Design and Development of a Novel Framework for the Identification of Roundabouts in a given Map

A Thesis

submitted for the award of the degree of

Doctor of Philosophy

in

Computer Science and Engineering Department

Submitted by

Rakesh Singh
(Reg no: 951703002)

Under the Guidance of

Dr. Prashant Singh Rana
Associate Professor

Dr. Neeru Jindal
Associate Professor

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

Thapar Institute of Engineering and Technology, Patiala, Punjab - 147004, India
July 2023

Certificate

I hereby certify that the work, which is being presented in the thesis, entitled "A novel approach for detecting roundabouts in maps based on analysis of core map data", in partial fulfillment of the requirements for the award of the degree of Doctor of Philosophy and submitted to the institution is an authentic record of my work carried out under the supervision of Dr. Prashant Singh Rana and Dr. Neeru Jindal. I have cited the reference about the text(s)/figure(s)/table(s) from where they have been taken.

The matter presented in this thesis has not been submitted either in-part or full to any other University/Institute for the award of any other degree.

Rakesh Singh

(Rakesh Singh)
Registration No. 951703002

This is to certify that the above statements made by the candidate are correct and true to the best of my knowledge.

Verified by:

(Dr. Prashant Singh Rana)
Supervisor
Computer Science and Engineering Department,
Thapar Institute of Engineering and Technology, Patiala, Punjab, India

Neeru

(Dr. Neeru Jindal)
Co - Supervisor
Electronic and Telecommunication Department,
Thapar Institute of Engineering and Technology, Patiala, Punjab, India

dedicated to God.

Acknowledgements

I thought this day would never arrive. Now that it has, my heart overflows with gratitude for those who have been with me throughout the ordeal.

I want to express my deep gratitude to Dr. Prashant Singh Rana and Dr. Neeru Jindal, my research supervisors, for their patient guidance, enthusiastic encouragement, and useful critiques of this research work. Their guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better team to mentor my Ph.D. study. I would also like to thank my technical guru Mr. Dhaneesh Nair.

I thank my fellow labmates for not only their stimulating discussions and valuable suggestions; but also for all the fun we shared. I truly appreciate my colleagues, including those from other disciplines, for enriching me by sharing their experiences. We've all been there for one another and have taught ourselves and each other many tools. I know that I could always ask for advice and opinions on any issue that I may be dealing with. I'm thankful to the graduate and post-graduate students at TIET. My life would have been very dull without their fun interactions.

Numerous faculty members have always been very kind with their words of encouragement. Crossing paths with them while walking on campus has ever brought a smile to my face. I wish to give them my sincerest thanks. The office staff has always been available and ready for the assistance of any kind. The security guards, the hostel staff, and mess staff have not only been extremely diligent with their duties but have often accommodated requests with a smile. They have made my stay at the campus a pleasurable one.

I wish to thank my family for always having my back. Irrespective of what they were dealing with on an individual level, they were always available for a conversation. I owe a lot to my parents, who encouraged and helped me at every stage of my personal and academic life, and longed to see this achievement come true.

My research would have been impossible without the support HERE Maps. Finally special thanks to God for making this possible.

Abstract

We improve autonomous vehicle navigation by enhancing roundabout detection accuracy. Our streamlined method integrates fundamental map data with existing techniques, simplifying machine learning and achieving precise results. We focus on roundabout detection, junction identification, curved road approximation, road kink detection, and roundabout size validation.

The first scheme centers on extracting vital geometric data from map point descriptions to identify map features, including roundabouts. The approach has two components: an algorithm successfully detects roundabouts in 80% of cases, and a machine learning model achieves exceptional accuracy for the remaining cases. Overall, the combined system yields a roundabout detection rate of over 97%. A novel scheme identifies roundabouts in the Map domain by extracting closed loops in roads and using a deterministic algorithm to find common junctions. Data from America and Europe are used to extract features using logical and domain knowledge. The machine model achieves 81% accuracy, reducing manual efforts in verification and offering practical solutions for road mapping.

Two adaptive algorithms are proposed to convert image curves into continuous straight lines or polylines, addressing limitations of existing approaches in map-related scenarios. The algorithms use parameterized equations with a tuning value (τ) to achieve customizable results, showcasing improved performance compared to other methods. The solution offers tunability and enhanced curve approximation. The fourth scheme addresses a practical open problem: identifying kinks in 2D road maps formed by linear collections of lines. Kinks are inversely related to the number of lines, so more lines mean fewer kinks. Our proposed technique utilizes spline interpolation with length manipulation to pinpoint these kink locations. This method is unique, as no other practical technique currently exists for identifying kinks in real map data. With a 10cm threshold, the technique achieved 91% identification accuracy for Frankfurt and 83% for the USA.

The proposed Fifth Scheme validates roundabouts through geometry and a spline technique, using a non-rational uniform B-spline function. It calculates the radius based on curvature, fits an ellipse or circle to the shape within a bounding box, and achieves a 92.2% success rate in Europe with a 10-meter threshold. Time saved on manual verification and correction.

Keyword: Roundabout; Maps; Spline; Machine learning; Autonomous vehicles

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Map domain is very challenging and dynamic in nature, our research is about one of the map feature. It is divided into multiple part.

Initially we identify the roundabout by determinisitic and nondeterministic technique. The need for accurate spatial databases and their automatic updating is increasing rapidly. Road networks form key information layers in topographic databases since they are used in such a wide variety of applications. As the extraction of roads from images is still generally manual, costly and time-consuming, there is a growing imperative to automate the process.

However, such a feature extraction task has proved to be difficult to automate. The problem for automatic road extraction lies mostly in the complex content of aerial images. Many authors uses approach to ease the complexity of the image interpretation task, prior information can be used [5, 6, 7, 8]. A modern roundabout is a type of looping junction on which road traffic travels in one direction around a central island and priority is given to the circulating flow. Robinson et al. shows that Signs usually directs traffic entering the circle to slow down and give way to traffic already on it [9]. In this way, a roundabout helps manoeuvre traffic in a safe and convenient way. Few roundabouts are large enough to become famous landmarks or recreation squares in cities.

In general, a roundabout is a closed loop that controls traffic from an incoming road as shown in Fig 1.1. Generally speaking, roundabouts vary in radius from 12 to 30 meters depending upon the type, location and grade (state highway, national highway, etc.) of the road [10].

The proposed approach begins with the analysis of the raw map data and is a deterministic one arrived at by thorough domain analysis. On analysing the huge data, it was observed that the problem of identification of roundabouts is not a single complex problem but a collection of many relatively simpler problems. The proposed technique takes this into consideration while analysing the data and helps identify the important attributes which separate an average circular 2D graph from a meaningful roundabout.

Prediction of the accuracy of junction in roundabout improves the overall accuracy of

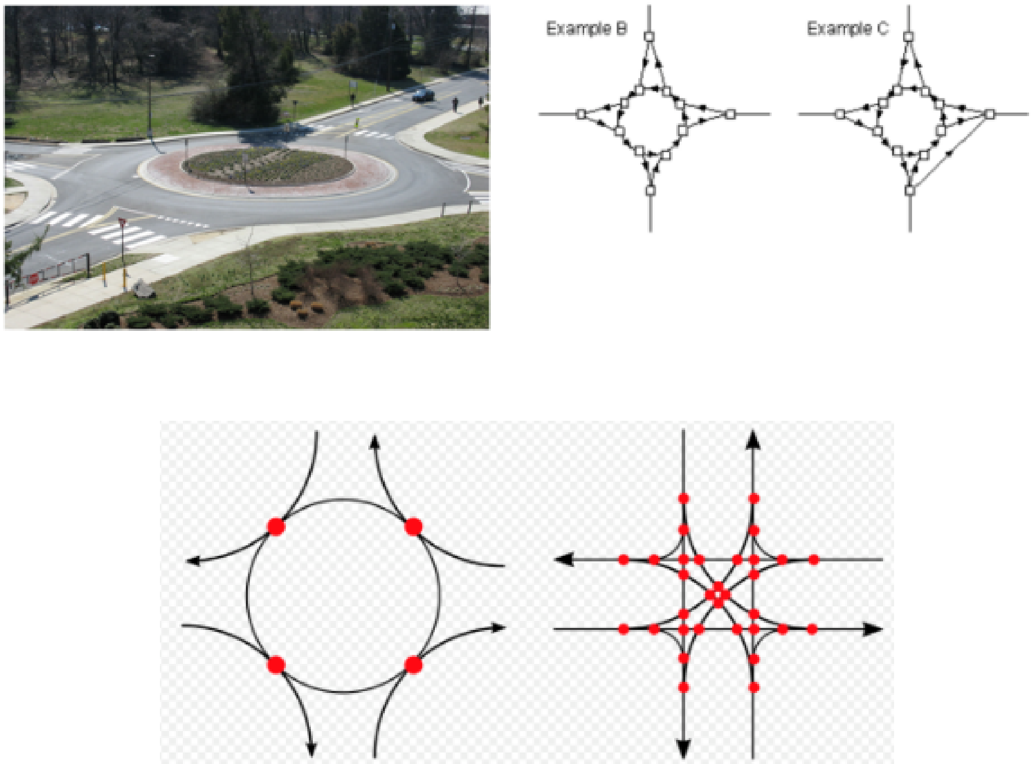


Fig 1.1: Representation of a roundabout, its geometry and possible traffic flows

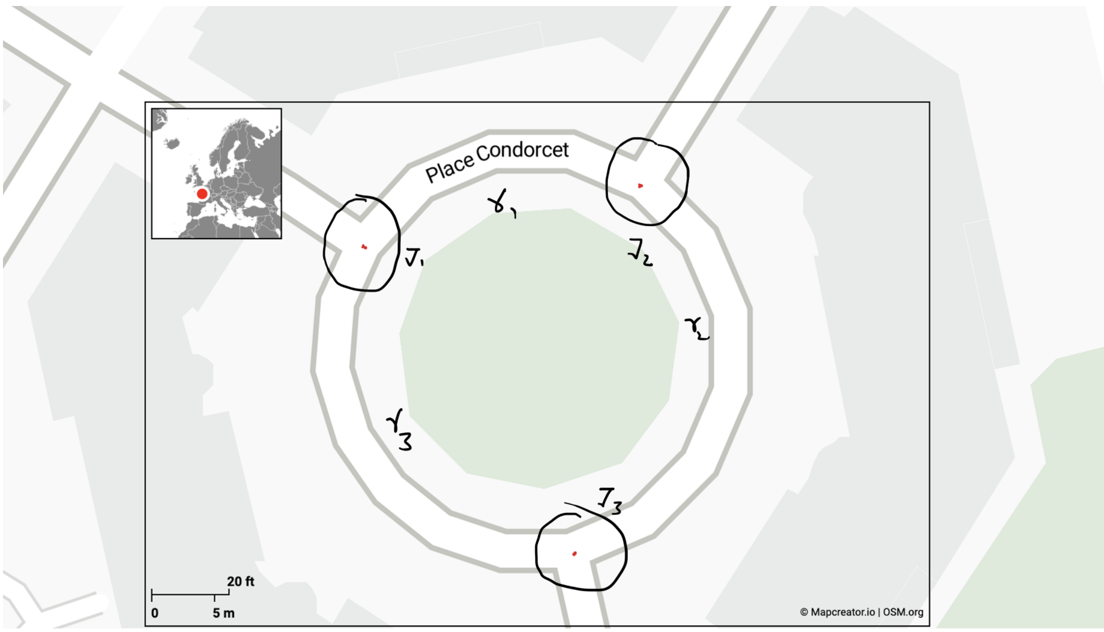


Fig 1.2: 2D representation of Junctions in roundabout

prediction of roundabouts. An attribute is interchanged as qualities or sub-features. Minimal research has been done using actual data from maps and field knowledge. Multiple roads connect via junctions. The road is a collection of lines; the connection of the roads r_1, r_2, r_3 and r_4 is via junction j_1, j_2, j_3 and j_4. The map feature is not roundabout in the figure, it's the closed-loop of roads. Fig 1.2 shows the roundabout having junctions, r_1, r_2, r_3 represent the road and j_1, j_2, j_3 represent junctions. Proposed work describes an approach to identifying the correct intersections which can be potentially part of the roundabout. Further, roads are collections of one or multiple lines.

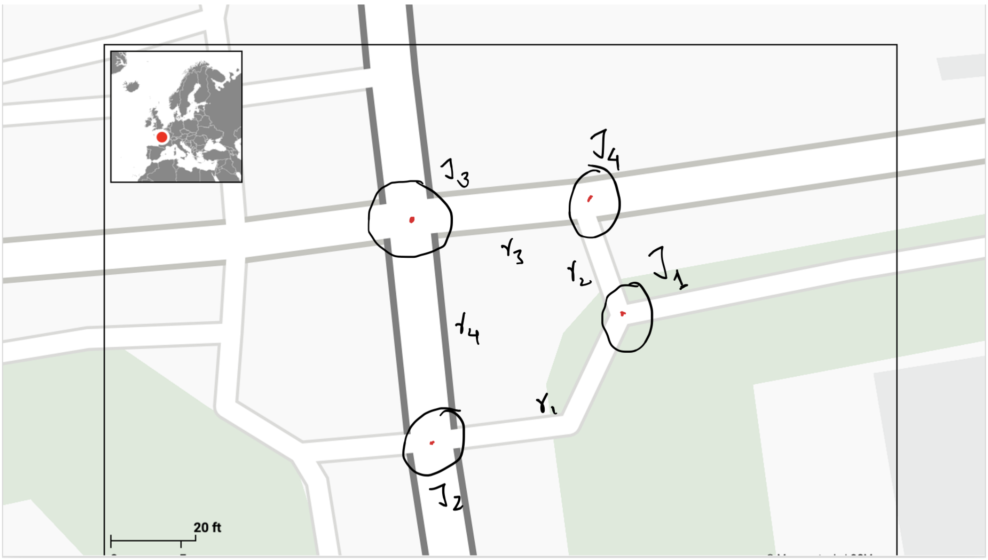


Fig 1.3: 2D representation of Junctions in Closed Loop

It is observed that any complex intersections on the road can be represented using simple geometry. A roundabout is nothing but a closed loop, a collection of links. In the past, the majority of work is globally done on image processing [11, 12], probe [13, 1] data, sensor [14] and traffic data analysis [15, 16] for the identification of roundabouts and junctions. Very less work is available on the useful raw map data with the map domain information.

When approximating a curve with the help of polylines, there's always a challenge of what would be the optimal number of lines that can correctly describe the curve. Precision required from this kind of conversion is very subjective as it depends on the kind of application one is looking to build upon it. In the current scenario, proposed work involves finding paths in a geographical map where a primary requirement is to convert a curve representing any kind of road into polylines in order to aid in navigation. In this case an approach is required which can give some flexibility with respect to the precision of the polylines in approximating a given curve. Such flexibility should be available for tuning at run-time in order to be truly effective.

There are several proposed techniques to solve the smoothness of graphs but not much has been done with real Map data for identification of kink after the 2D geometry is created. The proof of applied implementation of the proposed technique for identification of same. The spline technique has been used to interpolate the points from the geometry.

Splines are the mathematical representation of curves. In 2D map, roads is collection of multiple line segments, the curvyness of road is represented using small segment of lines and hence we get more geometry for more curvy road. The curvyness can be controlled by using any cubic function which should be continous at the geometrical nodes. Proposed work discuss about the identification and validation of a map feature (Roundabouts). The idea is to get the spline from te geomerical points if the roundabout and calculate the kappa value, reverse of the kappa value is the radius of curve for that point. All the points in the roundabout map feature is projected over the spline and for that particular point curvature value is calculated, which indeed provide us radius, then threshold is defined for 10 cm. Radius of roundabouts will be within the threashold value. Then this is cross validate with the validation technique where the bounding box of roundabout is calculated and hence the radius.
In chapter 3, authors have used the map data to detect the roudabouts using pure geometry. machine learning is used to get the results on the fuzzy part of detection. It describe the detail process of map making using geometries. Image processing is widely used in identifying the features from the road [17, 18]. Probe data is another source of input in creating and detecting the features in the map. Zinoune et al. uses probe data for identification of roundabout [19]. Other source of input may be the sensor, lidar data.

Splines are created using map data (Geometries). Spline creation is widely used in field of engineering for creation of smooth surfaces. Whole idea of spline creation is over the fitting of functions. The best used function for smoothing of road is cubic [19]. Many medical and engineering applications have proposed the technique of creating the feature in map for autonomous vehicle using three step process, B-spline is used to smooth the sensor data [20]. Berglund et al. obstacle on the road is detected using the spline technique [21]. They use spline technique to smooth the trajectory, path, road lane detection, 3d image creation and many more [22, 23, 24, 25].
For solving the problem of roundabout verifications and detection, B-spline in the NURB library is used. Once the values are extracted, projection of the points over the spline is done, there can be many technique to do the same but in the proposed methodology the nearest point is selected from the set of splines point and then the curvature value is calculated. The radius is reciprocal of curvature and hence the cross verification is done on the basis of the idea that the curvature value will be smooth for the roundabouts over

the span of complete spline points.
Brandin et al. use the idea of moving the vehicles on the roundabout and using the vehicular motion detection and stering angular change try to detect the roundabouts [13]. Finally, another approach is used to verify the results with approximate calcualtions, the technique used is creating the manual bounding box over the feature and calculating the best feated geometry i.e., eclipse or the circle, then radius is caluclated and verified with the spline calcuated technique. The result shows that this approach is easy and practically can be implemented with minimum infrastructure and has the success ratio of more than 92% detection.

1.1 Background

1. Roundabout is basically a Map domain problem.
2. Approaches for detecting roundabouts in maps are heavily dependent on looking at the problem from a machine-learning powered computer vision perspective
3. There are many techniques which help in identification of roundabout with probe and other GPS data.
4. Lines and kink calculation in the Map features are still manual process.
5. Automation to predict the Map feature with real time data is still not explored.
6. A practical and economical solution is the demand of time as we are moving towards autonomous devices.

1.2 Research Gaps

1. The existing solutions is not based on real raw map data.
2. All solutions are very much therotical in approch, a fresh practical approach needs to be considered.
3. Identifying and detecting roundabouts using real Map data is not yet explored.
4. Smoothing of line and identifying the kinks at centimeter level is still not achieved.

1.3 Objectives

The broad objectives of this research work are:

1. To study and review existing identification techniques for roundabouts.
2. To propose the new approach for identification of lines in the roundabouts from the curve.
3. To design and implement the framework for the identification of roundabout using deterministic/non-deterministic models.
4. To test and validate the performance of proposed framework.

1.4 Thesis Organization

Chapter 1: Introduction
This chapter provides an overview of the Map, starting with its background. It then delves into the roundabout and explores various strategies for identifying it. The chapter goes on to examine roundabout junctions in greater detail. Basic mathematical approaches for approximating curves are discussed, as well as identifying kinks in roads. Finally, the validation of the roundabout is accomplished by creating a bounding box.

Chapter 2: Literature Survey
The literature examines various approaches for detecting roundabouts, including the use of probe data, GPS data, and robotics. Additionally, probabilistic approaches are explored in relation to images data on the Map. Multiple machine learning models are applied to images and media data related to the Map, such as sensors and lidar. The chapter also explores the process of semiautomating the centerline geometry of the Map using images, which is currently in its early stages. Various spline techniques used to smooth graphs are analyzed, and the available research in this field is described.

