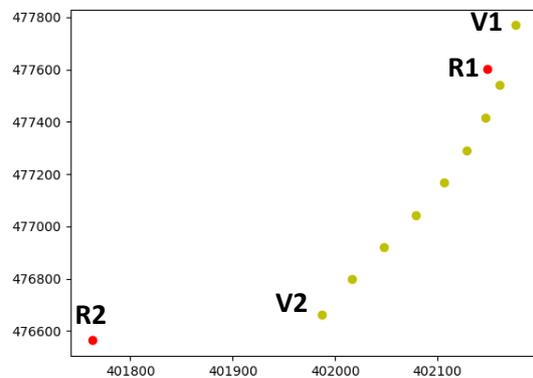
(a) Initial road vector layer V1, V2 with the misalignment.



(b) Initial road vector layer V1, V2 broken down into a set of points.

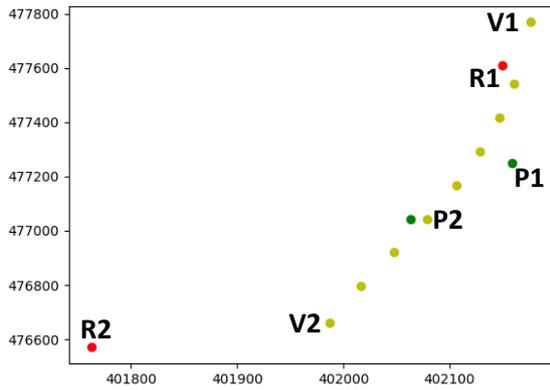


(c) R1, R2 Control points of the real road in the satellite image (shown in red color) and V1, V2 control points of the vector layer (shown in yellow).

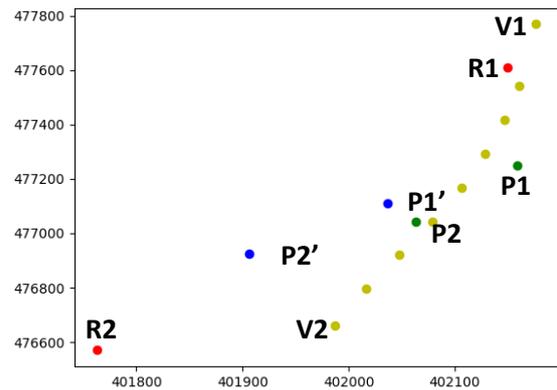


(d) Control points described in (c) presented in a coordinate system. Yellow points represent the initial road vector layer points.

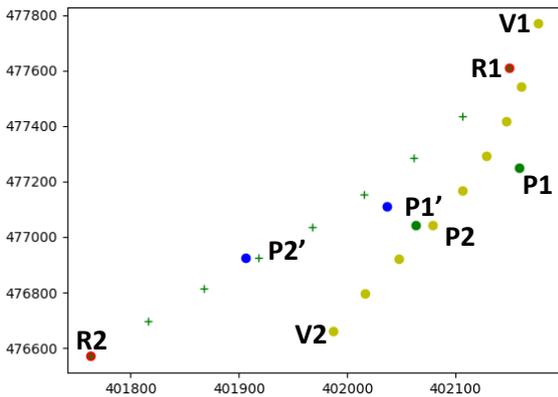
Figure 5.4: Main steps of the proposed approach II considering order three bezier curve



(e) Control point P1, P2 (shown in green) generated by solving the Bezier equation of order three.



(f) The blue points P1', P2' represents the new location of the green control points P1, P2 considering relative distance.



(g) The green '+' points represent the new coordinates of the vector layer points shown in yellow color.



(h) The final aligned road segment (shown in green color points).

Figure 5.4: Main steps of the proposed approach II considering order three bezier curve contd.

The proposed solutions were tested on four test scenarios which consists of different road misalignments/deviations. Test scenarios 1, 2, 3, 4 consists of 5, 7, 10 and 9 misaligned road segments respectively (Images of the test scenarios attached in Appendix A, Figure A.1 – A.4). Test scenarios 1 and 2 contains simple road segments with one or two bends while test scenarios 3 and 4 contains complex road segments with multiple bending points.

The piecewise linear rubber-sheeting algorithm described in Section 2.3.1 was implemented (Code listing attached in Appendix B) and the results were compared with the proposed approaches. Table 5.1, Table 5.2, Table 5.3 and Table 5.4 showcases the results obtained for the area of the polygon for test scenarios 1, 2, 3, 4 respectively for all alignment approaches (The images of the aligned road networks for the proposed approaches are attached in Appendix A, Figure A.6-A.9). The values in bold in the tables represent the minimum value obtained for the polygon area from all alignment approaches for each road segment.

Table 5.1: Results obtained for the Test Scenario 1.

Road Segment	Road Length (km)	Polygon Area (in Map Units)				
		Initial Vector Layer (Misalignment)	Approach I –distance ratio and angle	Approach II – Bezier Curves Order two	Approach II – Bezier Curves Order three	Piecewise Linear Rubber-Sheeting approach (Delaunay Triangulation)
1	0.590	24,787.0	12,211.3	12,481.2	14,561.1	10,758.8
2	1.811	567,989.7	213,104.9	218,306.1	159,706.5	204,303.9
3	0.582	60,598.4	24,136.3	25,110.8	19,273.4	25,807.9
4	1.145	57,427.9	909.5	4,829.3	1,620.9	838.5
5	2.419	245,179.0	73,398.8	12,680.9	86,900.1	NA

According to Table 5.1, the polygon area between the ground truth data and the aligned road segment has been reduced for all alignment approaches for road segments 1, 2, 3 and 4. However, for road segment 5 the piecewise linear rubber sheeting approach with delaunay

triangulation cannot be applied as the road segment is situated out of the triangles generated from the provided control points (scenario illustrated in Appendix A Figure A.5) whereas the other approaches has reduced the polygon area. This can be avoided by providing an additional control point such that the road segment is situated within a triangle.

Table 5.2: Results obtained for Test Scenario 2.

Road Segment	Road Length (km)	Polygon Area (in Map Units)				
		Initial Vector Layer (Misalignment)	Approach I –distance ratio and angle	Approach II – Bezier Curves Order two	Approach II – Bezier Curves Order three	Piecewise Linear Rubber-Sheeting approach (Delaunay Triangulation)
1	0.8729	108,976.1	13,834.2	13,949.9	13,589.8	NA
2	1.192	93,166.4	2,250.0	3,370.8	4,652.8	1,760.7
3	0.539	26,574.6	3,163.0	3,199.9	2,361.4	2,921.9
4	0.588	31,996.1	7,675.7	7,487.2	7,024.0	5,496.7
5	0.689	40,334.1	24,358.6	24,935.5	22,145.0	26,926.0
6	2.061	12,091.1	129,433.3	164,459.1	115,760.7	115,110.2
7	1.490	62,678.6	6,587.6	16,835.2	7,847.8	6,676.1

According to Table 5.2, the polygon area between the ground truth data and the aligned road segment has been reduced for all alignment approaches for road segments 2, 3, 4, 5 and 7. However, for road segment 1 the piecewise linear rubber sheeting approach with delaunay triangulation cannot be applied similar to the road segment 5 in test scenario 1 whereas the other approaches has reduced the polygon area. The road segment 6 is a special situation where the misaligned road segment has a lower value for the polygon area than the aligned road segments of all approaches. However, it can be observed that the misaligned vector layer has cut off the original road segment in multiple places in this situation.