Chapter 3: Detecting roundabouts in maps based on analysis of core map data
This chapter presents a novel approach that leverages core map data to supplement existing techniques, resulting in a significant reduction in the machine learning effort required. Consequently, our approach filters the training set to greatly reduce the scope of the ML problem, which enhances accuracy. At the heart of this approach lies the fact that data fields used to describe maps encapsulate geometric details

about map points. When interpreted correctly, these details can be used to identify various map features, including roundabouts. The proposed approach consists of two parts. First, an algorithm has been proposed that interprets core map data to identify roundabouts. The second part involves handling the remaining fuzzy parts using a machine learning approach.

Chapter 4: Detecting Junctions in Roundabout using map data
This chapter addresses the practical problem of determining whether a junction is part of a roundabout using the Map domain, which is a novel approach to the best of the author's knowledge. In a roundabout, the junction is the node that has entry or exit of roads into or away from it. Closed loops in roads are extracted by traversing each link and checking for common junctions using a deterministic algorithm. Loops can be formed not only by roundabouts but can also be present at any intersection of roads. Identifying the pattern and finding the right junctions of a roundabout is a challenging and exciting problem. The authors have solved this problem in two parts. First, data extraction is performed by collecting all the roads of America and Europe, and then features are extracted that depend on logical and domain knowledge. Finally, for the fuzzy part of features, a machine model is created, providing an efficient accuracy of 81%. This applied methodology is easy to implement and reduces manual effort for verifying and correcting junctions.

Chapter 5: Identify Kink In 2D Map Using The Spline Technique on Real Map Data
A Map is collection of lines in the form of linear and polygonal geometry in 2D space. In real Map these lines act as different features such as road, carto, point of interest...etc. This chapter focuses to provide solution on one of the basic, though practical, open problems; there is a need to identify kink in the formation of a Road in 2D Maps, which is basically the linear collection of the lines. The kink in road is inversely proportional to number of lines presenting them, which means more number of lines less is the kink.

In practical cases, there is a balance between them, which results in kink on certain points. Authors proposed a technique to identify those points by spline interpolation with length manipulation, where difference in the interpolated lines with points are calculated and accordingly the threshold is defined. The novelty of the work is yet no practical technique identifies kink in the Real Map data, which is deterministic and can be used with ease as best knowledge of the authors. With the threshold of 10cm, for Frankfurt and USA the identifications were 91% and 83% respectively.

Chapter 6: Approximating the Structure of a Curved Road using Poly-

lines for Maps and Related Applications

In maps and other path-finding applications, it is very essential to redraw real-life curved paths into their spline-based digital counterparts. Converting a curve in an image into a collection of continuous straight lines (polylines) in an image is quite an interesting challenge. Although many algorithms exist, which provide solutions to this problem, none are adaptive in nature or work specifically well in case of maps.

To address these issues it is required to have an adaptive and dynamic algorithm which can be tuned for precision, with significantly decreased complexity for a large number of cases, depending on the triviality of the concerned case. In this Chapter, two algorithms have been proposed which perform the operation of converting a curve into a collection of continuous straight lines in a customisable manner. For this purpose, parameterized equations have been implemented which take in a tuning value (τ) and return a result. Our implementation is tunable at run time with respect to the value of τ which will be taken from a pool of constants defined for specific scenarios. Significant performance improvements have been achieved on comparing the proposed method with existing approaches.

Chapter 7: Validating Roundabout Using Spline and Curvature for Map Data

2D Map is collection of pure lines which are connected to form different features like roundabouts, culdisec, buildings, special figures..etc. In real Maps these lines are created using the latitude and longitude. This Chapter identify and validate the roundabouts using the geometry of the Map features by spline technique where the geometries is pass over the non rational uniform B spline function to get the knot vectors and hence the new control points, then the curvature(κ) is calculated and hence the radius. Validation of the roundabout is done using a technique where bounding box of roundabout geometry is created and best fitted Eclipse or circle is carved.

Chapter 8: Conclusion and Future Directions

This chapter summarizes the conclusions drawn from the thesis along with the possible future directions.

Chapter 2

Literature Review

2.1 Determinisitic and Nondeterminisitic approaches for roundabout

Literature specific to this topic is very limited due to the closed nature of the problem as it is being dealt with mostly by industries in this domain. But a lot of work has been done related to the identification of roundabouts represented in the form of images using machine learning.

Bordes et al. proposed an approach which takes in satellite images and computes the radial and angular distribution of the pixels to extract geometric information from it thereby creating a descriptor named RAFA [26]. However, due to relying heavily on the (strong) hypothesis that roundabouts must have a certain shape, have certain objects present in the nearby zone and have a size in a fixed range, this approach gives sub-optimal results though they're much better than those of Zernicke moments Khotanzad et al. and Choksuriwong et al. which are often used as rotation-invariant features for object detection [17, 27].

Ravanbakhsh et al. has proposed an approach for automatically extracting roundabouts from aerial imagery by using pre-existing knowledge gathered from topographic information bases. This approach extracts 'central islands' with a buffer width of 1m and shows good results in roundabouts that are clear of trees or surrounding obstructions but suffers a lot in their presence [28].

Li et al. presented a four-step approach for automatic roundabout extractions from satellite images having very high resolutions [11]. This involves combining object-oriented extraction, support vector machines (SVM) classification and spatial relationship estimation. It makes full use of the image to extract as much information as possible. Watershed segmentation and merging are performed before rule-based extraction of roundabouts. SVM is used to classify roads based on the holes detected earlier. This approach has yielded up to 85% accuracy in various conditions though it cannot deal with certain special cases such as roundabouts that are directly painted on the road.

Muñoz-Organero et al. propose a general method to automatically detect road elements based on the use of GPS estimated locations [14]. The method takes into account the speed of vehicles and the total variation distance between the speed probability mass function (PMF) at each location. This helps guesstimate the presence of roundabouts based on their classification of possible roundabout driving patterns. However, it is a general approach and not strictly focussed on detecting roundabouts. It can detect other road features as well such as traffic signals and street crossings.

After labelling car manoeuvres using pre-recorded data from Volvo vehicles in the euro-FOT database, Jorge et al. provides a method of roundabout identification using a single signal, which they have defined as the yaw rate, and the use of the correlation coefficient [29]. They have noticed a relatively high positive prediction value (87.5%) due to the fact that some of the localizations regarded as roundabouts are in fact road constructions that mimic part of an actual roundabout.

This approach is promising. However, it requires more manual intervention than required from a good solution. Another approach given is to detect the presence of a roundabout while it is being driven upon. This unique idea, Zinoune et al. proposes to update the map using the vehicle's actual on-ground data itself [19]. This data is collected using the vehicle's sensors, e.g., odometer, gyroscope, speedometers and GPS. It uses bayesian-classifier based graphical pattern recognition strategies to identify the presence of a typical signature in the shape of the path followed by a vehicle when traversing a roundabout.

García Cuenca et al. provide another machine learning-based technique to build a predictive model and generate rules of action to detect roundabouts [12]. This approach consists of building a predictive model of vehicle speeds and steering angles based on data related to driver-vehicle interactions, the traffic environment, roundabout geometry and a number of available lanes obtained after some video processing. This requires additional hardware as well. The predictive model was built using supervised learning algorithms like SVM, linear regression and deep learning.

To assist a driver in the identification of a roundabout many things play important roles. Zhao et al. have done a field study and has identified that driver behaviour plays a significantly important role in deciding whether the vehicle will enter and move away from the roundabout or not [30]. They did this field study in the identification and modelled the human driver behaviour in interactions with the roundabouts. It was found that support vector machines proved to be an efficient classification method. With these results, they were able to tell whether a given vehicle will stay on or leave the roundabout. The study also states that the steering wheel angle and angular velocity also play an important role in this prediction Kathner et al. It is worth noting that the analysis was done on human behaviour and it turned out to be a good approach in dealing with the roundabout situation. In conclusion, the study confirm human behaviour changes when the vehicle approaches a roundabout situation.

Roundabouts are well known to cause fewer traffic accidents than traditional intersections, which is not the case for bicycles, unfortunately Hels et al. Often, crashes between vehicles and bicycles happen because car drivers can overlook bicycles Herslund et al. [31, 32].

Lots of predictions are done in the process of identifying roundabouts. One of the approaches is to use the graphical pattern recognition method as shown in Fig 2.1. In this approach, work is done on the data probed from real-life traffic to identify roundabouts using the Bayesian Classifier [19]. It is then proposed that it's possible to update traffic data continuously by probing for information through the movement of vehicles.

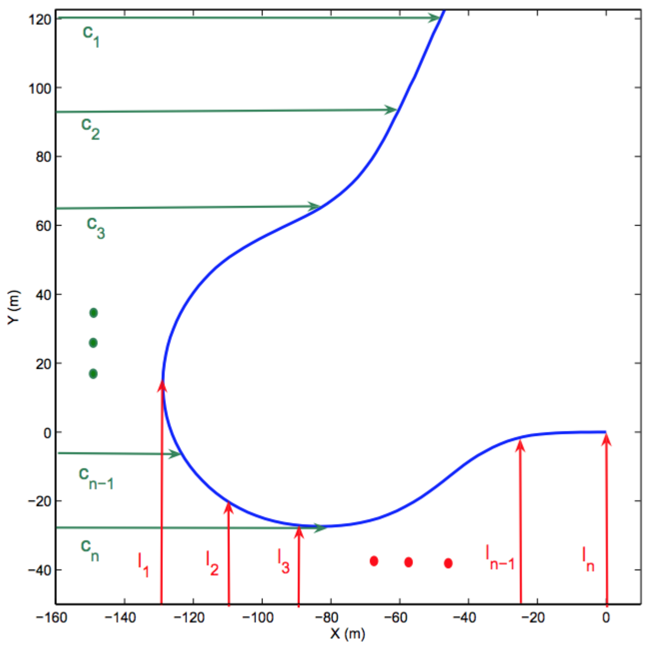


Fig 2.1: Roundabout using the graphical pattern method

Zinoune et al. has proposed a low computational complexity. Therefore, it can be easily incorporated into embedded applications [19]. This approach was tested on real-time data and the results were found to be quite promising. This is a good approach of using the traffic data to detect roundabouts and there are some good ideas described that allow for using the data probed from traffic in order to make a prediction.

Patent by Hofmann et al. describes the method of identifying a roundabout using data probed from traffic which are basically the latitudes and longitudes [33]. The main crux of the solution is using the concentration of the points to identify the path. The points are collected from different sources and are used to virtually recreate the real path.

The Brandin et al. patent describes a way of identification of roundabouts by having a special arrangement of vehicles, as shown in Fig 2.2, which acts as an information provider and aids in identifying the direction of the vehicle [13]. Any given vehicle is provided data from the vehicle preceding it and a chain of this information is passed to a mathematical algorithm to help identify roundabouts.

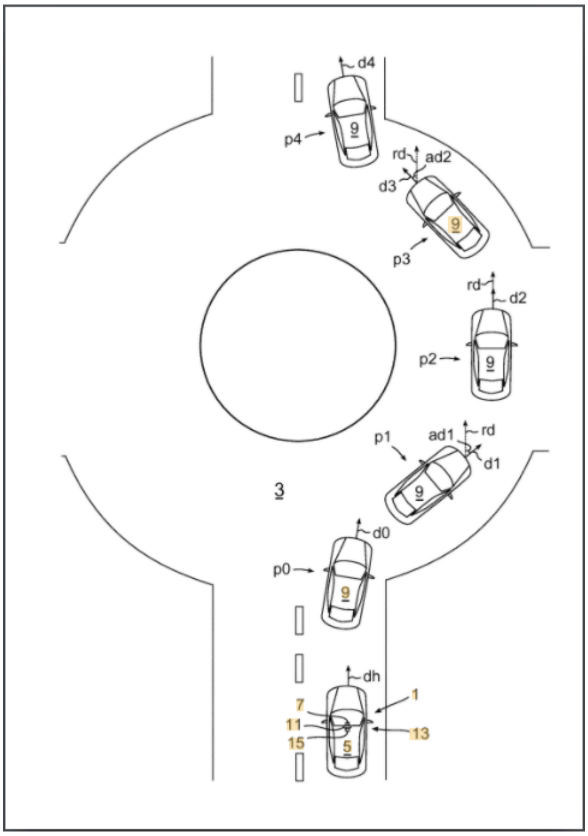


Fig 2.2: Vehicles in roundabout

Ali et al. proposed a three-wheeled robot to identify different junctions on the road [34]. Path planning and roundabout detection scenarios are determined based on sensor fusion of a laser range finder camera and odometry measurement. This is more like the real-time analysis of data.

Our exhaustive analysis of the existing literature points to the fact that hardly has this problem been attempted using a combination of methods or a divide-and-rule strategy which we feel is essential because it can be clearly seen that this domain cannot have a one-size-fits-all solution.

2.2 Roundabout Junctions

Literature review on this topic is minimal as this is a domain-specific problem. However, many approaches are used to identify roundabouts and normal junctions using image processing and probe data. Not much work has been done using domain knowledge and real-time data.

The majority of the Literature review is separated into three categories: Traffic and control impact on roundabout intersections using sensors. Image and probe data to identify roundabouts. Autonomous Vehicle impact on the roundabout junction to direct and control movements.

The traffic and signals challenge in urban roundabouts decreases the vehicle's travelling time. Cisneros et al. shows the results from a simulation of the surveyed data on roundabouts in the metropolitan area of Lima. The simulator used is VISSIM 9. Results show that the travel time where reduced between 9 to 81%. The data is processed on simulation at level F. Correlation between traffic and signal is vital in controlling travel time. The problem of controlling the vehicle in the roundabout when there is no traffic signal is targeted [35].

Zhang et al. have proposed control and merge systems, which are placed in the roundabout, where the vehicle is controlled for its speed before entering the roundabout, ensuring that the vehicle is collision-free. They proposed the optimal control system for the connected vehicle to have the best results [36].

Traffic is controlled in one way by introducing roundabouts. Shatnaw et al. proposed the impact of speed in the roundabout to increase vehicle performance and smooth movement. The simulator used is MATISSE, MATSIMLab and ITSUMO. Overall, the result shows that increasing the vehicle's speed at the highly congested part of the roundabout improves the traffic flow [37].

Supramaniam et al. introduces the regression model to identify the turning radius at the roundabouts [38]. Using sensors and other helping devices, it's still impossible to locate the exact roundabout turning radius. The data is collected by using the augmenting sensors. Methodology predicts the radius using the regression machine model with an accuracy of 99%.

The vehicle is tracked in roundabouts by using a Camera sensor and microprocessor. Eleuch et al. describes the approach and method for doing the same; Cameras used are intelligent cameras capable of vehicle identification and feature extraction [39]. Database is created from real-time roundabouts using these sensors. Now the extracted feature is used to find the trajectory of the vehicles. Finally, this approach shows that the vehicle can be tracked.

Sutharsan et al. proposed a prototype of the actual roundabout at colombo city at a peak congestion time [40]. The prototype is divided into two parts; the first part uses a vision-based smart camera to extract the traffic density and vehicle type and the second part uses a mathematical approach to predict the timing of signals at the peak congestion time at the roundabout. This prototype was tested on the existing system to control the traffic .

Bordes et al. proposed a technique for identifying roundabouts using satellite images has been proposed. Authors have used pixel density to calculate the radial and angular distribution of the pixel to determine the geometric figure from it [26]. Though authors have an assumption that the roundabout should be of a particular shape or structure, there are different optimization techniques defined in the [17, 18] which uses rotation-invariant features for object detection.

Brandin et al. propose an approach to identifying roundabouts using vehicles which pass through the roundabouts and are used to communicate the information among themselves. This approach needs the practical movement of vehicles in roundabouts. The car in the front shares data with the car behind and forms a chain of cars. This information is used to identify roundabouts. The Fig 2.2 shows this concept. Vehicles move in the entry node and move out from the exit node and they are used to getting information that helps infer the roundabout [13].

Ali et al. have used three-wheeled robots to identify the different junctions in the road. They use sensors and other devices to identify roundabouts [34]. Patent by Jehlicka et al. describes the use of probe data. Probe data is a combination of latitude, longitude and bearing data. Latitude and longitude are the coordinates that identify the location of a point, and the bearing angle defines the vehicle's direction. The Fig 2.2 shows that the probe data is used to identify roundabouts by having a sufficient number of probes and then after some threshold they identify the roundabouts [1].

Li et al. have described a stepwise approach to automatic identification from satellite images, which should have high resolutions [11]. They have used object-oriented extraction, support vector machines (SVM) classification and spatial relationship estimation. Segmentations and merging of the image is used to extract information. This approach cannot deal with certain exceptional roundabout cases.

Mu et al. focus is on automatically identifying road features from GPS data which is nothing but the coordinates i.e., latitude and longitude [14]. A new algorithm is developed

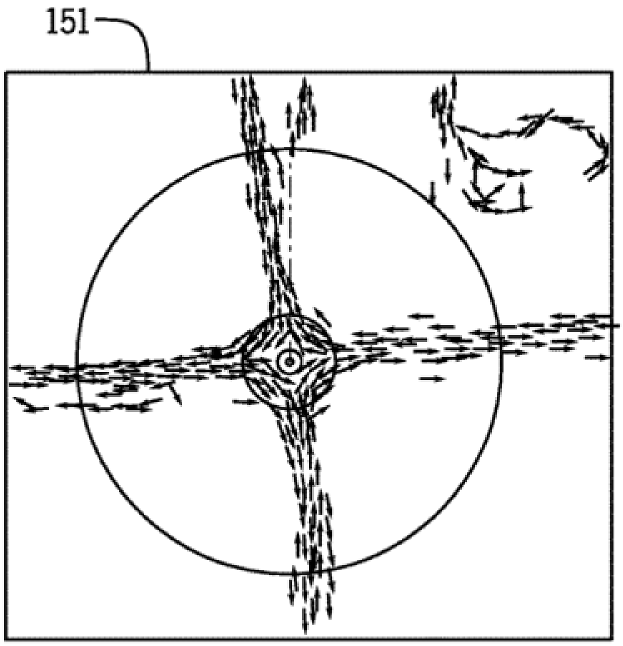


Fig 2.3: Probes in roundabouts as proposed in [1]

that uses the total variation distance instead of the statistical moments to improve the classification accuracy. They have validated this with detecting traffic lights, roundabouts and street-crossing in real scenarios with the accuracy of 75%.

Zinoune et al. has been proposed to detect roundabouts when travelling through vehicles [19]. They have performed experiments using real-time traffic data and applied a machine learning model over it. They also used the vehicles in build parameters i.e., odometer, gyroscope, speedometers and GPS. In their proposed approach they were able to detect the newly constructed roundabouts.

Wondimagegnehu et al. have focused the study in identifications of junctions on the basis of traffic data [41]. They have compared the Teklehaimanot junction using recurrent multilayer perceptron (RMLP) and multilayer perceptron (MLP). Results shows that the RMLP performed better. The crux of this study is to predict road junctions using traffic data.

Buades et al. has been proposed where they have analyzed video frames of road traffic for the tailgate of a vehicle in an urban road junction [42]. They have proposed a traffic light signal monitoring technique, vehicle tracking and road congestions using real-time video analysis. They have applied their technique at road junctions of the Huashan/Taian road, Shanghai, China, which achieved an accuracy of 90%.

Heung et al. a controller has proposed which can control the traffic flow at the junctions of roads [43]. It uses dynamic programming to switch the lights according to the traffic. It also uses the coordinate position of adjacent junctions for computation. The simulation has shown that the delay per vehicle is reduced, particularly when the junction capacity is at par with the traffic. They have also claimed that the hardware is not complicated and demanding. This is geared more towards controlling traffic at the junction.

Zhao et al. proposed the analysis of the vehicle which, while entering a roundabout will choose the upcoming exit junction or will stay in the roundabout [30]. They have considered the on-field driver behavior simulated data and found that support vector machines proved to be an efficient model. They also found that the steering wheel angle and velocity plays a vital role in the prediction of driver behavior at roundabouts. Authors focused on human behavioral changes when it enters the roundabout and indeed the behavior changes when inside a roundabout.

Garc et al. presents a machine learning technique which could be useful for autonomous vehicles to move in roundabouts [12]. They have processed information from open street maps and offline videos. This study generates the rules of actions for autonomous vehicles for completing roundabout maneuvers using information related to vehicle speed and the steering angle. Supervised machine models like linear regression, support vector machines and deep learning were used for making the predictions.

Ho et al. have proposed a fuzzy logic controller which controls the traffic as per its flux [44]. It is adaptive in the sense that as the traffic changes the fuzzy logic changes. They have compared the controller with the Pappis's controller and shared the results. In conclusion, it is about controlling the flow of fuzzy traffic using a fuzzy controller.

Waterson et al. has investigated the different algorithms over probe data [45]. They have described the impact of the probes on different junctions. They have further proposed a new junction control algorithm that makes use of the probe data and improves performance by minimizing the error especially at the junction.

Di et al. propose a new distributed interaction protocol is proposed which connects self-driving cars to move in traffic junctions [15]. They have done a simulation by forming a virtual platoon alongside the vehicle. This spacing is used to avoid collisions. The experiment was done with 3 autonomous cars which were using V2V communication based on the IEEE 802.11p protocol. The results confirms the effectiveness of the proposed protocol.

Setchell et al. has used video images to monitor the road traffic through which the junction entry and exit are identified [16]. Authors have used computer vision techniques to get the traffic parameters e.g. entry/exit statistic, journey times and incident detection. They used models to identify the vehicle in complex road images. This data is passed to a high level algorithm to fetch other properties. Finally, they have defined the system and the results of the testing with the real images are presented. There is no need to disturb the traffic as they're using video capture mechanisms which are easily installable at road junctions.

Ekinci et al. have defined a computer vision technique where an autonomous vehicle is capable of identifying complex road junctions and intersections at run time [46]. In the proposed model the information is extracted using two steps - find the boundary of the road surface and then locate the junction surface. This mode is used by the autonomous vehicle for guidance. The final results are presented with real road intersection images. This is another image processing technique which is highly dependent on the quality of the captured images.

Broggi et al. have defined an architecture and system configuration for autonomus vehicle to move in simple as well as complex driving scenerios like roundabouts [47]. The test results shows that the proposed applied approach was a open door for the autonomous vehicles to move on complex intersections.

Ravanbakhsh et al. shows a knowledge based system is created from geo-spatial data and then from those images a circular region is identified at the center of a junction. This approach was tested with black and white images of rural areas [48]. Comparing with Brandin et al. the proposed approach is practical and can be used directly on the raw map data and is, therefore, more efficient and easy to implememnt [13].

Majority of work has been done with images and traffic data. Yang et al. describe the simulated environment in which they have successfully identified junctions to find or estimate roundabouts or junctions in a road [49]. There are quite a few studies which use probe data to achieve the same. In the proposed work, we have used real raw data, which is used to make maps and used domain knowledge to solve this identification problem using machine learning models.

2.3 Kink Identifications

There are many techniques available for smoothing the graph, Van et al. describe the applied method to create smooth curve using spline. The authors use the angle as threshold to find kink and tested on simulated environment [50].

Peters et al. survey the technique of creating the spline inside the barrier which analyze multiple technique to create the linear spline out of the curve [51]. Zygmunt et al. propose technique for extracting the lines from the curve, here the curve is divided into the circular arc and on that basis the lines are calculated [52]. Liu et al. is the applied implementation of spline method to create shape reconstruction from the electrical impedance tomography (EIT). The authors results show that compared to traditional fourier transformation technique this technique is very efficient [53]. Gao et al. is the applied version of the spline technique over the blast wave monitoring devices. There is multiple literature for the applied version of the spline technique used in the different field of technology [54].

Dai et al. propose the technique to identify road from incomplete textures and geometry [55]. As the decision on mere geometry and data noises is fuzzy in nature due to image data. The accuracy of road is maintained by proposing new semi road-extraction method. Proposed technique uses different resolution images, different scenes and finally parsing them to extract road features like curvature, roundabouts. Experimentally this technique improved the correctness to more than 98% and human intervention was reduced by 2/3.

Jo et al. focus on approach to create map for autonomous car. Complete process is divided into three parts: Data gathering, Data Processing, Road Modelling. The data is acquired from GPS probe data, sensor data [20]. GPS data is not accurate, hence other supporting sensor data are used. Finally, B-spline technique is used to get the control points. Complete Experiment was done on the data generated from the RTK-GPS and sensors like (rate gyro, wheel speed sensor). It was verified from this data that map generation using the proposed techniques are more accurate and reliable for autonomous driving.

Berglund et al. focus on obstacle detection for heavy loaded machine used in mining process [21]. Using B-spline function, curvature of boundaries around obstacles are detected. From data provided by swedish mining company luossavaara- kiirunavaaraAb (LKAM), sesults showed that the proposed techniques results are better than the one used by the LKAM. This method overall improve time to 5-10% which include traveling, loading and unloading of the machine.

Andrade et al. focus on lane detection and lane tracking using images. Idea behind the process revolve around three steps: Image dimensionality, sharpness and area of interest distance from the vehicles [56]. Author use hough transformation and shape preserving spline technique to fetch the smoothness of the lanes. The results were compared with ground reality and strategy showed good accuracy on different levels, such as shadow, curves and road slopes.

Sun et al. propose four step algorithms for identifying irregular boundaries and obstacles on 3D lidar data [57]. The fitting of boundaries from data is done using spline technique. Testing is done on the "Xinda" autonomous vehicles using various road scenario over 1000 km distance. Performance was 93% above the average accuracy and processing is 36.5ms per frame.

Li et al. focus on the trajectory planning of autonomous vehicle in Urban area [58]. The approximate path is created using conjugate gradient non-linear optimization algorithm and cubic B-spline technique is used to smooth the trajectories. It has been tested on the traffic data and field experiment is carried out on autonomous vehicle, results show that the proposed technique has the capability to detect the trajectories following the road traffic.

Zeng et al. propose a novel DBO framework to analyze traffic data and come up with the trajectory following same [59]. To make is smooth and continuous B-spline technique is used. For selecting best trajectory, HAHP decision maker is developed. Finally, several scenarios are tested over the algorithm, with the lateral tracking error less than 0.2 m and the speed error less than 05 km/h.

Schreier et al. have proposed the grid based technique to map the area in the form of object, filter like interacting multiple model unscented kalman probablistic data assosiation (IMMUK-PDA) is used [60]. Further image processing is used to filter the boundaries. The system is tested over the experimental vehicle on the real traffic. Multiple proposed techniques are the applied implementation of spline on different branch of science and engineering fields like transportation, trajectory smoothing, power point tracking method for photovoltaic systems, switched reluctance machines, motion planning for car navigation,road lane detection using front camera [22, 23, 24, 25].

Wang et al. proposes the technique to recommend the vehicle to join the roundabout [61]. The method presented here depends on human behaviour. The model considers the significant parameters in deciding to move inside the roundabout. This is a driver assistance model to help drivers decide when traversing the roundabout.

Avcsar et al. uses the moving and traversal of vehicles to analyse the roundabout's trajectory; this is possible using the video camera on cars [62]. The complete process is divided into two parts; only the traverse vehicle is considered, and trajectory is made from this traversed vehicle. The detection is done using YOLOv4. Compared to the benchmark of 33 vehicles with a 3.704% error rate, the result shows an improvement of 1.571%. For road cycling, the path is calculated using different computer vision, and machine learning techniques. De at al. describes the automated process to calculate the penalized route. The map used is OSM (OpenStreetMap). The results provide the score for the legit paths which can be used in Road cycling for the organizers for safer rides [63].

In map domain, after the lines are created from the curve, the proposed methodology identify any kink present.

2.4 Polylines and Validations

Much research has been done with respect to converting curves into a collection of lines. Gribov et al. have described a new approach of approximate fitting a circular arc when two points are known [64].

$$\sum_{i=1}^{n}\left(\sqrt{(x_i - x_c)^2 + (y_i - y_c)^2} - r\right)^2 \approx \frac{\sum_{i=1}^{n}\left(\left((x_i - x_c)^2 + (y_i - y_c)^2\right) - r\right)^2}{4r^2} \tag{2.1}$$

They have solved the problem of fitting in O(1) time complexity. Their approach was to approximate the available formulae to adjust in such a way that the time complexity is reduced to O(1).

Zaverucha et al. have presented an algorithm to transform a path of polylines into a differentiable one comprising of straight lines and circular arcs [2]. They have proposed a novel algorithm to tackle this problem.

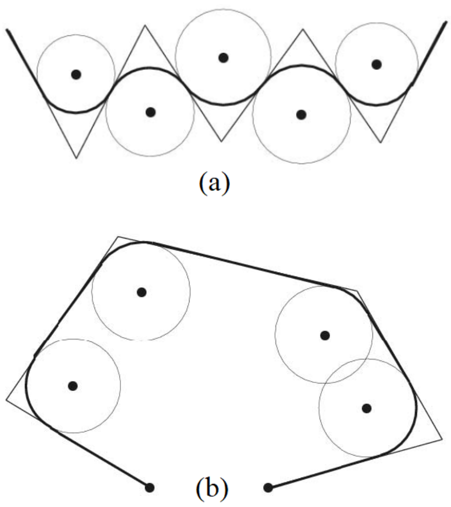


Fig 2.4: Worst case scenarios for algorithm proposed in [2]

They utilize a varying side arc to provide a smooth path for polylines and have proposed a couple of algorithms to help in calculating this path.

A very important contribution in this field has been given by Horng et al. which proposes a dynamic programming approach for fitting digital planar curves with line segments and circular arcs. They have discussed two subjective judgement criteria for perceptual significance [65].

i.e., we seek to minimize the perceptual error

$$E_p\left(N,M\right)=\min_{VCS,F}\sum_{k=1}^{M-1}e_p\left(v_k,v_{k+1}\right)\qquad\qquad=\min_{VCS,F}\sum_{k=1}^{M-1}\eta_k\left[e_k+\frac{C}{L_k}\right]\tag{2.2}$$

Where η_k,L_k and e_k are the type wieght, the segment length and the integral square error of f_k, respectively.

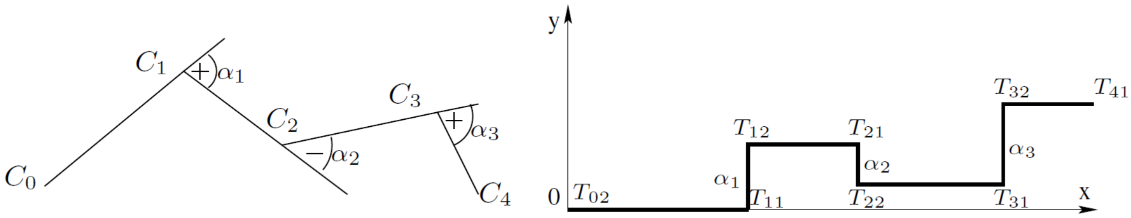
They have used dynamic programming to locate the optimal segmentation points.

$$e_{\mathbf{Line}}\left(s_i,s_j\right)=\sum_{s_l=s_i}^{s_j}d^2\left(s_l,v_{f_{Line}}\right),d\left(s_l,f_{Line}\right)=\frac{\left(x_l-x_i\right)\left(y_j-y_i\right)-\left(x_j-x_i\right)\left(y_l-y_i\right)}{\left[\left(x_j-x_i\right)^2+\left(y_j-y_i\right)^2\right]^{1/2}}\tag{2.3}$$

$$e_{Arc}\left(s_i,s_j\right)=\sum_{s_l=s_i}^{s_j}d^2\left(s_l,f_{Arc}\right),d\left(s_l,f_{Arc}\right)=R-\left[\left(x_l-x_c\right)^2+\left(y_l-y_c\right)^2\right]^{1/2}\tag{2.4}$$

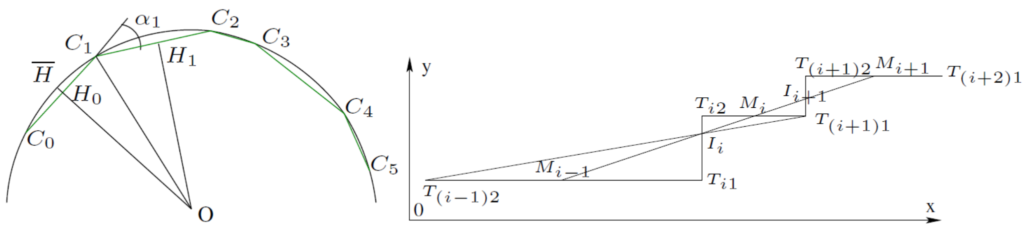
Their approach has inspired many researchers to work in this direction. One of the most significant one by nguyen et al. which works up on this approach to develop a more efficient approach [3]. The overall time complexity achieved in this method is O(nlogn) and they have done a comparative study against Horng et al. supporting their claim of a speedy implementation [65].

Most of these approaches are very complex and computationally heavy in their implementation. The proposed approach builds on these ideas but attempts to be simpler on the mathematical side without compromising on efficiency. It has two prerequisites - a method of dividing a curve into multiple curved subsections that resemble circular arcs and a method of calculating the radius given a circular arc. Much work has been done in these two fundamental domains as well.



(a) Input polygonal curve (b) Tangent space representation

a. Tangent space representation



(a) A set of sequential chords of an arc. (b) Its property in tangent space representation.

b. The chords in tangent space.

Fig 2.5: Approach described in [3] for converting curves to polylines.

Wan et al. have defined quite a rigorous approach for segmenting curves into straight-line segments and elliptical arcs [66]. Two types of corners were identified - corners and smooth joins. Adaptive smoothing was applied to the corners and the smooth joins were located using dynamic focusing fitting technique. The process was recursively repeated for the whole curve to optimize the result. It provides a solid heuristic way of achieving the segmentation requirement of the proposed approach with reasonable computational complexity.

A very different approach to solve the same problem had been given by Sheng-Feng et.al [67]. Their method used a knowledge-based non-linear thresholding approach on hand-made sketches and proved to be quite an innovative idea in curve segmentation. It did not involve heavy computation and was adaptive in some ways, making it very practical.

For the second requirement of the proposed work, i.e., calculating the radius of a given arc, one can go all the way back, when Euclid wrote about it. Heiberg et al., Fitzpatrick et al. provides a very basic geometrical way to find the center of a circle [68, 69]. The same idea can be extended to find the center of a given arc and therefore its radius. Further work has been done on Euclid's work mainly towards generalizing and formalizing it. For e.g. Avigad et al. have worked out a formal method to describe the propositions in Euclid in a system called 'E' which is very helpful in removing ambiguities and pushing towards an algebraic or, going further down the line, computing-friendly understanding of geometrical methods such as the ones described in detail in [70, 71].

Chapter 3

Detecting roundabouts in maps based on analysis of core map data

3.1 Introduction

The need for accurate spatial databases and their automatic updating is increasing rapidly. Road networks form key information layers in topographic databases since they are used in such a wide variety of applications. As the extraction of roads from images is still generally manual, costly and time-consuming, there is a growing imperative to automate the process. However, such a feature extraction task has proved to be difficult to automate. The problem for automatic road extraction lies mostly in the complex content of aerial images. Many authors uses approach to ease the complexity of the image interpretation task, prior information can be used [5, 6, 7, 8].

A modern roundabout is a type of looping junction on which road traffic travels in one direction around a central island and priority is given to the circulating flow. Robinson et al. shows that signs usually directs traffic entering the circle to slow down and give way to traffic already on it [9]. In this way, a roundabout helps manoeuvre traffic in a safe and convenient way. Few roundabouts are large enough to become famous landmarks or recreation squares in cities.

Generally speaking, roundabouts vary in radius from 12 to 30 meters depending upon the type, location and grade (state highway, national highway, etc.) of the road [10]. As per the foregoing definition, some road structures may look like roundabouts, geometrically, but may not really represent roundabouts in reality. There are several such special cases which give a hard time to machine learning tasks. Take the example of Cul-de-sacs-dead-end streets with only one inlet/outlet. They look like roundabouts but are not, as can be observed from Fig 3.1.

Another set of roundabouts, called turbo roundabouts, have legal dividers or small physical dividers that restrict lane changes and ensure smoother traffic flow. At turbo roundabouts, it is not possible to drive in circles and, in most cases, it is not possible to make a U-turn as shown in Fig 3.2.

There are more such scenarios which may not be common enough to have standard nomenclature. One such example is Fig 3.3 which is a special figure that looks like a roundabout but is not. In a similar structure, sometimes the central part is differently shaped so it could be a

Fig 3.1: Geometric representation of cul-de-sacs

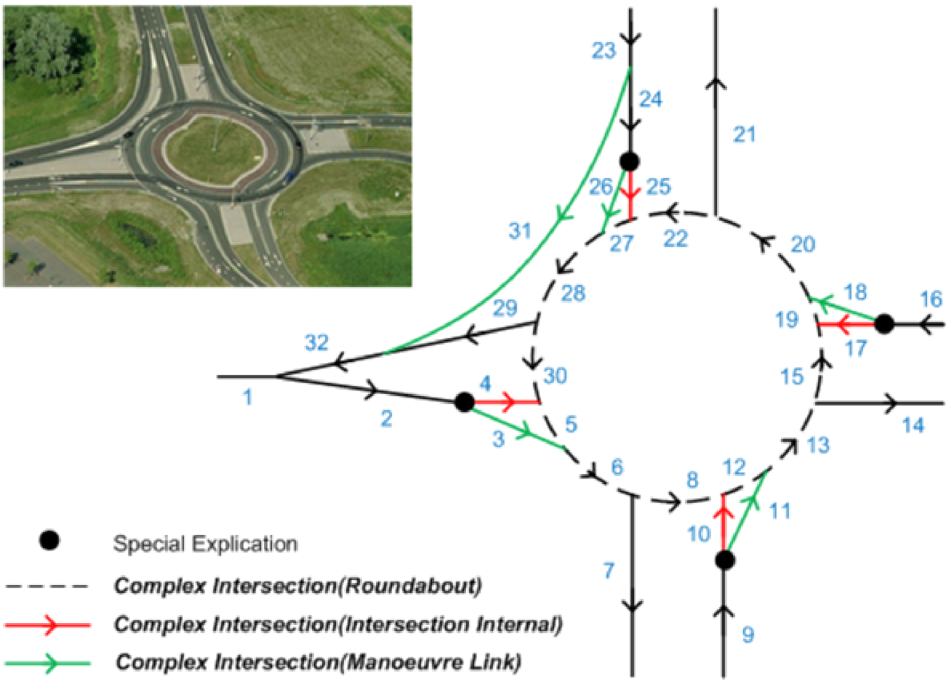


Fig 3.2: Geometric representation of a turbo roundabout

circle or a triangle.