Similarly, in test scenario 3 the polygon area of the aligned road segments have reduced for road segments 2, 4, 5, 6, 8, 9, 10 in all alignment approaches as shown in Table 5.3 and in test scenario

4 the polygon area of the aligned road segments has reduced for road segments 1, 3, 4, 5, 6, 7, 8, 9 in all alignment approaches as shown in Table 5.4. Although, road segments 1 and 7 in test scenario 3 and road segment 2 in test scenario 4 are special situations where the misaligned road segment has a lower value for the polygon area than the aligned road segments of all approaches similar to road segment 6 in test scenario 3.

Table 5.3: Results obtained for Test Scenario 3.

Road Segment	Road Length (km)	Polygon Area (in Map Units)				
		Initial Vector Layer (Misalignment)	Approach I –distance ratio and angle	Approach II – Bezier Curves Order two	Approach II – Bezier Curves Order three	Piecewise Linear Rubber-Sheeting approach (Delaunay Triangulation)
1	4.422	21,009.4	448,730.9	536,358.5	557,461.2	535,995.5
2	0.975	861,469.4	215,228.1	185,023.0	193,534.0	398,922.8
3	3.549	312,766.7	310,864.8	508,038.5	111,342.5	222,078.7
4	6.485	917,703.6	74,491.3	876,460.0	690,300.0	585,805.0
5	3.215	389,327.1	84,625.2	121,729.8	48,164.8	93,373.7
6	4.253	293,946.1	97,744.2	91,737.7	114,163.9	120,772.8
7	2.984	28,474.1	179,556.6	205,359.9	173,948.9	156,569.4
8	4.861	1,738,710.7	240,495.0	362,351.3	332,935.5	437,556.9
9	2.725	843,001.8	132,961.3	67,467.9	207,080.9	143,630.4
10	2.182	642,937.2	10,265.7	49,710.3	20,441.1	12,669.2

Table 5.4: Results obtained for Test Scenario 4.

Road Segment	Road Length (km)	Polygon Area (in Map Units)				
		Initial Vector Layer (Misalignment)	Approach I –distance ratio and angle	Approach II – Bezier Curves Order two	Approach II – Bezier Curves Order three	Piecewise Linear Rubber-Sheeting approach (Delaunay Triangulation)
1	1.684	89,245.7	43,807.6	43,002.3	46,163.1	63,007.4
2	3.543	99,511.9	194,819.1	216,570.1	166,891.4	194,264.2
3	0.922	107,604.1	22,592.2	13,688.9	23,109.7	22,095.2
4	9.260	2,623,742.7	35,139.7	314,305.2	275,509.0	211,802.7
5	3.012	236,831.0	6,437.5	1,900.7	7,868.3	1,435.0
6	3.652	108,151.9	1,285.0	1,211.9	2,158.1	841.2
7	4.694	1,063,325.1	144,456.4	765,171.7	897,696.7	368,467.0
8	3.736	620,631.9	225,853.2	76,079.7	199,757.6	562,049.9
9	11.122	2,584,743.6	2,284,510.8	1,506,097.2	2,107,454.4	966,091.5

To test the accuracy of the alignment approaches the proposed evaluation model described in Section 5.2 was utilized and the Misalignment Reduction Percentage(MRP) was considered. MRP was calculated as shown below:

$$MRP = \frac{\left(\begin{array}{c} \text{Polygon area of} \\ \text{the initial} \\ \text{road vector layer and} \\ \text{real road} \end{array} - \begin{array}{c} \text{Polygon area of} \\ \text{the aligned} \\ \text{road vector layer and} \\ \text{real road} \end{array} \right) * 100}{\text{Polygon area of the initial road vector layer and real road}}$$

Therefore, higher MRP value indicates higher accuracy of the alignment approach while lower MRP value indicates lower accuracy of the alignment approach as the objective of the alignment approaches are to reduce the polygon area between the real road and the aligned road vector layer.

Considering the above values obtained for the polygon area of test scenarios, the MRP was calculated. The MRP for the test scenarios 1, 2, 3, 4 is shown below in Table 5.5, Table 5.6, Table 5.7, Table 5.8 respectively. The values in bold in the tables represent the best performing approach for each road segment. Special road segments where the misaligned road segment has a lower value for the polygon area than the aligned road segments of all approaches were not considered for the MRP calculation.

Table 5.5: MRP for Test Scenario 1.

Road Segment	MRP			
	Approach I – distance ratio and angle	Approach II – Bezier Curves		Piecewise Linear Rubber-Sheeting approach
		Order two	Order three	
1	50.7	49.6	41.3	56.6
2	62.5	61.6	71.9	64.0
3	60.2	58.6	68.2	57.4
4	98.4	91.6	97.2	98.5
5	70.1	94.8	64.6	NA

Table 5.6: MRP for Test Scenario 2.

Road Segment	MRP			
	Approach I – distance ratio and angle	Approach II – Bezier Curves		Piecewise Linear Rubber-Sheeting approach
		Order two	Order three	
1	87.3	87.2	87.5	NA
2	97.6	96.4	95.0	98.1
3	88.1	88.0	91.1	89.0
4	76.0	76.6	78.0	82.8
5	39.6	38.2	45.1	33.2
7	89.5	73.1	87.5	89.3

Table 5.7: MRP for Test Scenario 3.

Road Segment	MRP			
	Approach I – distance ratio and angle	Approach II – Bezier Curves		Piecewise Linear Rubber-Sheeting approach
		Order two	Order three	
2	75.0	78.5	77.5	53.7
3	0.6	-62.4	64.4	29.0
4	91.9	4.5	24.8	36.2
5	78.3	68.7	87.6	76.0
6	66.7	68.8	61.2	58.9
8	86.2	79.2	80.9	74.8
9	84.2	92.0	75.4	83.0
10	98.4	92.3	96.8	98.0

Table 5.8: MRP for Test Scenario 4.

Road Segment	MRP			
	Approach I – distance ratio and angle	Approach II – Bezier Curves		Piecewise Linear Rubber-Sheeting approach
		Order two	Order three	
1	50.9	51.8	48.3	29.4
3	79.0	87.3	78.5	79.5
4	98.7	88.0	89.5	91.9
5	97.3	99.2	96.7	99.4
6	98.8	98.9	98.0	99.2
7	86.4	28.0	15.6	65.3
8	63.6	87.7	67.8	9.4
9	11.6	41.7	18.5	62.6

It can be observed in Table 5.7 that road segment 3 in test scenario 3 has a significantly lower MRP for alignment considering bézier curves of order two approach. However, the MRP has significantly increased when the alignment considering bézier curves of order three approach is

considered. This is due to the shape of the road segment which cannot be modeled by an order two bézier curve. It can be observed (Figure A.9, A.8 in Appendix A) that the bézier approach deforms the shape of the existing vector layer when the shape of a road segment cannot be modeled considering an order two or order three curve. However, the positional misalignment between the real road and the vector layer is reduced.