Due to the existence of many such scenarios, it is difficult to programmatically identify roundabouts correctly if looking at maps as images.

After having worked with navigational maps for quite some time and realising the scale of the data that was being dealt with, the objective was to look for a problem which can be solved using a practical approach at the fundamental level to avoid complex solutions at the other end

Fig 3.3: A special case of a non-standard road structure that resembles a roundabout

of the spectrum. Some possible areas were point address identification, address range correction, roundabout detection and so on. The roundabout detection problem was found to be especially intriguing. Also, it happens to be one of the foremost fundamentally challenging problems in automatic map generation. Literature analysis revealed that it is a relatively less explored problem. Most of the research was focused on solving the problem from a machine-learning-based image-processing standpoint which had an unacceptable accuracy given the enormous scale of the problem.

Gerke et al. describes a way of using aerial images to identify roads at intersections using image processing and Boichis et al. describes the use of some machine learning models to extract knowledge from the images in identification of the complex intersection like roundabout. Brandin et al. has suggested a practical way of identifying roundabouts by arranging cars in such a way that share information amongst each other [5, 6, 13]. We considered all the approaches and have come up with a novel technique which purely uses domain knowledge.

The proposed approach begins with the analysis of the raw map data and is a deterministic one arrived at by thorough domain analysis. On analysing the huge data, it was observed that the problem of identification of roundabouts is not a single complex problem but a collection of many relatively simpler problems. The proposed technique takes this into consideration while analysing the data and helps identify the important attributes which separate an average circular 2D graph from a meaningful roundabout.

Implementation of the proposed approach revealed that it is possible to separate a non-roundabout from a probable-roundabout with perfect accuracy. However, from the detected probable roundabouts, few false positives were observed. Those cases were studied, and a machine learning model was created to identify the false positives from the probable roundabouts.

In summary, an unconventional approach to solve the problem was taken by eliminating the non-roundabouts with perfect accuracy and then using machine learning to process the remaining ones.

3.2 Motivation

Given the confusing special cases and the fact that the confidence of machine learning results is lower, the roundabout detection results from conventional methods are required to be manually verified. Manual verification process itself involves looking for certain basic characteristics in the map. These characteristics are mathematically identifiable if core map data is available and correctly formatted.

The proposed work is on the actual traffic data, which is basically a huge collection of numbers. On those numbers, links and other map objects are formed logically. Then they are validated using our business validation rules, which are needed to check the integrity of the data. The proposed approach is based on practically available data and using that data the best practical solution has been formulated which is being currently used to identify roundabouts in the map's application of Here Technologies. The primary motivation for solving this problem emerges from a basic business need i.e., making our maps application faster and better for our end users. It was found out that identifying roundabouts was quite a problem as it required lots of research and coding experience if it was approached using image processing and machine learning technologies alone. After looking at the real traffic data, we felt it was easier to approach the problem by compartmentalizing it and solving different types of cases in different ways.

3.3 Proposed Methodology

Initially, from spatial data, splines are calculated using one of the many available approaches such as to approximate fitting a circular arc when two points are known [64]. Their objective to reach the perfect solution with average time complexity of O(1), however, doesn't really materialise. In another approach, Zaverucha et al. have presented an algorithm to transform a path of polylines into a differentiable one consisting of straight lines and circular arcs by utilizing a varying side arc to provide a smooth path for polylines [2].

In our practical scenario, this spline calculation is done in the background for all available data which is then made available to us in the form of latitude-longitude pair for each point, information about adjoining points and the direction of traffic flow between any two points as shown in Fig 3.4.

The nodes are points which connect the links in the 2D representation of the Maps. The number of nodes varies as per the regions. However, for crowd regions, the maximum number of nodes may be more than 87434568, and for fewer mobility regions, the number of nodes may be less

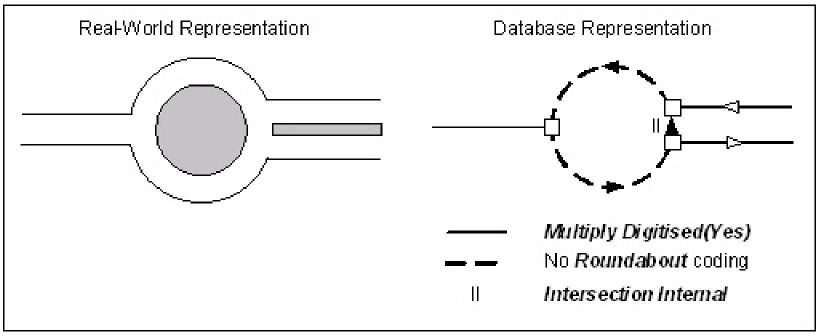


Fig 3.4: Collection of links formed from points available in the database.

than 3009170. In the proposed work, authors have considered real present nodes from the databases like Western Europe (approx. 55088475), Taiwan (approx. 3009170), North America (approx. 55601380), and Eastern Europe (approx. 87434568) nodes for efficient simulations point of view. Apart from latitude and longitude, a point can have z level which helps in identifying the elevation of the node from the ground and a bearing angle which helps in identifying the direction. Concerning about proposed work, the roundabout is on the same plane, and hence z level is of no importance. About bearing angle, that's fuzzy and different for each point which does not play important part and effects on efficiency.

Here's a generic example of the first kind of data we have collected through the (HERE) maps application. Any GPS data will also have similar information about points on the map.

```
node1:{                    node2:{                    node3:{
        id:1,                      id:2,                      id:3,
        lat:6189540,               lat:6189505,               lat:6189525,
        long:2684551,              long:2684533,              long:2684540,
      }                          }                          }
```

Given other spatial characteristics such as type of terrain, vegetation, etc., which depend on the use case, this simple data about the (latitude, longitude) pair of a point (called 'node' in our terminology) can be plotted on a map as shown in Fig 3.5. In this figure, x and y represent the latitude and longitude respectively of a node.

After this, traffic information is probed (automatically in the background) and that gives us access to the second kind of data that will be used in the proposed methodology. This data concerns itself with the relation between nodes in terms of the kind of traffic that flows between them.

For the direction of travel, there is need of a fixed terminology through which the nodes in the 2D maps can be represented, its similar to north, south, east and west. Authors agree that only

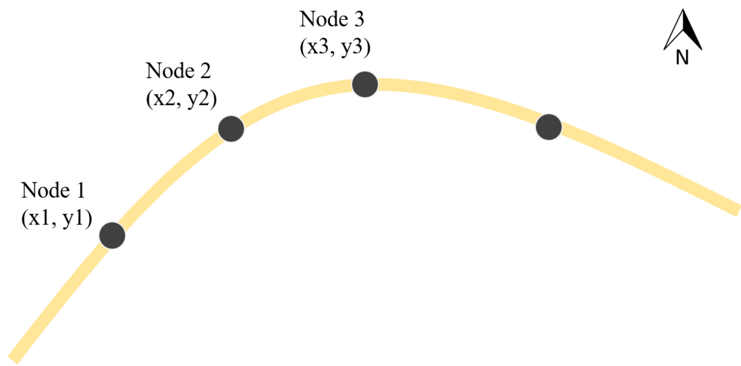


Fig 3.5: Plotting nodes on a map using latitude-longitude pair and spatial data

the east and west directions are considered as reference combinations of directions exist and the terminology followed here is if the node is on south-west, it's the reference node and if it's north-east, it's the non-reference node. That means the single direction traffic will be moving from south-west to north-east, north-east to south-west, west to east, east to west, north to south and south to north. For the case; where the traffic would be bidirectional that is in both directions, it will be from south-west to and from north-east, west to and from east, north to and from south. Roundabouts follow single direction traffic. The focus is on two cases of single direction link that is either towards or moving from the reference node. In reality, there are three traffic cases which are towards, from, or both represent as 'T', 'F' and 'B'. For a single direction, the representation is as 0 and 1 in the JSON. It's the nomenclature to identify the flow of traffic and can be changed. Here's a generic form of how this data looks like:

```
link1:{                          link2:{
      id: 2,                           id: 1,
      refNodeId: 2,                    refNodeId: 2,
      nRefNodeId: 2,                   nRefNodeId: 2,
      dot: 0,                          dot: 0,
     }                                }
```

In this data, there are two new, but generic, concepts which can be identified from any traffic information available. One is that of reference and non-reference nodes. In map terminology, nodes located in the south (or east, in case of horizontally oriented links) direction from a given node are called reference nodes, whereas those located in the north (or west, in case of horizontally oriented links) direction from a given node are non-reference nodes. Reference and non-reference nodes are relative terms and can exist only for a pair of nodes or in other words, when a spline link is defined between two nodes. As explained in the literature, there are various techniques to calculate splines from curves. Thus, these links represent straight lines joining two points which, in this case, are nodes on a map. This is shown clearly in Fig 3.6.

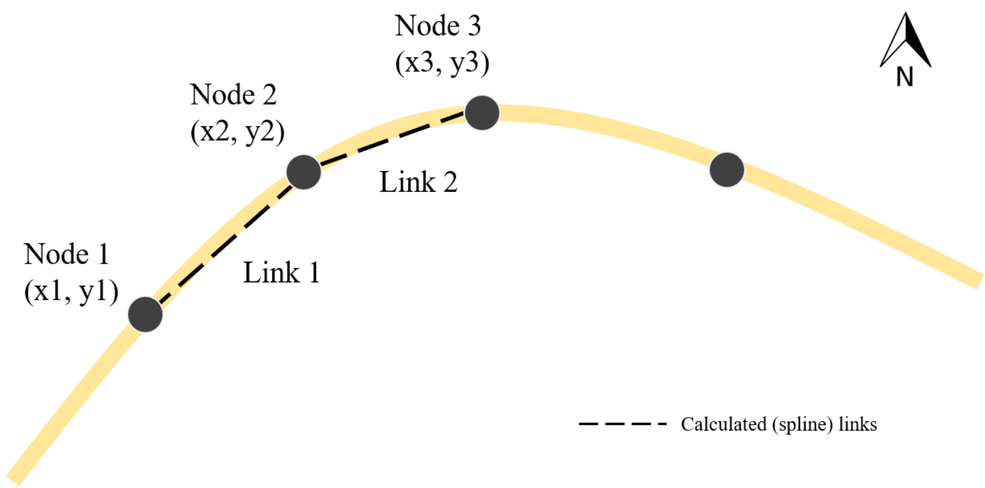


Fig 3.6: Representation of links that connect two points (nodes) on a map

The second concept is 'direction of travel', abbreviated as DOT. The direction of travel is a binary value which represents the direction of flow of traffic. If the value of DOT is set, i.e., it is 1 or TRUE, then that means the traffic has been observed to move from the reference node to the non-reference node. Conversely, if traffic has been observed to move from non-reference node to the reference node then the value of DOT will be unset or set to 0 or FALSE.
Fig 3.7 represents the DOT in case of the currently given data. Consider 'link 1' in which node 1 is the reference node (located southward) and node 2 is now its non-reference node (located northward). Further, the value of 'DOT' is unset therefore the traffic must be moving from the non-reference node to the reference node i.e., from node 2 to node 1.

Similarly, consider 'link 2' in which node 2 is the reference node whereas node 3 is the non-reference node. The value of 'DOT' is the same as in the case of Link 1 therefore traffic must be moving from node 3 to node 2.

These two concepts are central to the proposed methodology. We have researched and come up with all the conditions and scenarios that such data could represent. Thereafter, this problem can be divided into deterministic and non- deterministic types which are explained in the following subsections.

3.3.1 Phase 1 - Deterministic Approach

Closed loops are just a mathematical form of roundabouts which means that every roundabout should be having a closed-loop. Therefore, the first step is to check whether a set of nodes form

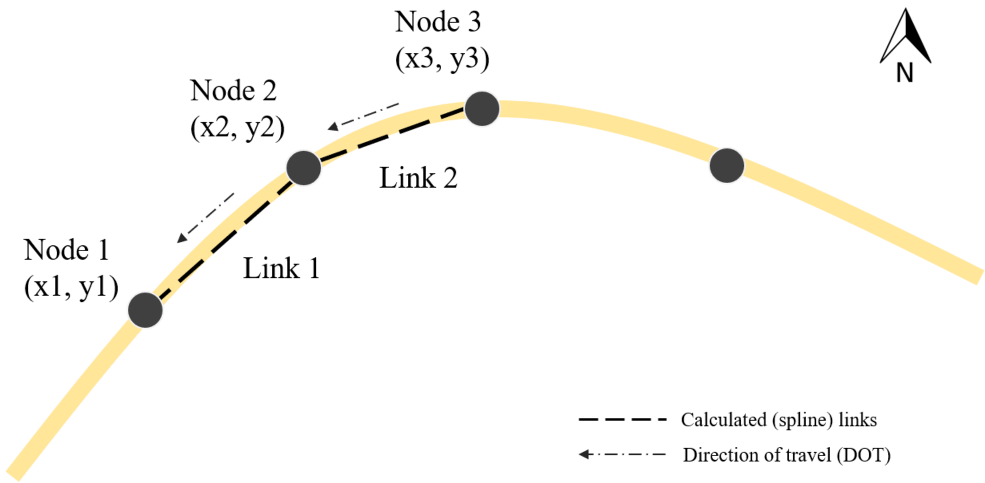


Fig 3.7: Representation of direction of travel between two nodes based on data

closed loops. Many approaches are available in geometric studies for Plotting a set of nodes described by latitude-longitude pairs on an x-y plane [72, 73].

However, not every closed loop would be a roundabout as explained earlier. The challenge is to identify this pattern based on domain and field study which provide some extra parameters to perform this detection, in addition to the primary characteristic i.e., direction of travel. Every closed-loop identified in the first step is, therefore, passed through the following algorithm using the properties that have been identified with respect to roundabouts.

1. Start with a link representing two nodes on a closed-loop.
2. Consider one of the two nodes, say, X. It will either be a reference node or a non-reference node. Let's assume it is a reference node.
3. Check direction of travel between X and the other (assumed non-reference) node, say, Y i.e., 'dot' value of the link.
4. Take the next connected link to Y on the closed-loop (now the reference node) having, say, Z as its other (non-reference) node and check if the 'dot' value is the same, given that Z is still a non-reference node for Y. If Z has become a reference node then the dot value should be the opposite of the foregoing dot value.
5. If the dot value in the previous step is different from the expected value, it can directly be stated that this is not a roundabout.
6. After completely passing through all the nodes in a closed-loop, check the following extra parameters of each link on the closed-loop -

a. Is any link on the closed-loop marked as 'ControlledAccess'?

b. Is any link on the loop marked as having 'UnpavedRoad'?

7. If any of 6(a) or 6(b) are true, then this cannot be a roundabout.

3.3.1.1 Controlled-access roads

Controlled access roads are roads with limited entrances and exits. Fig 3.8 that allows uninterrupted high-speed traffic flow. For example, the interstate/freeway network in the United States and the Motor way network in Europe. If a road is a controlled access road then roundabout is not possible. In the domain data for links, a special attribute is added to specify this parameter.

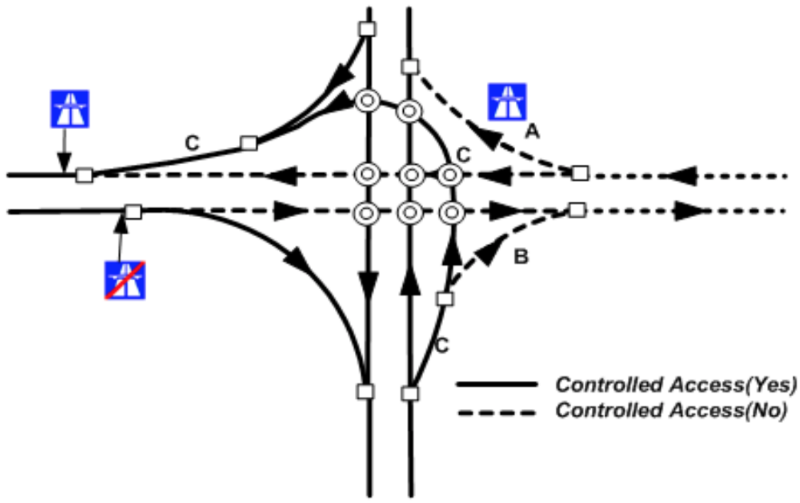


Fig 3.8: Controlled access on the road

Paved (cemented) roads:
Paved roads are made of materials that create a solid surface. Therefore, paved is a necessary condition for such roads which could be a part of roundabouts. In the domain data for links, a special attribute is added to specify this parameter. This makes it possible to identify whether a closed loop is possibly a roundabout or not.

There were two more conditions that were included in the initial algorithm in step 6. These conditions are given in the following subsections.

3.3.1.2 Angles between links

Initially, the algorithm was designed for calculating the angle between the links but later on it was found out that this problem is quite fuzzy in nature. So, this property will be fine-tuned in the second phase. The approach for identifying roundabouts with this approach is as given below.

1. Select any two links from the list
2. Find the starting point which can either be a reference node or a non-reference node
3. Take the link attached to the starting point
4. Calculate the angle and check whether it is acute or obtuse. In case it is greater than 180 degrees, find its 360-degree complement and check for the acute/obtuse condition.

The basic idea is that the angle between these links, when connected in a closed-loop, should be obtuse. Otherwise, it cannot be part of a roundabout. Since this data is plotted manually there is always some error or fuzziness associated with it. As stated earlier, this was later included in the fuzziness parameter as there wasn't much clarity about which angle range should be considered as a part of the roundabout.

3.3.1.3 Number of links at the nodes

None of the nodes should be having more than 3 links (connected lines). This was a very important attribute which was calculated and identified from the field study. However, later it was included in the non-deterministic approach due to lack of clarity.

These results have been cross-checked with a database of different regions like North America, Taiwan, etc.

3.3.2 Phase 2 - Non-deterministic Approach

Some scenarios might be fuzzier in nature. Identification of a roundabout is quite deterministic in nature but confirming the presence of one in border cases needs a probabilistic approach and for that machine learning techniques have been used to resolve the remaining fuzziness. For e.g. Fig 3.9A is not a roundabout as the links associated are just connecting two arms though the direction of travel criterion is satisfied, Fig 3.9B is a special acute roundabout while Fig 3.9C is a pedestrian cut roundabout.

So, further research was done in order to find relevant features and, after that, different models were applied to validate the results using real data. Different models like the random forest, logistic regression, decision tree and support vector classifier were used to come up with the predictions. The exact details of these models are beyond the scope of the thesis.

On analysis of the available data, it was concluded that for digitizing a roundabout feature, the following criteria need to be taken into consideration –

1. Physical roundabouts (looking at the imagery) must be present with a minimum diameter of 25 metres. But this could be fuzzy as shown in Fig 3.10.

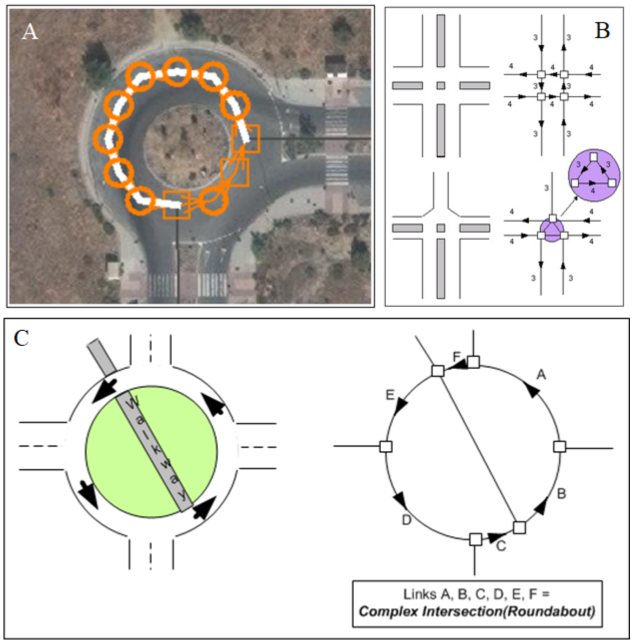


Fig 3.9: Special cases of a roundabout

2. Minimum speed category needs to be applied.

3. Links in the roundabout must be unidirectional.

4. Number of intersections must be more than 2. (>2 links should be connected to the roundabout)

Once the model was made, it was found that the results returned were not accurate enough. So, a new approach was devised by understanding machine learning needs to be applied on a group of links as a roundabout instead of on a single link. After this study, the following attributes were identified instead.

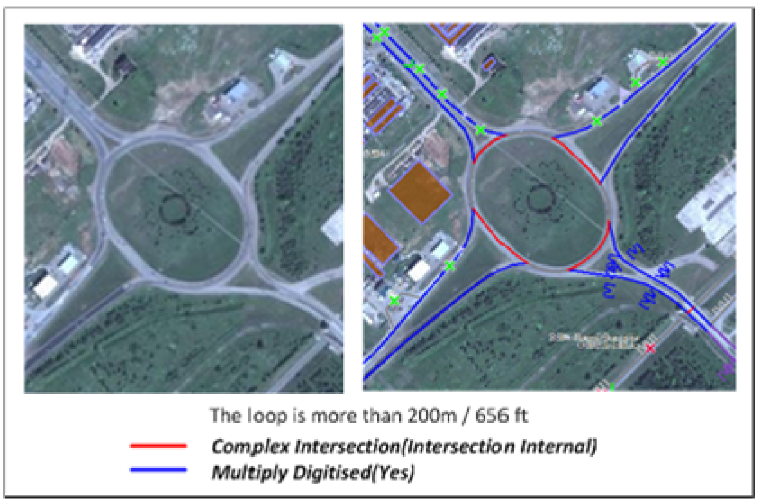


Fig 3.10: Circumference of a roundabout is fuzzy in nature

3.3.2.1 Loop

Loop is a basic requirement for a roundabout. If a loop is present, only then can it can be checked whether a given section is a roundabout or not. In case a loop is not present, it can be easily said that it is not a roundabout.

3.3.2.2 Total Links

This is the parameter that provides information about how many links are needed to form a loop since every map feature is purely geometric in nature i.e., collections of lines, polygons and other geometric constituents.

3.3.2.3 Minimum angle and maximum angle

This component is fuzzy in nature because it is prepared manually by coders and there are no common patterns. But it can be concluded from the data that this is majorly obtuse although some rare cases have been found where it is acute.

3.3.2.4 Maximum attached nodes

This is the attribute which indicates whether the links on a node are crossing any threshold or not.

3.3.2.5 Roundabout

This variable acts as a confirmation whether the data is a valid roundabout or not. This variable can either be true or false.

There is no benchmark dataset as this is a novel approach to solve the identification using the domain knowledge. Structure of dataset is:

Table 3.1: Raw data representations

Isloop	**Total Links**	**Minimum Angle**	**Maximum Angle**	**Maximum Attached Node**	**Roundabout**
0	6	90	177	3	0
0	5	60	141	3	0
0	3	26	175	3	0
1	5	110	174	3	1
1	4	85	150	4	1
1	6	73	168	3	0

The complete sample of unique 669 rows selected. The data roundoff and sanitization was done on it, Finally the feature correlation is presented in Table 3.2.

Table 3.2: Correlation of features

	Isloop	TotalLinks	Minimum Angle	Maximum Angle	Maximum Attached Node	Round-about
Isloop	1.0	0.476381	0.577648	0.310864	-0.195676	0.746011
TotalLinks	0.476381	1.0	0.583262	0.422006	-0.170654	0.628879
Minimum Angle	0.577648	0.583262	1.0	0.405263	-0.286125	0.806069
Maximum Angle	0.310864	0.422006	0.405263	1.0	-0.21219	0.400859
Maximum Attached Node	-0.195676	-0.170654	-0.286125	-0.21219	1.0	-0.322371
Round-about	0.746011	0.628879	0.806069	0.400859	-0.322371	1.0

Table 3.2 describes the correlation between the features. The Observation that is loop feature which checks the continuity of directions in the collection of links forming the roundabout is highly related to the decision of roundabout. The total links feature which identifies the number of links in the roundabout is highly associated with the roundabout check. The exciting feature, which is the minimum angle formed among the links in the roundabout is the most related. The Least related one is the minimum attached node, which checks the valency of the participating nodes in the roundabout. In most entry and exist cases of roundabouts they are three, but sometimes it has valency up to five. Then these complete features are passed through the various machine learning models.

3.4 Failure Rate

The process of identification of roundabouts is bifurcated into two parts - the deterministic and non-deterministic approach.

3.4.1 Failure context along with reasons

Failure case for deterministic approach: In the ingestion stage, raw data is passed through some validations, when we get the practical data, we have some validation run over it to check its credibility. With the deterministic approach, the failure rate totally depends on the accuracy of the data. We have the assumption of having the correct data, which we get through thorough validation.

Failure case for non-deterministic approach: For the non-deterministic approach, fuzzy attributes that have a significant failure rate were identified which are described below:

1. **Angle between links**: In certain extreme scenarios, such as the one shown in Fig 3.11, it can be seen that an acute angle is possible between the links even though this is an actual roundabout.

Fig 3.11: In a pear-drop roundabout, the angle between links is acute

2. **Size of a roundabout**: There is no straightforward answer to this question and therefore it cannot be deterministically stated. Certain roundabouts on highways are much bigger than an average roundabout, as shown in Fig 3.10, whereas some in smaller towns are much smaller. Therefore, it cannot be deterministically stated.

3. **Pedestrian crossing in Roundabout**: This is again one of the cases where there was a direct pedestrian walk over the roundabout and is a failure case. Though rare, if there's a direct pedestrian walkway over a roundabout, as seen in Fig 3.9, it fails our deterministic testing methodology.

4. **Speed Limit**: This factor varies across cities and countries therefore it is difficult to put it in an identifiable bracket. Therefore, this information was not considered while making predictions.

3.4.2 Classification Analysis Using Machine Learning

Raw data was coded into binary form that is 1 or 0, in which 1 indicates the positive case and 0 indicates the negative case. Fig 3.15-3.18 show the clear classifications of data between the feature level. After that, the data is split into random sets i.e., training set and prediction set. Finally, we were able to validate it using cross validation technique on the data and the results were quite promising with logistic regression providing the best result. Few parameters which were considered for the analysis are:

1. Gini Coefficient: It measures the inequality in the distribution, the range is between 0 and 1 where 0 means equality and 1 mean inequality

$$Gini = 2 * AUC - 1 \tag{3.1}$$

2. AUC (Area under Curve): It is the measurement to check the quality of the models and its value ranges from 0 to 1. The model is considered good when its value is closer to 1.

3. Accuracy: It represents the accuracy of the model and is between 0 and 100. It is represented by
$$Accuracy = (TN + TP)/(TotalData) * 100. \quad (3.2)$$

4. Sensitivity: Sensitivity is also known as the true positive rate. It is represented by
$$Sensitivity = TP/(TP + FN) \quad (3.3)$$

5. Specificity: Specificity is Known as the true negative rate. It is presented by
$$Specificity = TN/(TN + FP) \quad (3.4)$$

where, TP is true positive, FP is false Positive, FN is false negative and TN is true negative.

Table 3.3: Performance evaluation of various models

SN	Model Name	Gini	ACC	AUC	Spec	Sens
1	AvNNet	0.60	80.35	0.80	0.82	0.76
2	Glmboost	0.64	82.23	0.82	0.75	0.78
3	GAM	0.70	85.08	0.85	0.81	0.83
4	Blackboost	0.72	88.15	0.86	0.87	0.88
5	Gamboost	0.74	89.25	0.87	0.95	0.73
6	SVM	0.74	90.36	0.87	0.95	0.92
7	Decision Tree	0.90	95.55	0.95	0.97	0.90
8	Logistic Regression	0.94	98.50	0.97	0.98	0.94

We have predicted from different models; the logistic regression model performs better among all of them. Each model was trained using the complete dataset.

The feature selection is described in the above sections. Overall the data set was validated across 5 iterations. It can be inferred from the logistic regression, as presented in Fig 3.12, that for the 5 runs the accuracy range is between 97% and 98%. We did that with all the models and logistic regression provides best results among them which can be inferred from Fig 3.14. The sklearn function train_test_split was used for splitting the data arrays into 2 subsets- training data and testing data.

Using a sample set of data, we have plotted the confusion matrix. As can be seen there are two false positive cases and two false negative cases. However, an interesting observation is that the

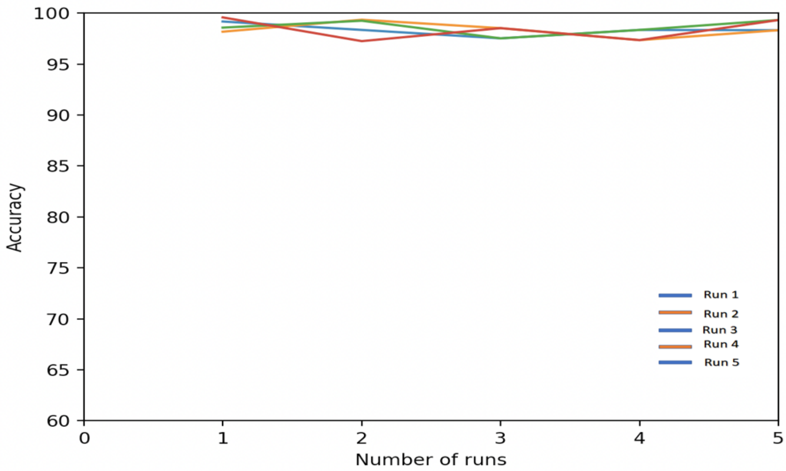


Fig 3.12: Repeated 5-fold cross validation of logistic regression

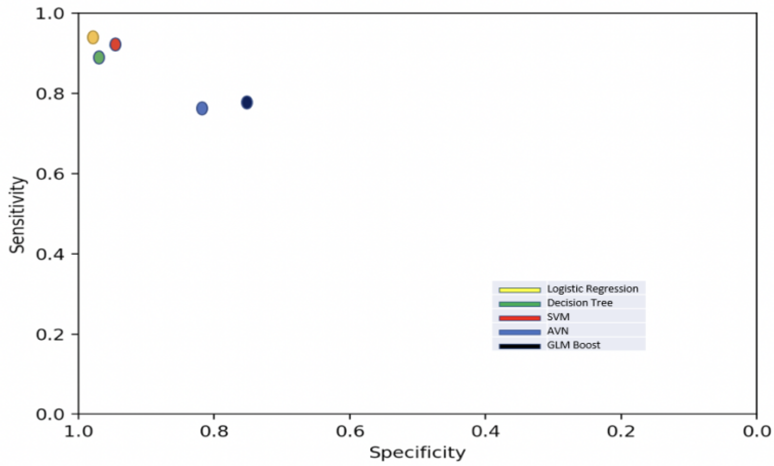


Fig 3.13: Relation between sensitivity and specificity of different models

actual data which we wanted to be negative was the same as the model predicted it and it's the same with the false positive cases.

Confusion matrix:
TP: true positive, FP: false Positive, FN: false negative, TN: true negative.
An: Actual No, Pn: Predicted No, Py: Predicted Yes, Ay: Actual Yes.
TN =50, FP=2, FN=2, TP= 52.

Table 3.4: Values of parameters: TN, TP, FP, FN

(AnPn)50	(AnPy)2
(AyPn)2	(AyPy)52

We train the model with the data in iterative mode and the result is shared below:
Precision is 0.9831245 and Recall is 0.94313

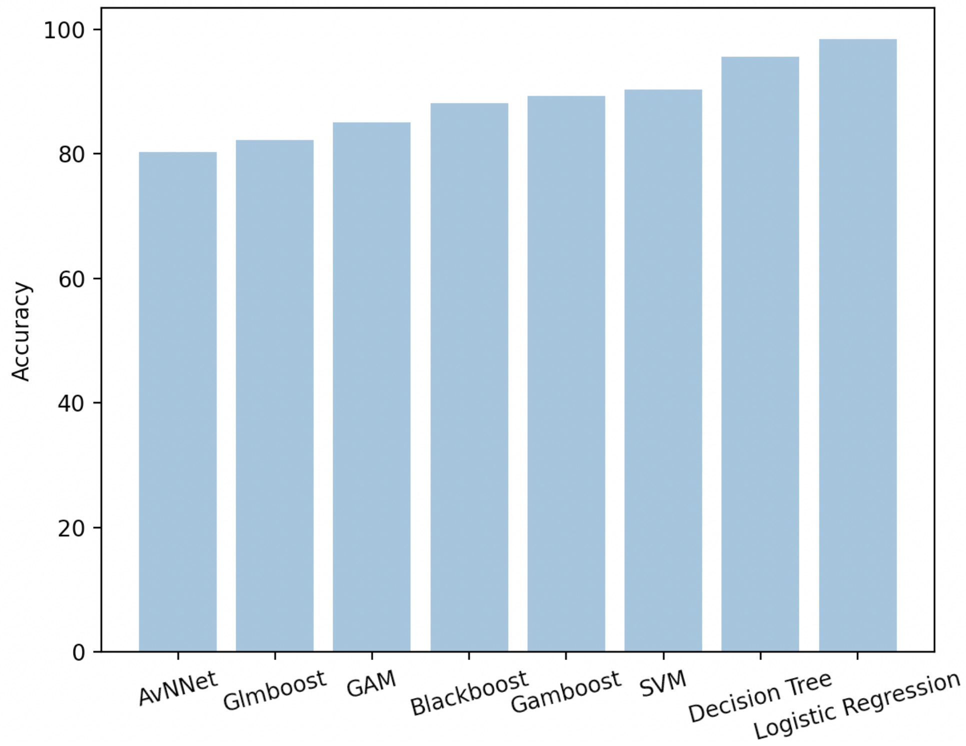


Fig 3.14: Relation between the accuracy and different models

Validation of the data was done on the random sets derived from the main data and folded multiple times to get the results which validate our models. Further, the data was cross validated with cv=5, which means the data is folded and tested. The accuracies in each iteration were 0.99173554, 0.98347107, 0.97520661, 0.98347107 and 0.98319328 respectively.

Finally, the cumulative accuracy is: 98.51%. Upon comparing the results, it can be seen that logistic regression has provided better results.

3.4.3 In-Depth Analysis and Discussion

Several previous works have attempted to solve the problem of identification of roundabouts. It is insightful to compare the proposed approach with the existing approaches.

Upon comparing Brandin et al. and Ali et al. it's interesting to require that both approaches require on-field analysis of roundabouts. In the former work, a novel approach is defined where a special arrangement of cars can give information on the presence of a roundabout whereas in the latter, a robot uses sensors and other meters to get real-time analysis. Practically, Brandin et al. is easier to implement. On comparing the proposed approach with Brandin et al. proposed work has the advantage of enormous data and the technique defined is not required on field analysis. Also, the technique is not feasible while detecting the roundabout as it needs physical presence whereas the proposed approach directly deals with the data and with the domain knowledge [13, 34].

Ali et al. uses robots and sensor data to identify the roundabout in real time. This approach is very probabilistic compared to the proposed technique, as this is theoretical in nature and cannot be used practically. A wheeled robot is used for data gathering which shows multiple errors such as collisions, missing data & feasibility, whereas the proposed approach uses available data on which we run domain specific algorithms to get results [34].

Similarly, Bordes et al. which basically works with satellite images and gets information from it by purely considering the pixels and the information inferred through that image. Comparing this with the proposed technique this is better as it's deterministic in nature. The whole purpose was to avoid the processing of images and cannot be used to get concrete decisions as Bordes et al. undergoes probabilistic analysis and estimation to get results whereas the proposed approach uses only domain knowledge to get better outcomes [26].

Analysis of the literature concluded that many approaches consider images as the base input to the system in identification of roundabouts. All that is needed is the mechanism of image processing which is probabilistic in nature. This is because it is difficult to have images which can be accurately processed due to factors such as pixel quality, overlapping images and so on.

Moving away from the probabilistic approach, we used domain analysed data which is practically used while forming the map and through this a novel approach was implemented which can solve this problem in a deterministic way.

Despite the enormity of the data, some similarities were observed in certain attributes upon thorough analysis of the cases that were identified to be roundabouts. Many such attributes were discarded to derive the most significant ones in order to get a deterministic solution for perfectly segregating the non-roundabouts, roundabouts and probable roundabouts from ordinary loops perceived from the data. The probable roundabouts were further processed through a machine learning model by taking into account one of the less significant or 'fuzzy' attributes.

Below are the results of the data which are filtered after our deterministic technique i.e., around 20% of total cases. So, machine learning techniques was implemented on this data set which has the accuracy 98.50746268656716% as can be seen in the following set of Fig 3.15 to 3.19.

Fig 3.15 shows the clear difference between the loops and roundabout, as its observed that if the collection of link form loop and don't have the continuous direction of travel on them, then it is not the roundabout. On the other hand, if the collection of link form loop and have the perpetual direction of travel on them, it may or may not be a roundabout.

Fig 3.16 shows the relations between the angles in the roundabout, the links which are part of the roundabout connected via a common node have an angle. These angles are fuzzy and cannot be differentiated. The above figure shows the relation between the maximum and minimum

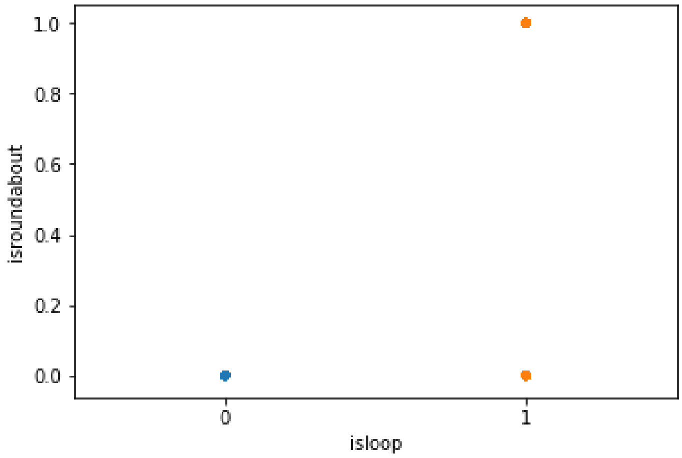


Fig 3.15: This figure shows relation between loops and roundabout

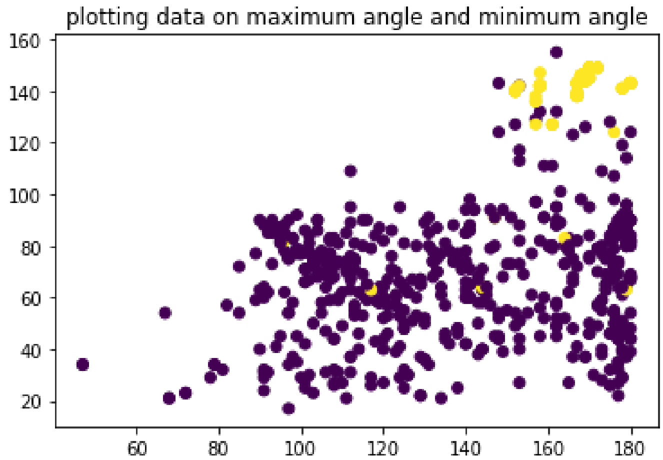


Fig 3.16: The connection between angles which are done manually.

angle in the roundabout. If its roundabout, maximum angle is in range 150 to 170. Similarly, minimum angle ranges from 60 to 150. If it is not the roundabout, there are angles which are 180 degree, which is not the roundabout having the discontinuous direction of travel.