The MRP values obtained showcase that the proposed approach 1 performs well over the existing piecewise linear rubber sheeting approach in road segments 3, 5 in test scenario 1, road segments 1, 5, 7 in test scenario 2, road segments 2, 4, 5, 6, 8, 9, 10 in test scenario 3 and road segments 1, 4, 7, 8 in test scenario 4. The proposed approach 2 considering order two bézier curves performs well over the existing piecewise linear rubber sheeting approach in road segments 3, 5 in test scenario 1, road segment 1, 5 in test scenario 2, road segments 2, 6, 8, 9 in test scenario 3 and road segments 1, 3, 8 in test scenario 4. Similarly, the proposed approach 2 considering order three bézier curves performs well over the existing piecewise linear rubber sheeting approach in road segments 2, 3, 5 in test scenario 1, road segment 1, 3, 5 in test scenario 2, road segments 2, 3, 5, 6, 8 in test scenario 3 and road segments 1, 8 in test scenario 4.

Average figures from MRP calculations are illustrated in Figure 5.5. A combined average is derived for the bézier approach by obtaining the maximum MRP for each road segment out of order two and order three bézier approaches and then calculating the average of resulting figures.

As illustrated in Figure 5.5 it can be observed that the combined bézier approach has a higher average MRP value of 75.8 than all other approaches. Whereas the bézier approach considering order two curves has the lowest average MRP value of 67.4. The proposed approach 1 considering distance ratio and angle performs fairly well with an average MRP of 73.6.

Table 5.9 presents a comparison of the piecewise linear rubber-sheeting approach, snake related algorithms and the proposed approaches under several criteria.

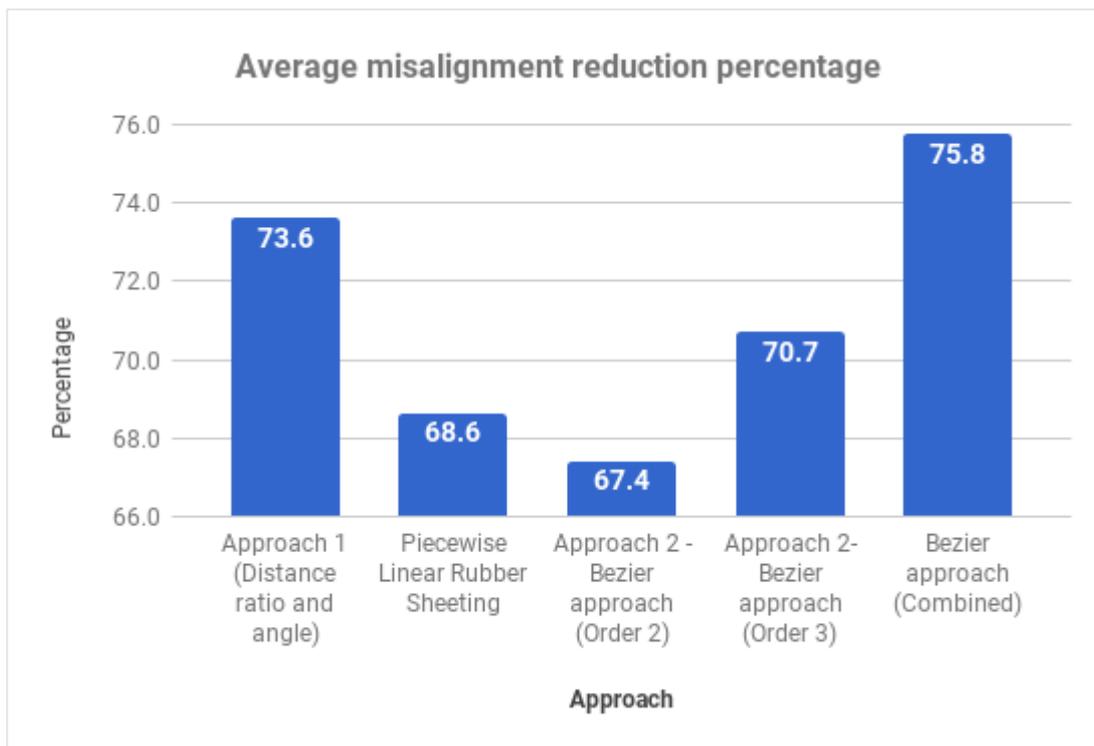


Figure 5.5: Average MRP for all alignment approaches

Table 5.9: Comparison of the proposed approaches with existing approaches.

Criteria	Piecewise Linear Rubber-Sheeting Approach	Snake Related Algorithms	Proposed Approach 1- Considering distance ratio and angle	Proposed Approach 2- Considering Bezier Curves
Requirement of road feature (edges) for refinement	No	Yes	No	No
Requirement of control points for refinement	Yes	Yes	Yes	Yes
Alignment possible when road vector data is inconsistent with the road shape in the imagery	No	No	No	No

Refinement can be carried out to a single road segment with two control points	No	Yes	Yes	Yes
Suitable for region alignment	Yes	Yes	Yes	Yes
Control points of the vector layer are required to be close to the real road segment to obtain a desired alignment	No	Yes	No	No
Deforms the shape of the road vector layer	No	Yes	No	Yes

5.4. Summary

This chapter elaborated on the proposed evaluation model for conflation approaches, which considers the area of the polygon covering the ground truth data of the real road and the aligned vector layer. Four test scenarios were considered to determine the accuracy of the proposed approaches. The results obtained for the test scenarios showcased that the proposed combined bézier approach and distance ratio and angle approach performs well over the existing piecewise linear rubber sheeting approach. However, there are certain special scenarios such as the situation where real road layer shape is inconsistent with the road vector layer and the situation where the real road layer cut off the road vector layer where all approaches are unable to produce a desired alignment.

Chapter 6 - Conclusions

6.1. Introduction

This chapter includes a review of the research aims and objectives, research problem, limitations of the current work and implications for further research.

6.2 Conclusions about research questions (aims/objectives)

The main aim of the research was to propose an approach to refine the existing road vector layer to align with the actual road network layer of Google satellite imagery. Two alignment approaches were introduced in this study to refine the existing road vector layers. The first approach considered the distance ratio and angle of the control points of the road layer in the satellite image and the road layer in vector format. Whereas, the second approach considered the concept of Bézier curves for the alignment process. An objective of the research was also to devise an evaluation mechanism to measure the success of the alignment process. Thus, a new evaluation model considering the area of the polygon bordering the real road in the satellite image and the aligned vector layer was introduced. The misalignment reduction percentage was calculated considering this evaluation model to measure the success of the refinement approaches. The existing Piecewise Linear Rubber Sheeting approach for alignment was implemented and evaluated against the two proposed approaches. The evaluation results showcased that the proposed combined bézier approach and distance ratio and angle approach performs well over the existing piecewise linear rubber sheeting approach with an average misalignment reduction percentage of 75.8% and 73.6% respectively for the tested scenarios. Also, the Piecewise Linear Rubber Sheeting approach requires additional control points when aligning some road segments in the selected regions apart from the two junctions (end points) of the road layer. However, the two proposed approaches in this study do not require additional control points to carryout the alignment process. Thus, it can be concluded that the proposed approaches can be utilized over the existing piecewise linear rubber sheeting approach in required situations to obtain a desired conflation/alignment between the road vector layer and the road layer in the satellite imagery.

6.3 Conclusions about research problem

The misalignment between the existing road vector layer (vector data) and the actual road layer represented in raster data models (Satellite images) can be refined by providing control points from both data models and carrying out an alignment algorithm. The alignment algorithm focuses on reducing the positional inconsistency among the two data models. This method of alignment/conflation is a geometric, vertical, semiautomated vector to imagery conflation approach as the alignment focuses on eliminating the geometric and spatial differences between the data models.

This study contributed to the domain of vector to imagery conflation by introducing two alignment/conflation approaches. The two proposed approaches are capable of handling misalignments/deviations of a road segment in the vector format with a minimum number of two control points while the existing piecewise linear rubber sheeting approach with Delaunay triangulation requires a minimum number of three control points to generate triangles. Also, comparison with the existing snakes related approach for alignment, the two approaches does not require extracting road features (road edges) for the alignment. The study also contributed to the domain of GIS by introducing a new evaluation model which can be utilized to measure the success of conflation approaches as well as road extraction from imagery approaches.

6.4 Limitations

The evaluation results showcased that there are certain special scenarios such as the situation where real road layer shape is inconsistent with the road vector layer and the situation where the real road layer cut off the road vector layer where the proposed approaches are unable to produce a desired alignment. The bézier approach considers a road segment as a quadratic or cubic bézier curve. Therefore, the bézier approach deforms the shape of the vector layer when the road segment cannot be modeled with an order two or order three bézier curve.

6.5 Implications for further research

The two alignment approaches proposed in this study as well as the existing alignment approaches as Snakes algorithm and Pricewise Linear Rubber Sheeting approach does not perform well when the existing road vector layer shape and the actual road layer shape is inconsistent from each other. Therefore, an approach to align the vector layer by providing additional intermediate control points in such situations can be explored. Similarly, the proposed alignment approaches and the existing approaches do not perform well in special situation where the real road cut off the misaligned road vector layer. Thus, the reason for this problem can be explored. Further, in this research the road is considered as a quadratic or a cubic bézier curve to carryout the alignment process. Evaluation results showcased that when order two and order three curves were combined the MRP increased. Thus, this approach can be generalized for any type of road by proposing a generalized curvature equation for all road layers or determining the number of control points required to generate the curve of the road when the points of the vector layer is given. Therefore, providing the capability of easily transforming any type of road layer without deforming the shape of the existing vector layer.

References

- [1] C. C. Chen, C. A. Knoblock, and C. Shahabi, "Automatically conflating road vector data with orthoimagery," *Geoinformatica*, vol. 10, no. 4, pp. 495–530, 2006.
- [2] W. Song, J. M. Keller, T. L. Haithcoat, and C. H. Davis, "Automated geospatial conflation of vector road maps to high resolution imagery," *IEEE Trans. Image Process.*, vol. 18, no. 2, pp. 388–400, 2009.
- [3] W. Song, J. M. Keller, T. L. Haithcoat, and C. H. Davis, "Relaxation-Based Point Feature Matching for Vector Map Conflation," *Trans. GIS*, vol. 15, no. 1, pp. 43–60, 2011.
- [4] A. Saalfeld, "Conflation Automated map compilation," *INT. J. Geogr. Inf. Syst.*, vol. 2, no. 3, pp. 217–228, 1988.
- [5] J. J. Ruiz, F. J. Ariza, M. a. Ureña, and E. B. Blázquez, "Digital map conflation: a review of the process and a proposal for classification," *Int. J. Geogr. Inf. Sci.*, vol. 25, no. 9, pp. 1439–1466, 2011.
- [6] C. Cao and Y. Sun, "Automatic road centerline extraction from imagery using road GPS data," *Remote Sens.*, vol. 6, no. 9, pp. 9014–9033, 2014.
- [7] M. a Cobb, M. J. Chung, H. Foley III, F. E. Petry, K. B. Shaw, and H. V. Miller, "A Rule-based Approach for the Conflation of Attributed Vector Data," *Geoinformatica*, vol. 2, no. 1, pp. 7– 35, 1998.
- [8] V. Walter and D. Fritsch, "Matching spatial data sets: a statistical approach," *Int. J. Geogr. Inf. Sci.*, vol. 13, no. 5, pp. 445–473, 1999.
- [9] S. Yuan and C. Tao, "Development of conflation components," *Proc. Geoinformatics*, vol. 2, no. 2, pp. 1–13, 1999.
- [10] A. Masuyama, "Methods for detecting apparent differences between spatial tessellations at different time points," *Int. J. Geogr. Inf. Sci.*, vol. 20, no. 6, pp. 633–648, 2006.
- [11] C. C. Chen, S. Thakkar, C. Knoblock, and C. Shahabi, *Automatically Annotating and Integrating Spatial Datasets*, vol. 2750. 2003.

- [12] H. Hild and D. Fritsch, "Integration of vector data and satellite imagery for geocoding," *Int. Arch. Photogramm. Remote Sens.*, vol. 32, pp. 246–251, 1998.
- [13] S. Filin and Y. Doytsher, "A Linear Conflation Approach for the Integration of Photogrammetric Information and GIS Data," *Int. Arch. Photogramm. Remote Sens.*, vol. XXXIII, pp. 282–288, 2000.
- [14] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.
- [15] M. S. White, and P. Griffin, "Piecewise Linear Rubber-Sheet Map Transformation," *Cartogr. Geogr. Inf. Sci.*, vol. 12, no. 2, pp. 123–131, 1985.
- [16] W. Song, J. M. Keller, T. L. Haithcoat, C. H. Davis, and J. B. Hinsien, "An Automated Approach for the Conflation of Vector Parcel Map with Imagery," *Photogramm. Eng. Remote Sensing*, vol. 79, no. 6, pp. 535–543, 2013.
- [17] J. R. Hwang, J. H. Oh, and K. J. Li, "Query transformation method by delaunay triangulation for multi-source distributed spatial database systems," *Proc. ninth ACM Int. Symp. Adv. Geogr. Inf. Syst. - GIS '01*, p. 41, 2001.
- [18] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *Int. J. Comput. Vis.*, vol. 1, no. 4, pp. 321–331, 1988.
- [19] M. F. A. Fortier, D. Ziou, C. Armenakis, and S. Wang, "Automated updating of road information from aerial images," *Am. Soc. Photogramm. Remote Sens. Conf.*, no. December, pp. 16–23, 2000.
- [20] W. V. A. H. R. Fernando, "Refining Vector Layers Using Satellite Images", University of Colombo School of Computing, 2011.
- [21] D. C. S. De Silva, "An Approach to Align Existing Vector Layers with High Resolution Satellite Images", University of Colombo School of Computing, 2012.
- [22] R. A. V. U. Rupasinghe, "A Methodology to Align Existing Vector Layer Used for Roads with High Resolution Satellite Images", University of Colombo School of Computing, 2015.
- [23] T. W. Sederberg, *Computer Aided Geometric Design*, pp. 109-120, 2012.