Fig 3.17 shows the relation between the minimum angle and the total number of links with reference to the roundabout. If the loop is not roundabout it may have more than 22 links and the minimum angle is mostly in the range of above 120, whereas if it's roundabout there are some links less than 10 in size and minimum angle is from 40 to 140 degrees.

Fig 3.18 shows the analysis of the minimum angle in the complete dataset. The aggregate data can be represented into two sets between 20 to 120 and from 120 to 150 degrees. As the data was massive, to present complete information, the used bin is 100.

Fig 3.19 shows the analysis of maximum angle in the complete dataset. It's been observed that the maximum angle starts from 90 to 180 degrees for a closed loop. As the data was massive,

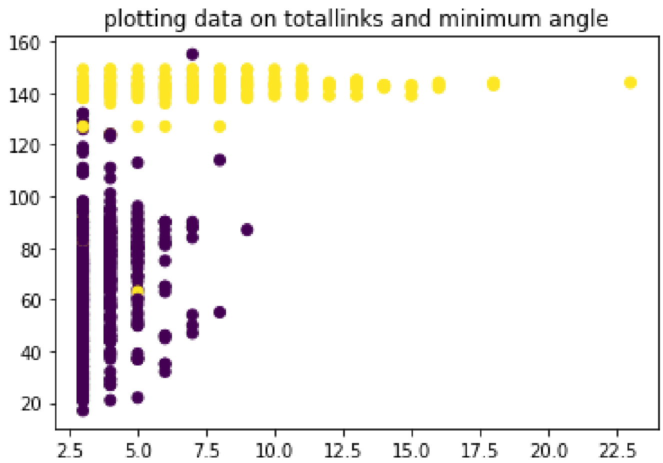


Fig 3.17: The relation between the total links and the minimum angle as a collection of links forming the roundabout

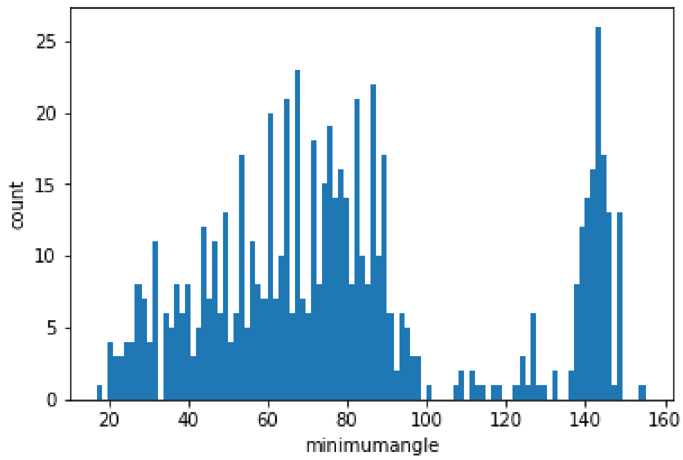


Fig 3.18: The relation between the minimum angle and the links

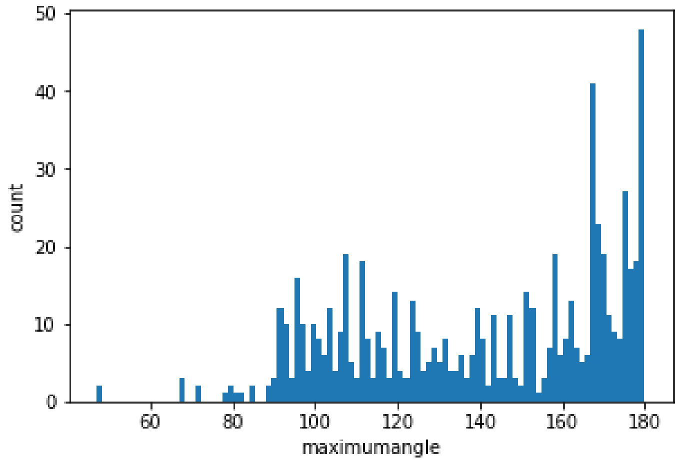


Fig 3.19: The relation between the maximum angle and the links

to present complete information, the used bin is 100.

The proposed approach is very practical and uses real world data. A deterministic algorithm has been developed to solve the majority of the cases and is complemented by a machine learning model to solve the rest in a more or less traditional way.

The primary concern was to have some deterministic solution to this problem which can be used in a real production environment, Since a majority of human workload was easily reduced by an algorithmic deterministic approach it was felt that using machine learning on the complete data was an overkill and hampered the accuracy of the models.

Initially, with the deterministic techniques, more than 80% of non-roundabouts from the closed-loop got eliminated. The feature extraction technique of the machine learning approach is achieved by analyzing the complete attributes, which enhanced the results drastically. In some regions, more than 98% of valid cases got detected. The idea of detecting roundabout depends upon the collection of links associated inside it.

The basic filtering mechanism on the direction of travel which is the native property of the roundabout was done by using this deterministic technique; that means a roundabout can be either clockwise or anticlockwise. The implementation of the algorithm helps to check these conditions by analyzing every link and forming the continuous direction of travel. Also, by implementing machine learning algorithms, the remaining fuzzy cases which needed some probabilistic approach to minimize the deviation got solved. With domain knowledge, the attributes got filtered out, which define the roundabout in the 2D closed-loop geometry.

Finally, by using multiple machine model, the performance was enhanced to more than 98%. As explained, the problem of identifying roundabouts was approached by using separate deterministic and non-deterministic models. The deterministic approach was able to detect more than 80% of the non-roundabouts correctly across all regions. This reduced the data set due to which it became possible to process the probable-roundabouts with approximately 98% accuracy. This technique has been implemented in HERE maps for various countries and the following results have been obtained.

Table 3.5: Results obtained by using the proposed approach for maps in Eastern Europe

Eastern Europe	**Before**	**After**
Number of exceptions	31487	∼6200
Man-hours to correct it	∼3100	∼600
Automation	0%	∼82%

Table 3.5 results show the analysis of the deterministic approach for eastern Europe. Initially, there has been 31487 closed loops which are potential candidates for the roundabout but after applying the deterministic algorithm, it has reduced to only 6200 exceptions, which lessen the working hour effort from 3100 hrs to 600 hrs. Hence, we were able to get the automation of

82%.

Table 3.6: Results obtained by using the proposed approach for maps in North America

North America	**Before**	**After**
Number of exceptions	55907	4473
Man-hours to correct it	∼5590	∼450
Automation	0%	∼94%

Table 3.6 results show the analysis of the deterministic approach for north America. Initially, there has been 55907 closed loops which are potential candidates for the roundabout but after applying the deterministic algorithm, it has reduced to only 4473 exceptions, which lessen the working hour effort from 5590 hrs to 450 hrs. Hence, we were able to get the automation of 94%.

Table 3.7: Results obtained by using the proposed approach for maps in Western Europe

Western Europe	**Before**	**After**
Number of exceptions	65591	5183
Man-hours to correct it	∼6560	∼520
Automation	0%	∼92.1%

Table 3.7 results show the analysis of the deterministic approach for western Europe. Initially, there has been 65591 closed loops which are potential candidates for the roundabout but after applying the deterministic algorithm, it has reduced to only 5183 exceptions, which lessen the working hour effort from 6560 hrs to 520 hrs. Hence, we were able to get the automation of 92.1%.

Table 3.8: Results obtained by using the proposed approach for maps in Taiwan

Taiwan	**Before**	**After**
Number of exceptions	886	45
Man-hours to correct it	∼80	∼4
Automation	0%	∼94.9%

Table.3.8 results show the analysis of the deterministic approach for Taiwan. Initially, there has been 886 closed loops which are potential candidates for the roundabout but after applying the deterministic algorithm, it has reduced to only 45 exceptions, which lessen the working hour effort from 80 hrs to 4 hrs. Hence, we were able to get the automation of 94.9%.

In each of the above cases, the remaining automation i.e. ∼18% in the case of Eastern Europe, ∼6% in the case of North America, ∼8% in the case of Western Europe and ∼5% in the case of Taiwan was done using the proposed machine learning models which was able to give more than 98% accuracy thereby significantly reducing the manual work needed.

These are real results which were calculated on the basis of application of the proposed methodology.

3.5 Conclusion

In conclusion, through this study, it has been demonstrated that just by using the domain knowledge and surface geometry it is possible to correctly solve the problem of identifying roundabouts. The proposed approach involved splitting up the practical scenarios into two - a deterministic one and a non-deterministic one. Using these approaches the required man-hours were significantly reduced and that resulted in improved automation in more than 98% of the cases where the technique was implemented. This methodology has a lot of scope and can be utilized to further automate the identification of roundabouts for maps in multiple countries.

Chapter 4

Detecting Junctions in Roundabout using map data

4.1 Introduction

Map features are the building blocks in making maps. Features are further divided into attributes; accuracy in attribute prediction is directly proportional to the quality of the feature. Our targeted feature is roundabout, and the problem is junctions identification. Data used in the research are real-time data from Here Maps. Prediction of the accuracy of junction in roundabout improves the overall accuracy of prediction of roundabouts. An attribute is interchanged as qualities or sub-features. Minimal research has been done using actual data from maps and field knowledge.

Multiple roads connect via junctions. Fig 4.1 shows the mathematical representation of a closed-loop in a map. The road is a collection of lines; the connection of the roads r_1, r_2, r_3 and r_4 is via junction j_1, j_2, j_3 and j_4. The map feature is not roundabout in the figure, it's the closed-loop of roads.

Fig 4.2 shows the roundabout having junctions. r_1, r_2, r_3 represent the road and j_1, j_2, j_3 represent junctions. Chapter 3 describes an approach to identifying the correct intersections which can be potentially part of the roundabout. Further, roads are collections of one or multiple lines.

In short, a roundabout controls traffic from incoming road as it from closed loop represented in Fig 4.3, which is a practical geometric representation. Roundabouts have some qualities for e.g. traffic in roundabouts cannot be in both directions. Practically, a roundabout is a collection of different links which geometrically can be just straight line. By using business or domain logic, a line can be converted into a link using these qualities.

Roads are a geometrical collection of lines. More the lines more are the smoothness of the road. The blue curve in Fig 4.4 is a collection of distinct lines and the common nodes are connecting multiple links. Here, it's just the two lines. Now, to form links, the lines should have some properties which are domain specific, such as direction of travel, reference node id (node1) and non-reference node id (node2). Here, node1 is a point which is towards the south, so this nomenclature has been used to define the node through which it is possible to identify

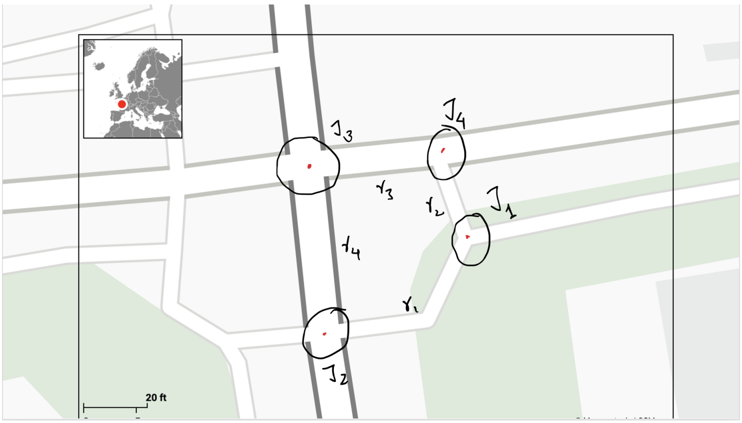


Fig 4.1: 2D representation of Junctions in Closed Loop

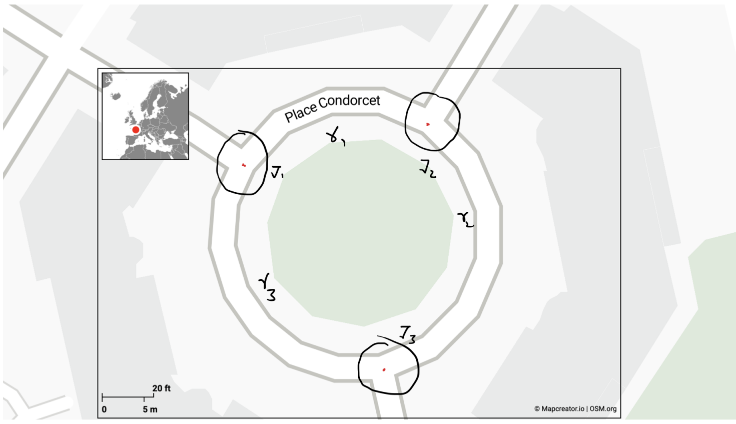


Fig 4.2: 2D representation of Junctions in roundabout

the direction of travel of the links. Similarly, node2 is a node which is towards the north. These specific rules define the direction of travel (DOT). If the traffic is in both directions, then DOT is 'both'. If traffic is from node1, the DOT is 'from'; finally, if the traffic is to node2, DOT is 'to'.

Fig 4.5 shows the maps are just a collection of pure lines and the junctions at which the two lines meet are the nodes. A line is the logical representation of a link in the Map domain. In Fig 4.5, four streets form a closed-loop, which is an acyclic graph. Chapter 3 first identifies the cyclic graph in the maps and then uses the domain knowledge to filter the data. After that, the machine learning models are applied to solve this problem.

Fig 4.6 represents the geometrical and practical view of acyclic graph, which is not a roundabout. It shows the 2D representation of a practical map. It is observed that any complex intersections on the road can be represented using simple geometry. A roundabout is nothing but a closed loop, a collection of links. Majority of work is globally done on image processing, probe data, sensor and traffic data analysis for the identification of roundabouts and junctions [11, 12, 13, 1, 14, 15, 16]. Very less work is available on the useful raw map data with the map domain information. Real map data is used to solve the problem. Sub-sections are divided into the

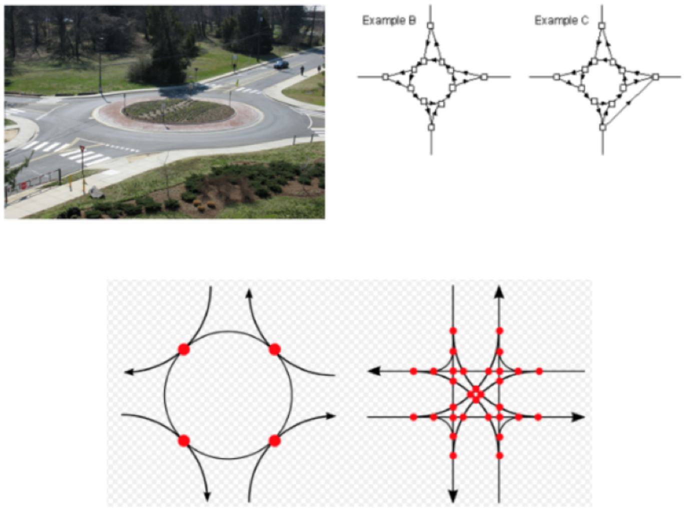


Fig 4.3: Roundabouts representation with one of permuted traffic flows

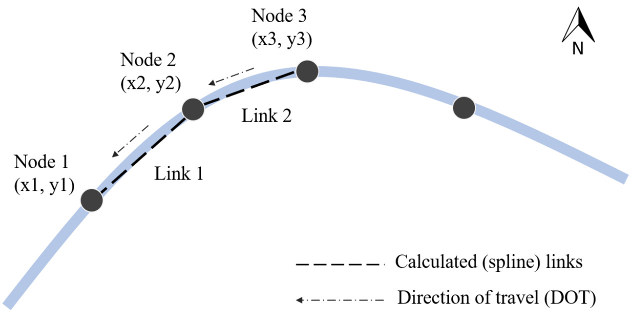


Fig 4.4: Direction of travel in two nodes based on data

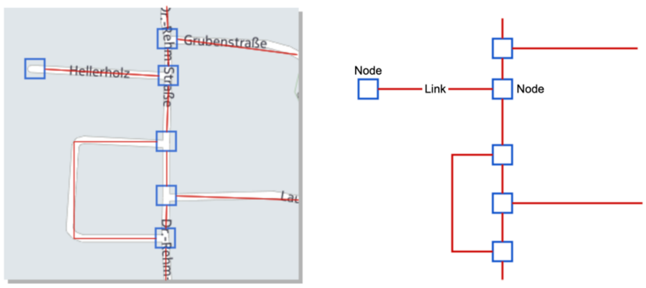


Fig 4.5: Represent the data into a geometrical map

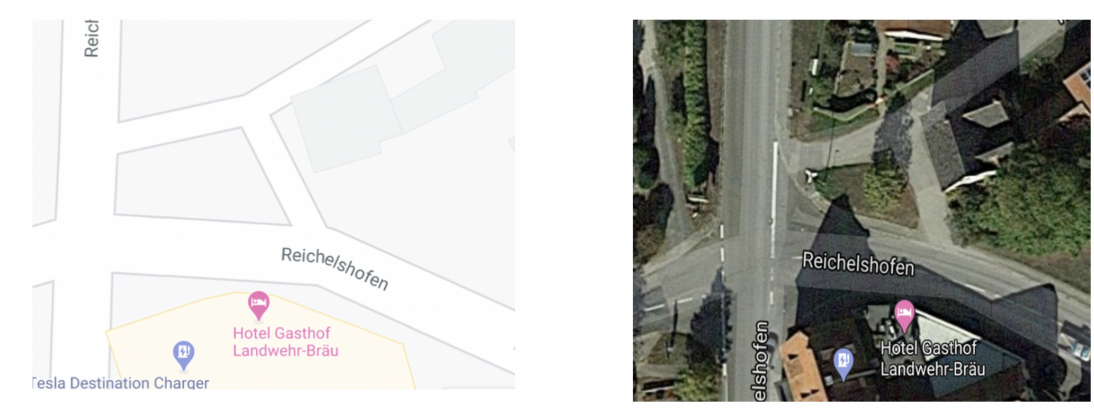


Fig 4.6: Geometrical and Real representation of closed loop which is not a roundabout

literature review, methodology, results, and conclusion.

4.2 Proposed Methodology

The Map domain has many real-time problems, such as the problem of the address range, point address, spline estimation, etc. The Map features considered is roundabout. On analyzing, it was possible to divide it into many sub-problems. One of the exciting parts was the analysis of nodes in a roundabout, which brings up the problem of junctions.