- [24] M. Kamermans, "A Primer on Bézier Curves," Pomax.github.io. [Online]. Available: <https://pomax.github.io/bezierinfo>. [Accessed: 17-Oct-2017].

Appendix A: Diagrams

Figures A.1 – A.4 showcases the four test scenarios considered for evaluation.

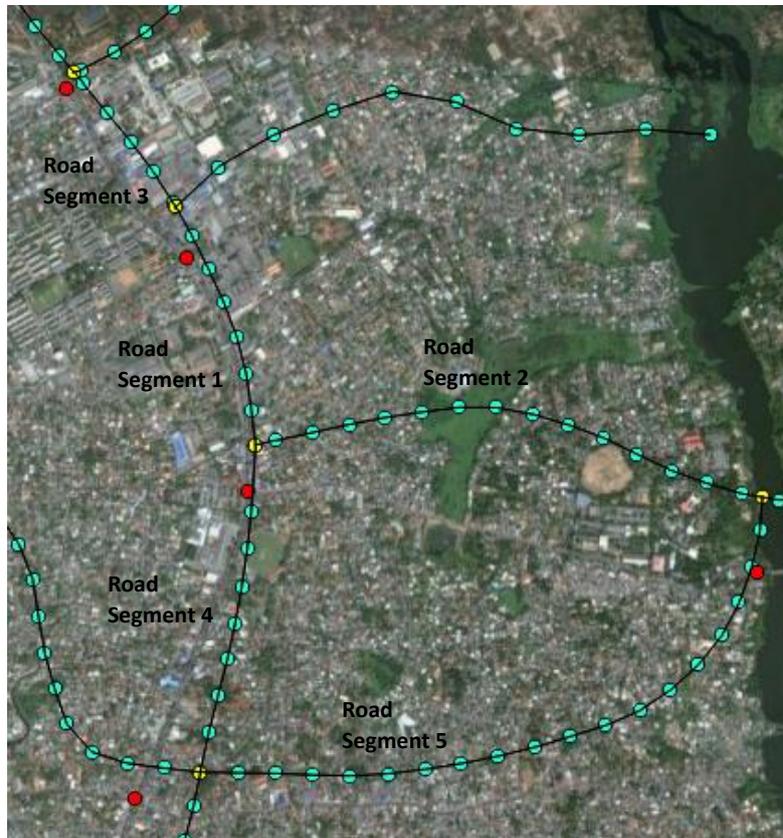


Figure A.1: Test Scenario 1 (Red points and Yellow points represent the control points provided from the satellite imagery and vector layer respectively)



Figure A.2: Test Scenario 2 (Red points and Yellow points represent the control points provided from the satellite imagery and vector layer respectively)

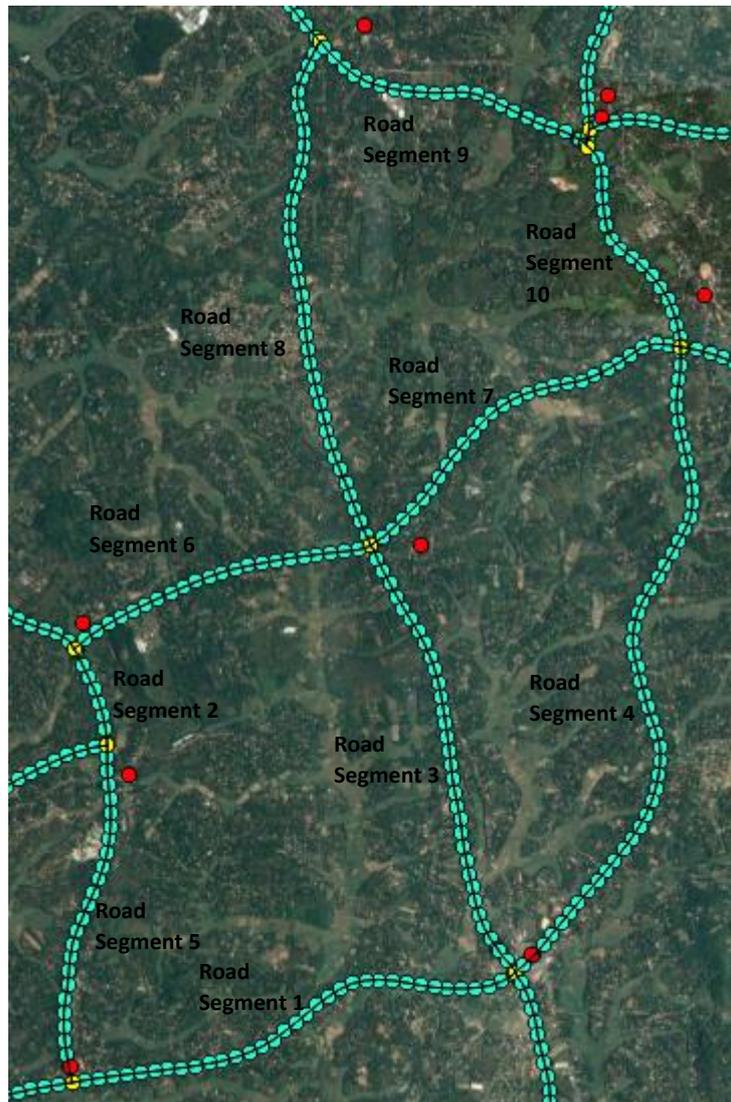


Figure A.3: Test Scenario 3 (Red points and Yellow points represent the control points provided from the satellite imagery and vector layer respectively)

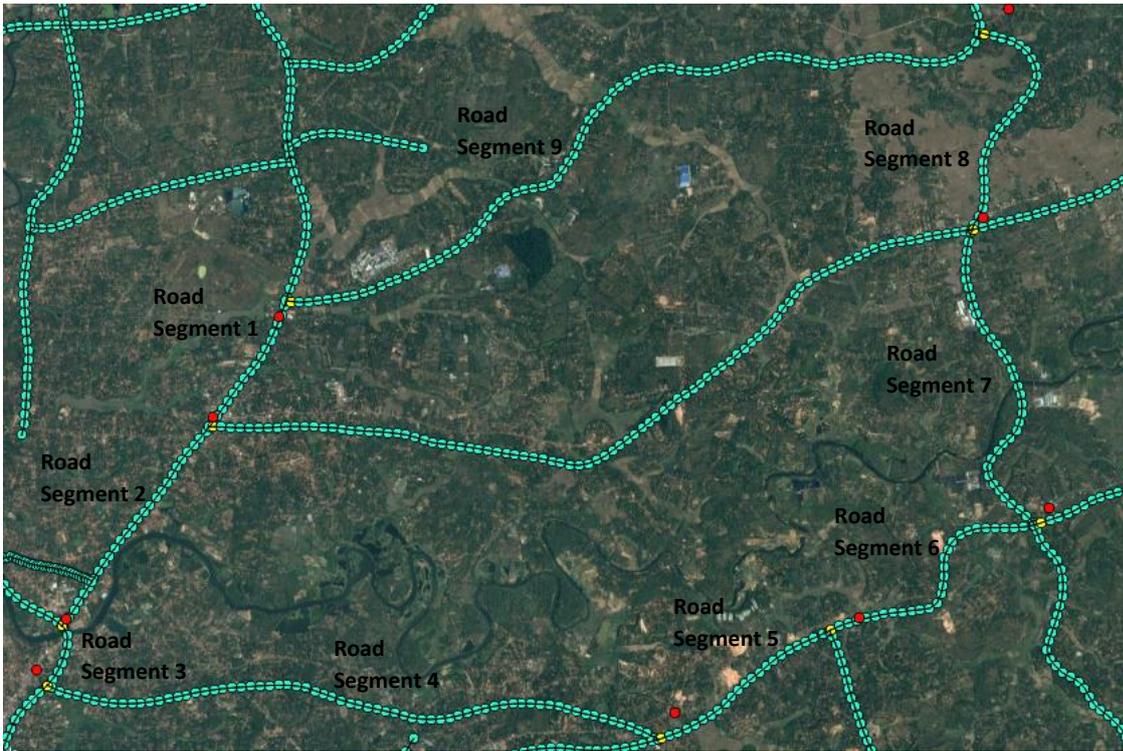


Figure A.4: Test Scenario 4 (Red points and Yellow points represent the control points provided from the satellite imagery and vector layer respectively)

Figure A.5 illustrated below showcase road segment 5 in test scenario 1 situating out of the generated triangles from the control points. (Blue '+' points represent the road segment 5, and the red and green points represent the provided control points of the satellite image and vector layer respectively.)

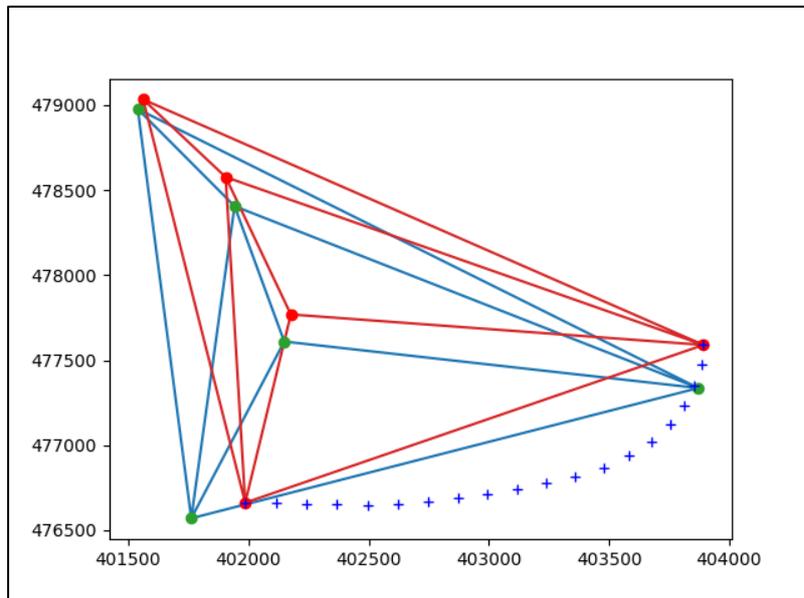
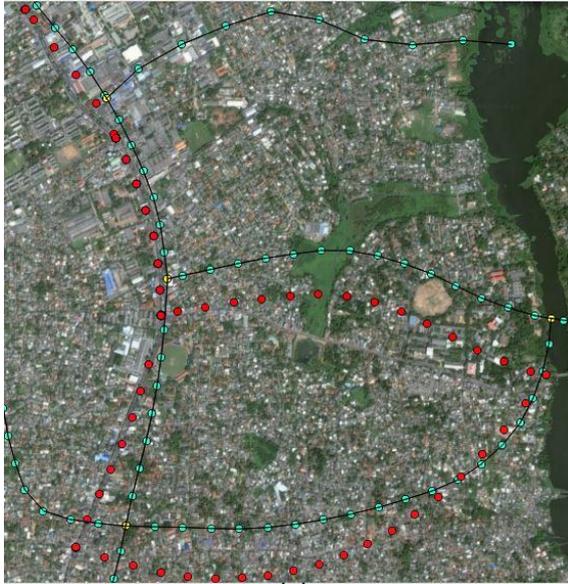


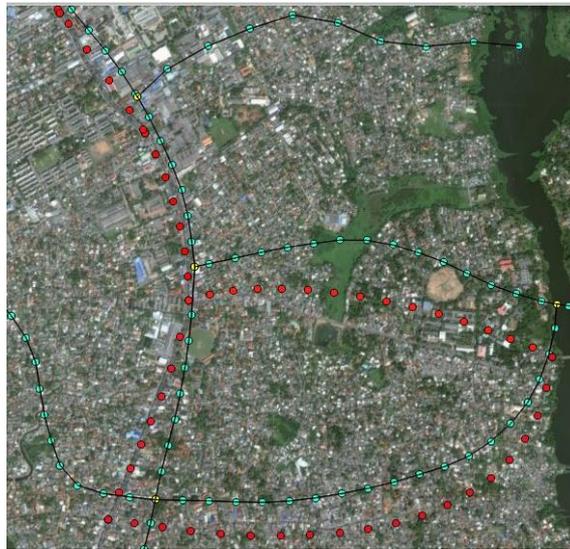
Figure A.5: Road segment situated out of the area of the generated Delaunay triangles.



(a)



(b)



(c)

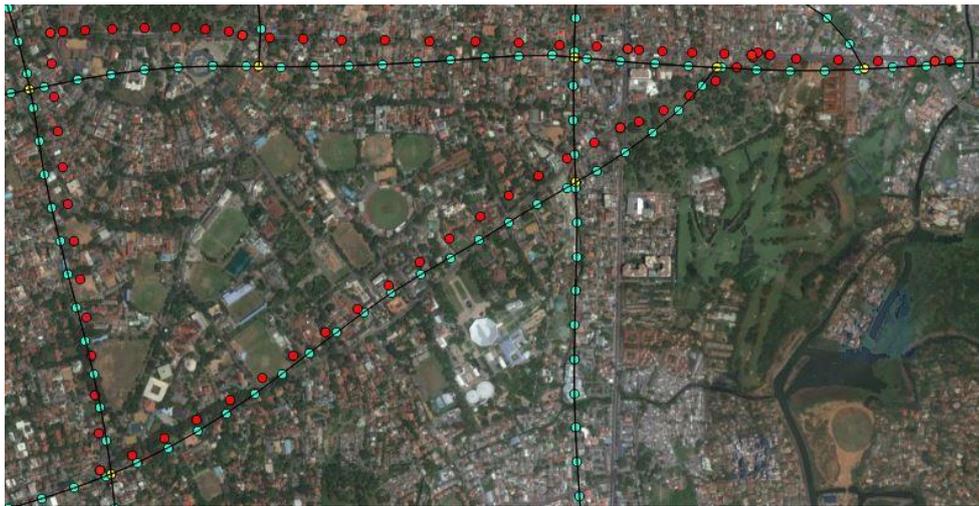
Figure A.6: Transformed road vector layer points (shown in red color points) by the proposed approaches for Test Scenario 1. (a) Approach I. (b) Approach II – Order two bezier curve. (c) Approach II – Order three bezier curve.