A roundabout is a closed loop, and there can be many exits and entry nodes which can act as junctions. Data in maps are generated via manual and automated processes; its quality is assured using geocoders. There are many input sources for data generation like lidar, images, manual feed, sensors etc. With access to the enormous real-time data, a solution has been proposed which can be used to predict junctions in a closed loop of roundabouts.

There are many approaches defined in past literature, but none of them used the actual practical map data. Therefore, the proposed solution uses real data using the map domain.

With the help of real-time map data, the complete process was divided into the following:

- Data Extractions
- Data Analysis
- Model Implementations and Improvements

For the process of data extraction, firstly the closed loops are required to be found in 2D maps. Once the data is extracted, the valid and invalid cases are identified. The valid cases are the ones which are roundabouts and the invalid cases are the ones which are not roundabouts.

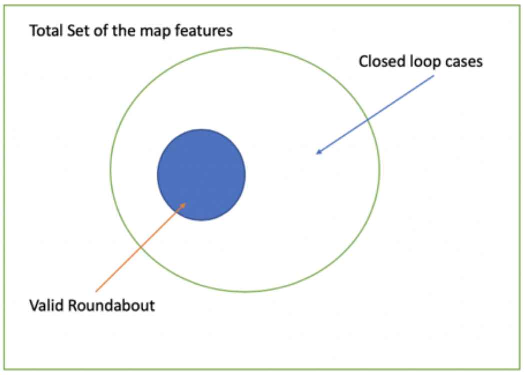


Fig 4.7: Venn Diagram for the different Cases

Fig 4.7 shows the Venn representation of the cases. All roundabouts will be closed loops, but not all closed loops need to be roundabouts.

From complete universal sets of the closed-loop, the valid cases of roundabouts are visually represented. Further, domain analysis has been performed and five attributes have been identified, which play a vital role in determining roundabout junctions. Below is the complete description.

4.2.1 Direction of Travel (DOT)

This is an important attribute which is considered in making decisions as the exit or entry nodes in a roundabout can have any of the three directions. Regarding the junction links in roundabouts, it was found that the majority of them have the direction 'BOTH'. Junction links which connect the roundabout at a junction have traffic flow in both directions.

For the junction link cases where the closed-loop is the roundabout, from the Fig 4.8 the direction of travel (DOT) is almost equally distributed with the maximum tending towards the case with traffic flowing in both directions where 1 represents DOT as 'both', 2 represent 'to' and 3 represent 'from'.

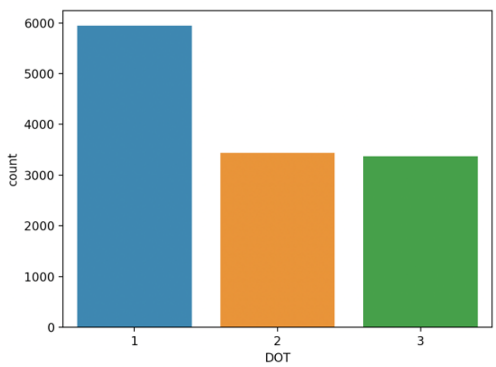


Fig 4.8: DOT for the dataset with the valid roundabout cases

In cases where the junction link is a part of a closed loop but is not the roundabout. From Fig 4.9, it can be concluded that majority of junctions data have direction of travel as 'BOTH'.

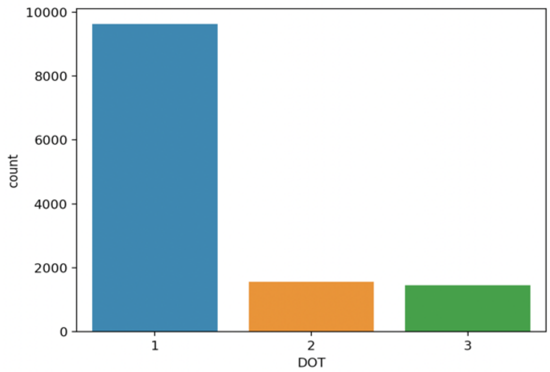


Fig 4.9: DOT for the dataset with the invalid roundabout cases

4.2.2 Minimum angle

The minimum angle connecting two links at the junctions of roundabouts is calculated. It is observed that the minimum angle is an essential attribute in deciding, but it is pretty fuzzy in nature. The range is usually between 50 to 125 degrees.

4.2.3 Maximum angle

The junction link associated at the common node has been used to calculate the maximum angle between the links of the roundabout. The value is usually more than 150 degrees and is quite fuzzy in nature.

4.2.4 Node Size

Valency is degree of the node, which means how many unique roads are connected on junctions. It is also an important feature and is fuzzy in nature. Fig 4.10 shows the distribution of the valency of the nodes in the roundabout. It is observed that majority of them have the valency of three.Lets degree of the node be N_i and the roads be L_i, then to be part of roundabouts, the summation of the degree of the node should be more than the lines connecting them.

$$N_0 + N_1 + N_2 + \cdots + N_i > l_0 + l_1 + l_2 + \cdots + l_i \tag{4.1}$$

4.2.5 Roundabout

This is the value which tells that for each junction link, whether the link is part of a roundabout or not, its value is binary i.e., either true or false. These features of the junction links were

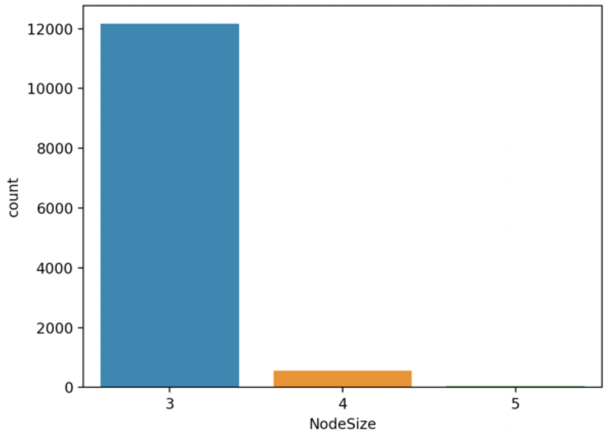


Fig 4.10: Valency of Nodes in the roundabout

used and different models were implemented on it which gave the accuracy of 69%, with the best model being logistic regression.

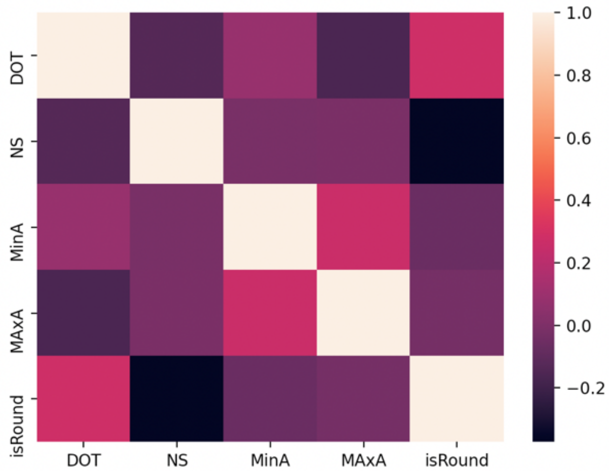


Fig 4.11: Heat Map of all the Atributes

Fig 4.11 represents the heat map of features after analysis of data. X and Y-axis are the relations between DOT (Direction of travel), NS (node size), MinA (minimum angle connecting the links at the node), maxA (maximum angle between them) and finally IsRound (checks whether the link is valid part of roundabout). DOT has maximum impact on every feature and link that is part of roundabouts. Finally, DOT is used as the grouping mechanism for the junction data to improve the accuracy by 81%.

In Fig 4.12, the complete dataset was divided into three different sets. Set one is the dataset with the direction of travel as 'BOTH', set two is the dataset with the direction of travel as 'TO' and finally set three is the dataset with the direction of travel as 'FROM'. The idea is that the junction link will have one of the three directions of travel and hence any one model

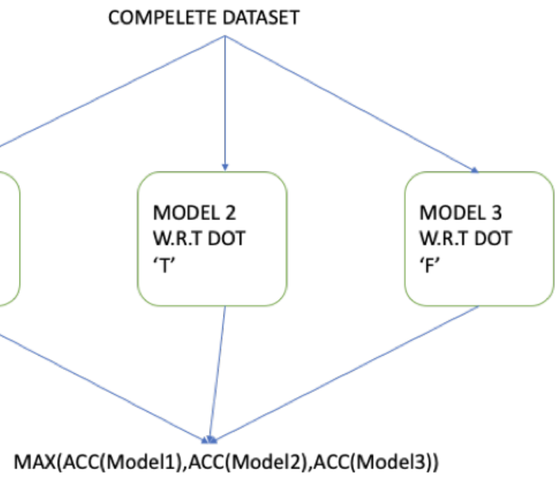


Fig 4.12: New Approach w.r.t DOT

will provide the best results.

Data Extraction

Data extraction was the main challenge for analyzing this problem. Working in the map domain, we were fortunate enough to have access to the map data. So the data has been extracted using two steps.

1. Getting data for valid junctions which is part of the roundabout.
2. Getting data for invalid junctions which is not the part of roundabout.

As the roundabout is a closed loop, firstly the cyclic loops as extracted which are the potential candidate for the roundabout. Thereafter, there are two cases: the loops which actually are roundabouts and those that are not.

$$\sum_{i=1}^{x} f(i) \tag{4.2}$$

From Eq 4.2, f(i) are function of different links which finally forms the closed loop.
The extraction steps is described as follows:

Loops which are not roundabouts (Case 1):

1. All the roads/links associated to the closed loop are captured.
2. Calculate the valency of the nodes.
   If (valency>2)
      Check the link which is not the part of the roundabout and store it
   Else
      Continue;

3. Get the stored link from step 2, calculate the angle it makes with links which are part of the cyclic graph, store the maximum and minimum value.
4. Get the total size of the node i.e. number of links connected to it.
5. Get the domain information, which is the Direction of travel (DOT) on that link.

Loops which are roundabouts (Case 2):

Links from the loops which are roundabouts were taken and the same steps were applied as defined above for the data extraction.
Finally, the direction of travel (DOT) atributes are mapped as defined below -
If DOT='B', then it is 1
If DOT='T', then it is 2
If DOT='F', then it is 3

Table 4.1, represents generated sample Dataset used for prediction, junction link which is leading to roundabout is predicted. Data set is uploaded at [74]. The authors created this dataset with real-time Map data.It does not have any benchmark. Data were analysed for north america with 128886 junctions, and features were calculated. DOT is the direction of travel and provides the traffic flow over the road, and node size is the valency of the connecting junction in the closed loop. Max Angle is the angle made by neighbouring links at the common intersection.

Similarly, Min Angle is the minimum angle made by the adjacent arms, and, finally, the boolean value indicates that the specific node, intersection or junction is part of the roundabout. Most junction parts of roundabouts have valency 3; many closed loops can have more than four valencies. It is observed that most roundabouts have an obtuse angle with few exceptions. This Automated script was used to extract the information from Europe and America.

Table 4.1: Data set sample representation

DOT	NodeSize	MaxAngle	MinAngle	IsRoundabout
1	3	178	84	0
1	3	166	134	0
2	3	159	85	1
2	3	161	57	1
1	3	108	81	1

4.3 Analysis Of Data

Data of 181533 junction links has been extracted. On removing the duplications, 25399 distinct link entries were identified. The aim of the research was to identify valid junctions in roundabouts. From the junction data set, this is a binary classification problem. The complete data has been divided into two cases based on whether the junction is part of the roundabout.

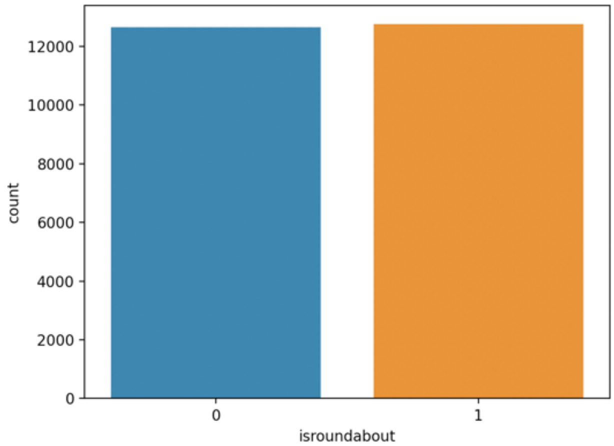


Fig 4.13: Classification of complete data w.r.t junctions in the roundabout

Fig 4.13 represents the total sample set into two groups viz. junctions which are part of roundabouts and those which are not. It is similar for both the cases. This classification was done after removing duplicates from the data.

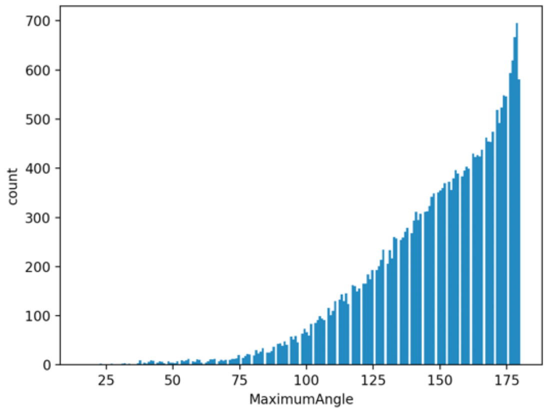


Fig 4.14: Shows the analysis of maximum angle

As the data is quite large it has been divided into different sized bins. From Fig 4.14, it can be seen that majority of the junction links make an obtuse angle usually measuring more than 150 degrees with the roundabout.

Similarly from Fig 4.15, a bin of 200 has been used to represent the data. It was found that majority of junction links make an angle between 50 to 125 degrees with the roundabout.

Fig 4.16 shows the complete analysis of the node valency of the junctions link with the roundabouts and it was found that the majority of them have the valency of 3 and 4.

Fig 4.17 represents the analysis of the direction of travel over the complete data of the junctions. As the majority of them belong to the links having the direction of travel as 'both', where as

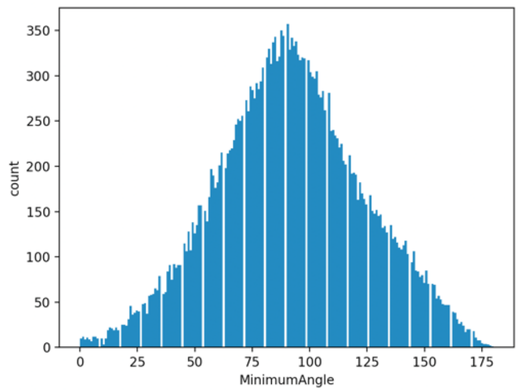


Fig 4.15: Shows the analysis of minimum angle

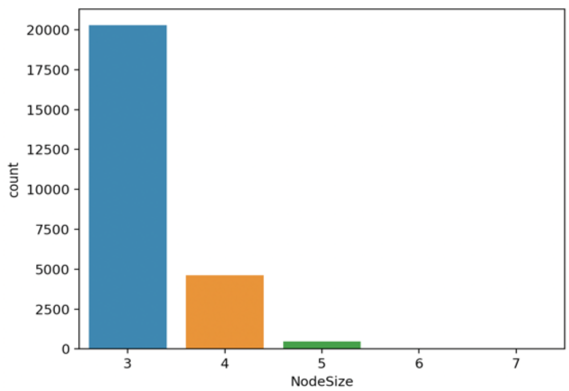


Fig 4.16: Valency of the nodes in the cyclic graph

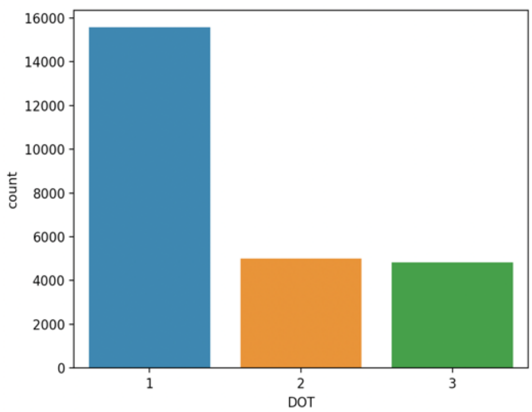


Fig 4.17: Analysis of data w.r.t DOT

the other two cases have almost similar values, it can be concluded that DOT can be used to group the data.

To further emphasize the idea the DOT with the maximum angle was analysed in Fig 4.18. It is interesting to see that for the DOT value of 'both' there are more invalid cases.

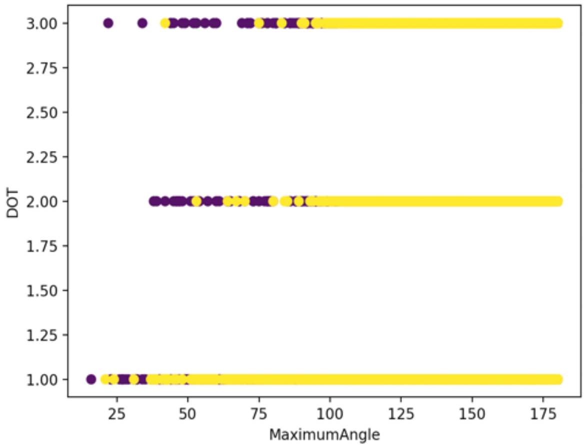


Fig 4.18: Shows the relation between the maximum angle and direction of travel w.r.t roundabout

DOT was used to group the data. In this new approach DOT was used as the main attribute to divide the junction data into three groups. Variety of different machine learning models were tried to get the results. The model was trained separately for the three groups which finally enhances the performance.

Given below is a description of the steps for the new approach

- Grouping the data into three group as per DOT
- Running all the models on this three group
- Maximum (G1, G2, G3) is the prediction

4.4 Results

This section is divided into two parts: Implementation with the raw data and then the improvisation with respect to DOT. First, the results are discussed with the raw data through which it was possible to get the accuracy of 69% and with the best implemented model as logistic regression. Thereafter, improvisation with respect to DOT will be discussed which improves the performance of the two groups to the accuracy of 81%.

Table 4.2 shows the comparative results of all the models used on the raw data. The last entry is the result of the improvised approach which has increased the efficiency. Fig 4.19 shows the ROC curve of different models. It can be concluded that logistic regression was the most efficient which has the accuracy of 69% which was further validated from the five-fold validation run in Fig 4.20.