(a)



(b)



(c)

Figure A.7: Transformed road vector layer points (shown in red color points) by the proposed approaches for Test Scenario 2. (a) Approach I. (b) Approach II – Order two bezier curve. (c) Approach II – Order three bezier curve.

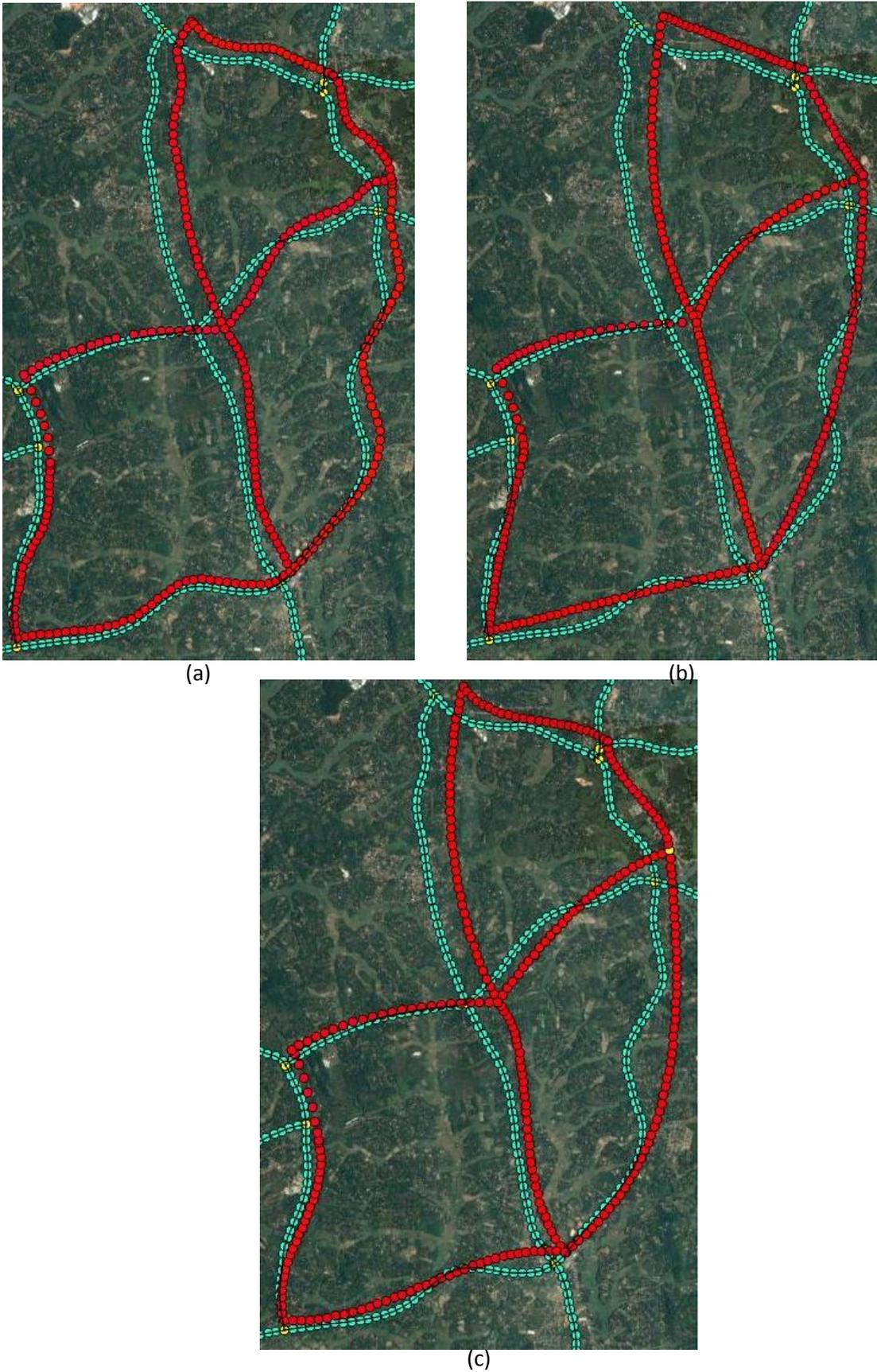
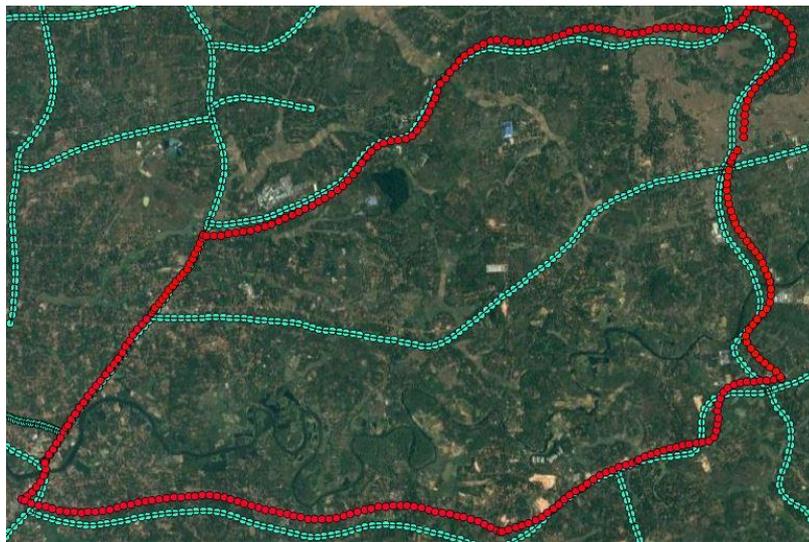
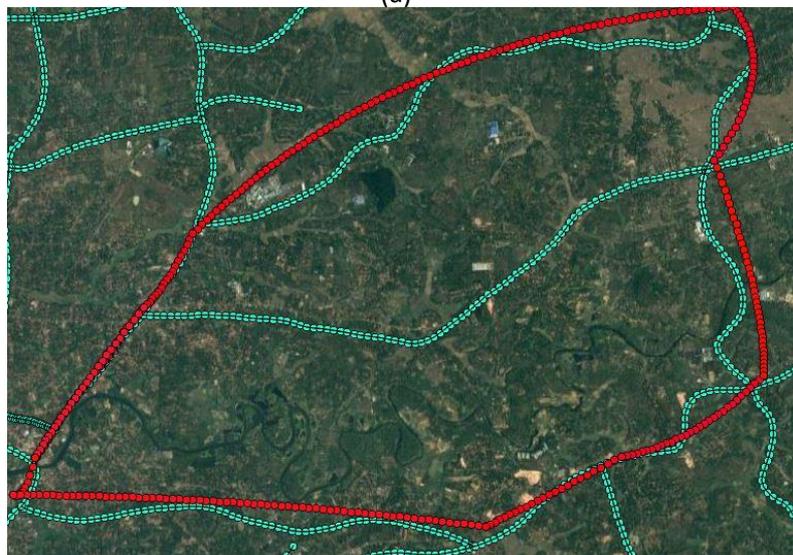


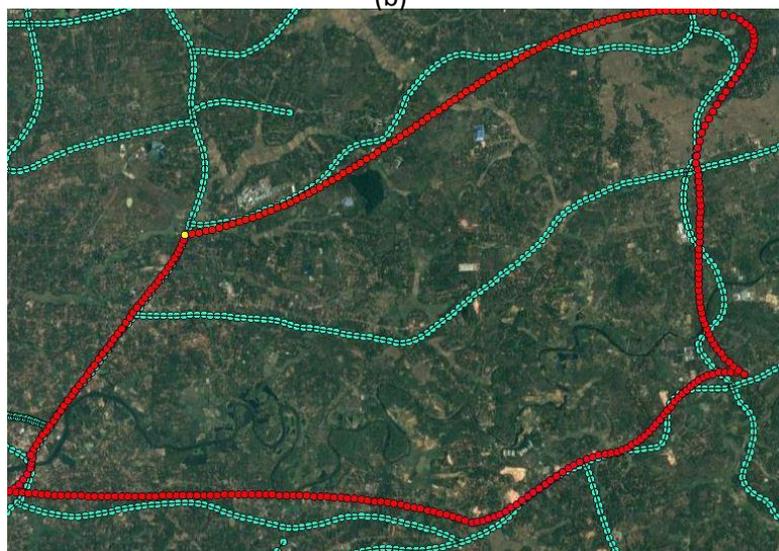
Figure A.8: Transformed road vector layer points (shown in red color points) by the proposed approaches for Test Scenario 3. (a) Approach I. (b) Approach II – Order two bezier curve. (c) Approach II – Order three bezier curve.



(a)



(b)



(c)

Figure A.9: Transformed road vector layer points (shown in red color points) by the proposed approaches for Test Scenario 4. (a) Approach I. (b) Approach II – Order two bezier curve. (c) Approach II – Order three bezier curve.

Figure A.10 represents the road segment 6 in Test Scenario 2 where the road vector layer has cut off the actual road segment in the satellite image.

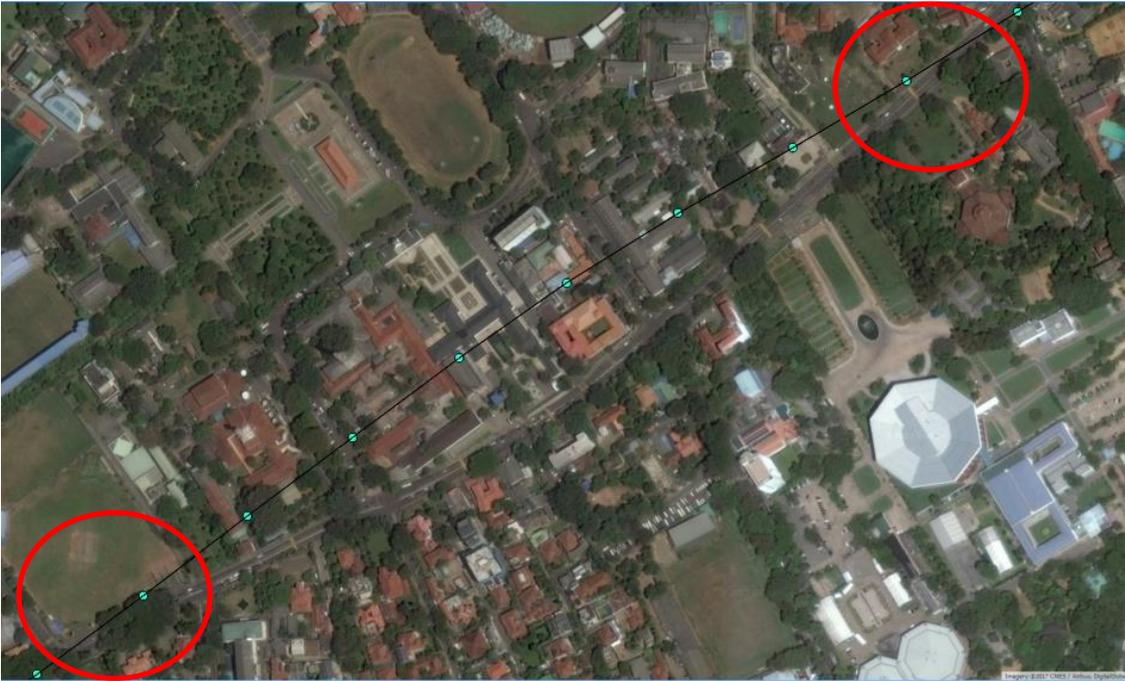


Figure A.10: Situation where the road vector layer cut off the actual road segment

Appendix B: Code Listings

A detailed implementation of the existing piecewise linear rubber-sheeting approach is provided below.

The following function 'transformation_matrix' showcase the process of generating the transformation matrix of the Piecewise linear rubber sheeting approach with Delaunay triangulation.

```
def transformation_matrix(vector, points, order):
    Tmatrix = []
    p1 = order[0]
    p2 = order[1]
    p3 = order[2]

    A = np.array(
        [[float(vector[p1][0]), float(vector[p1][1]), 1],
         [float(vector[p2][0]), float(vector[p2][1]), 1],
         [float(vector[p3][0]), float(vector[p3][1]), 1]])

    B = np.array([float(points[p1][0]), float(points[p2][0]),
                  float(points[p3][0])])

    Tmatrix.append(np.linalg.solve(A,B))

    C = np.array([[float(vector[p1][0]), float(vector[p1][1]), 1],
                  [float(vector[p2][0]), float(vector[p2][1]), 1],
                  [float(vector[p3][0]), float(vector[p3][1]), 1]])

    D = np.array([float(points[p1][1]), float(points[p2][1]),
                  float(points[p3][1])])

    Tmatrix.append(np.linalg.solve(C,D))

    E = np.array([[float(vector[p1][0]), float(vector[p1][1]), 1],
                  [float(vector[p2][0]), float(vector[p2][1]), 1],
                  [float(vector[p3][0]), float(vector[p3][1]), 1]])

    F = np.array([1, 1, 1])

    Tmatrix.append(np.linalg.solve(E,F))

    return Tmatrix
```

The function 'draw_roadsegment' showcase the process of finding the new coordinates of the vector layer point of a road segment considering the appropriate transformation matrix.

```
def draw_roadsegment(v, Tmatrix):
    newPoints = [['X', 'Y']]
    for p in v[2:-1]:
        x = float(p[0])*Tmatrix[0][0]+
            float(p[1])*Tmatrix[0][1]+Tmatrix[0][2]
        y = float(p[0])*Tmatrix[1][0]+
            float(p[1])*Tmatrix[1][1]+Tmatrix[1][2]
        plt.plot(x,y, 'g+')
        newPoints.append([x,y])
    return newPoints
```