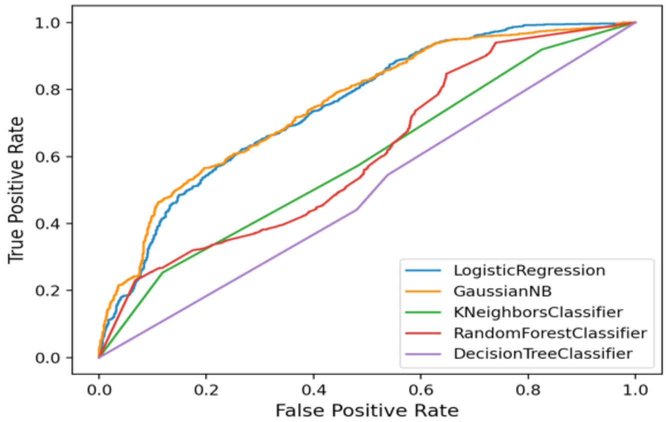


Fig 4.19: ROC curve for the implemented models

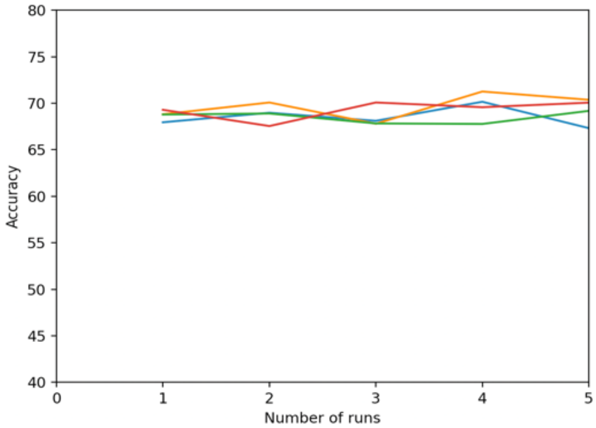


Fig 4.20: 5 Folds cross validation for logistic regression

4.4.1 Improvised approach

As described in the above sections, the complete data analysis indicates the direct impact of the DOT of the link on the junction node of the roundabout. Hence, data was divided into three groups as represented in the Fig 4.12. Different models like Decision tree random forest, K neighbors, Gaussian Nb and Logistic Regression were used on the data. An accuracy of 64%

Table 4.2: Comparison of different models on the raw data

Model Name	Gini	ACC	AUC	Spec	Sens
Decision Tree	-0.02	0.48	0.49	0.44	0.52
Random Forest	0.21	0.52	0.60	0.51	0.54
KNeighbors	0.18	0.55	0.59	0.57	0.52
GaussianNB	0.51	0.66	0.76	0.93	0.38
Logistic Regression	0.5	0.69	0.75	0.71	0.62
Improvement Group 2+3	0.55	0.81	0.77	0.98	0.42

Table 4.3: Analysis of different models for the Group 1

Model Name	Gini	ACC	AUC	Spec	Sens
DecisionTree	-0.10	0.48	0.45	0.28	0.60
Random Forest	0.05	0.51	052	0.33	0.62
KNeighbors	0.04	0.54	0.52	0.37	0.64
GaussianNB	0.41	0.61	0.71	0.86	0.45
Logistic Regression	0.39	0.64	0.70	0.44	0.76

for group with DOT as 'BOTH' and 81% for other two groups with DOT as 'TO' and 'FROM' was achieved. We describe the complete analysis and comparisons of different models in the tables and graphs below.

Analysis of the dataset having direction of travel as 'BOTH' - Group 1

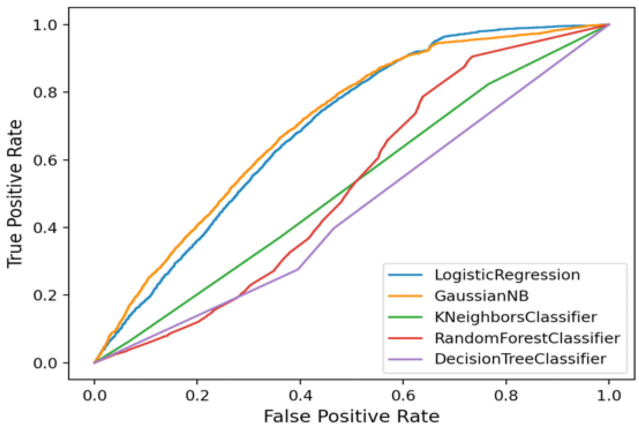


Fig 4.21: ROC curve for the Implemented Models for the group 1 of the data

Table 4.3, shows the results of the junction related data on five different models where the logistic regression has provided the best results. Accuracy of the same was 64%. With different test ratios the accuracy reached at 68%. But after five fold validation it was concluded to be 64%. In Fig 4.21, different models ROC have been compared where it can be observed that naive bayes and logistic are at par, compared to other models.

Analysis of the Dataset having the Direction of Travel as TO - Group 2

Table 4.4 shows the results of the junction related data on five different models. Gaussian naive bayes and logistic regression have provided the best results. The accuracy of the same was 80% and with different test ratios the accuracy reached to 83%. However, after five-fold validation it was concluded to be 80%. In Fig 4.22, ROC of different models have been compared where it can be clearly seen that naive bayes and logistic are at par, compared to other models.

Analysis of the Dataset having the direction of travel as FROM - Group 3

Table 4.4: Analysis of the different models for Group 2

Model Name	Gini	ACC	AUC	Spec	Sens
DecisionTree	0.39	0.71	0.69	0.77	0.60
Random Forest	0.56	0.77	0.78	0.83	0.62
KNeighbors	0.41	0.74	0.71	0.86	0.48
GaussianNB	0.64	0.80	0.82	0.98	0.42
Logistic Regression	0.61	0.80	0.81	0.98	0.42

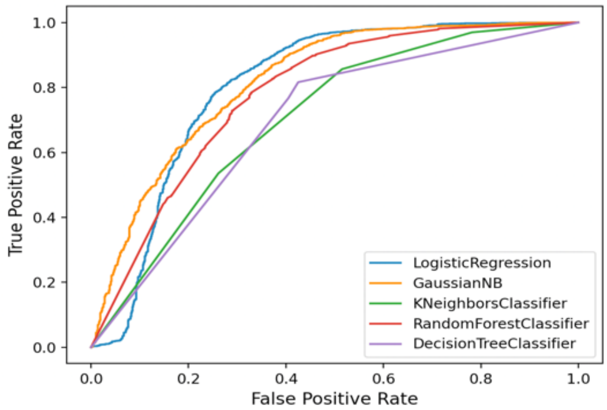


Fig 4.22: ROC representation for the group 2 dataset

Table 4.5 shows the results of the junction related data on five different models. Gaussian naive Bayes and logistic regression have provided the best results. The accuracy of the same was 80% and 81% respectively, which were run with different test ratios. Sometimes it reached 83%, but after five-fold validation the average results were concluded. In Fig 4.23, ROC of different models have been compared where it can be seen that naive bayes and logistic are at par compared to other models. Fig 4.24 shows the five-fold cross validation results for the logistic regression and it can be confirmed through this that the accuracy is 81%.

A total of four approaches have been implemented. In the first variation, the raw data is used to get an accuracy of 69%. Thereafter, using the domain knowledge, the other three approaches were used which improved the results to 81% accuracy, especially for the two sets of data.

Table 4.5: Analysis of different models for the Group 3 data

Model Name	Gini	ACC	AUC	Spec	Sens
DecisionTree	0.36	0.71	0.68	0.77	0.57
Random Forest	0.53	0.75	0.77	0.83	0.58
KNeighbors	0.42	0.74	0.71	0.87	0.45
GaussianNB	0.63	0.80	0.81	0.98	0.39
Logistic Regression	0.55	0.81	0.77	0.98	0.42

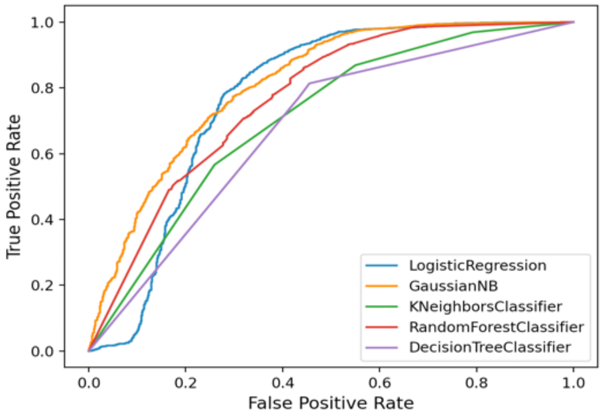


Fig 4.23: ROC representation for the Group 3

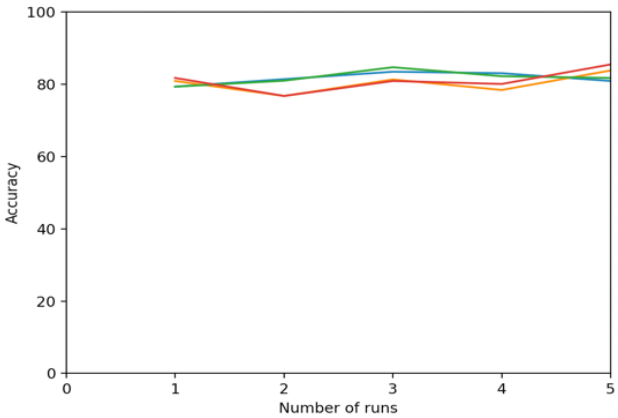


Fig 4.24: The 5 folds cross validation for Group 3 for logistic regression

4.5 Conclusion and Future Scope

Working in the map domain, it was a privilege to access raw map data. Domain knowledge has been used to perform a complete analysis of the junction links in roundabouts and provided a new methodology for their identification. We have extracted the required data as per the problem statement, improvised the approach and implemented different models to achieve an accuracy of 81%.

In the future, some other attributes, like the speed limit on junction links, can be considered, which may enhance the model's performance in identifying the junctions in roundabouts. We may add sensors and other lidar data to conclude the results with more accuracy.

Chapter 5

Identify Kink In 2D Map Using The Spline Technique on Real Map Data

5.1 Introduction

There are several proposed techniques to solve the smoothness of graphs but not much has been done with real Map data for identification of kink after the 2D geometry is created. This approach is the proof of applied implementation of the proposed technique for identification of same. The Spline Technique has been used to interpolate the points from the geometry.

Fig 5.1: Representation of a real road

Fig 5.2 represents the detailed technique for the creation of lines from road. The approximation is done with real road by creating a linear 2D geometry for the Maps. Blue curve represents 4 control points/shape points. Fig 5.1 represents real roads. Real representation of road is basically the collection of lines example.

Fig 5.3 represents road with 2-D representation of collection of lines. In Fig 5.4, just the graphical representation of real points of the lines which are real roads are presented. The figure clearly shows the kink in curve.

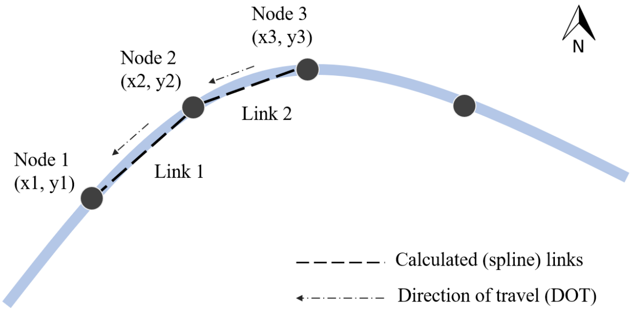


Fig 5.2: Representation of 2D representation

Fig 5.3: Represent the data into a geometrical map

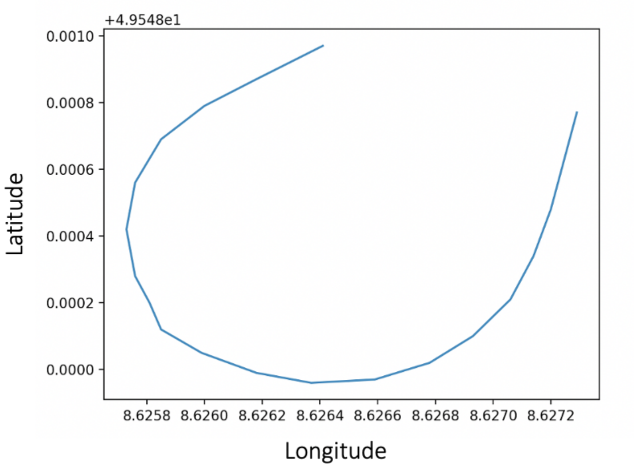


Fig 5.4: Geometrical representation of road in 2D

5.2 Proposed Methodology

The proposed methodology is using spline technique to form a new best fitted curve using available geo-geometry. The forced control points are extracted from the curve which should match with the original geometry. Lines are created from the new control point and the variation is calculated. If the variation or the perpendicular distance is greater than 10 cm, then there is possible kink on the geometrical representation of the features which could road, topology or any complex combination of the roads like Roundabouts. In mathematical terminology splines are the polynomials which when connected makes the curve smooth. The used technique is cubic spline and for the calculation of the perpendicular distance from the point to the interpolated lines are used form the book [75].

$$APdotAB = |AP||AB|cos(PAQ)(Fig5.5) \tag{5.1}$$

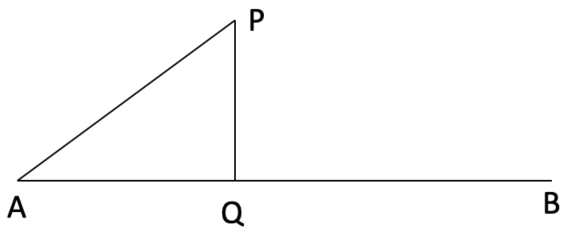


Fig 5.5: Perpendicular line

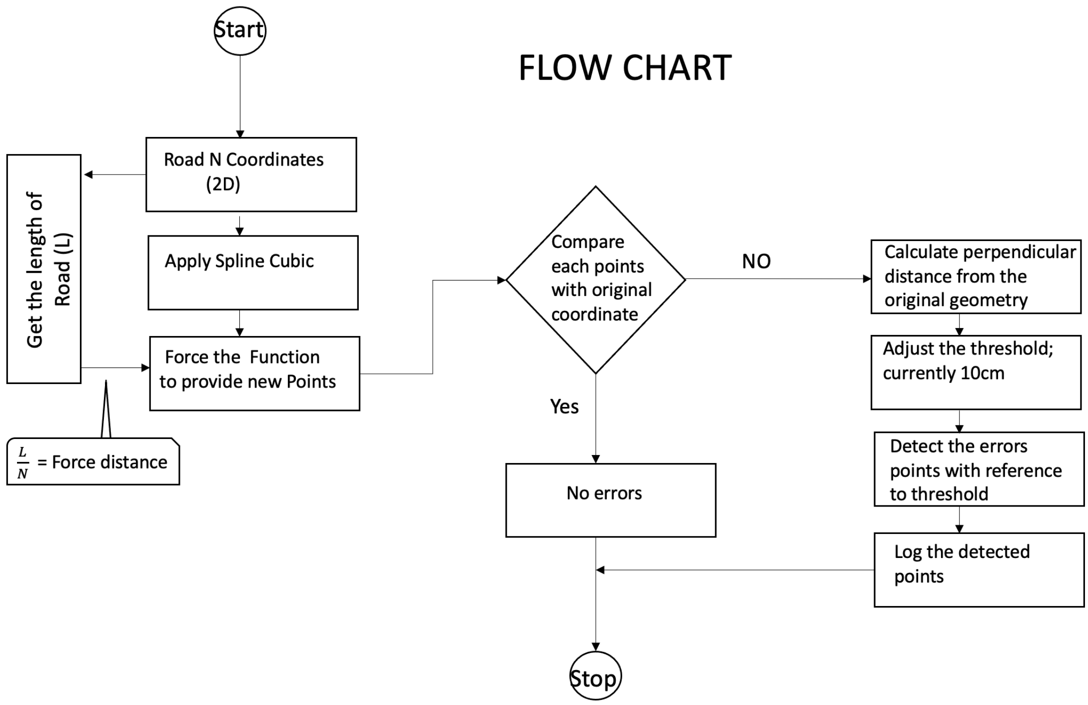


Fig 5.6: Flow Diagram of the Proposed Algorithm

Where AB is the line segment and PQ is the perpendicular distance.

5.2.1 Flow Diagram

Fig 5.6 represent the Flow Diagram of Algorithm, The data is Geospatial data and have road representation in the form of geo-geometry. With the use of Spline technique the smoothing is done and new set of points are extracted which is further compared to get the deviation of the original points.

5.2.1.1 Algorithm

- Get the 2D geo-geometry of the road.
- Curve Fitting Technique (cubic spline):

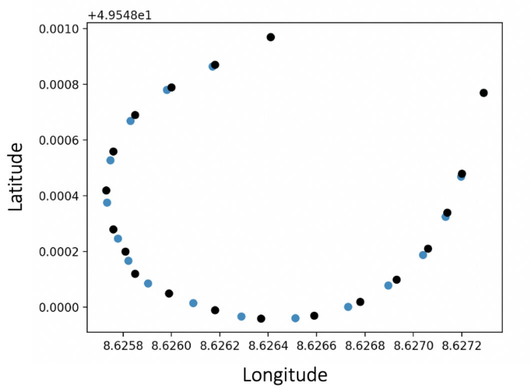


Fig 5.7: Points representation of original and interpolated lines

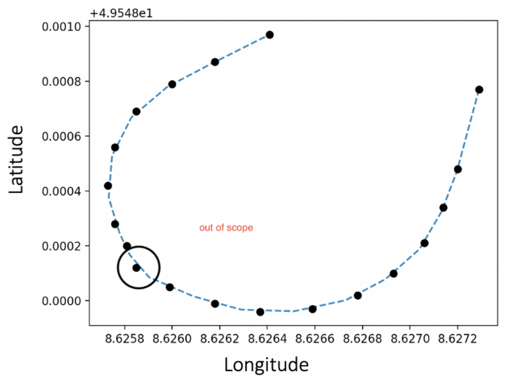


Fig 5.8: Line and points representation

The best optimize representation of spline with the given geometry is interpolated with cubic spline technique. The interpolated spline from the real geometry is forced to give the control points, with reference to the size of real geometry points by using the distance between the first and the last point of the curve, so that the best fitted curve can be made using the interpolated spline providing the exact number of control points with reference to the real geometry coordinates.

- Compare each approximated spline lines with the original coordinates.
- Calculate the perpendicular distance between the point and the interpolated spline line (this will help in identifying the variation from the interpolated line).
- If the distance is greater than $1.0 * 10^-5(10cm)$, than that point needs to be corrected.

$$sum(c(x,y)) = \sum_{i=1}^{n-1} L(c(x_i, y_i), c(x_{i+1}, y_{i+1})) \tag{5.2}$$

where c(x,y) is coordinates, L is the length and n is the control points that is the coordinates.

The used interpolation is basically the cubic polynomial equation which is represented by

$f(x) = ax^3 + bx^2 + cx + d$

$fd = L/n$

where fd is forced distance, L is the Length and n is the control points.

The error rate was tuned from 10 cm, 20 cm, 30 cm, 40 cm, 50 cm. As the error is reduced the noise increases, hence after reducing to less than 10 cm the noise was very huge. It was decided after multiple iterations that the best threshold is 10 cm. The complete Algorithm was tested on 2 regions Frankfurt and USA.

Fig 5.9: 2D representation of both original and interpolated lines

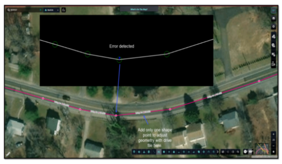


Fig 5.10: Detection of the Kink

Fig 5.7, shows the interpolated control points with the real points. Fig 5.8, shows the variation of the point from the original geometry. Fig 5.9 , shows the linear representation of both the geometries. Orange is the original one and black is the interpolated.Above figures shows the different representation of the road in the real Maps and the presentation of the proposed methodology to identify the kink.

Fig 5.9 and Fig 5.11 show the variation of the points for 2 different roads. For the r1 (road1) the circled points are above the thresholds and possible candidates of kink. Similarly for road2 (r2) there is one possible point which is having the kink.

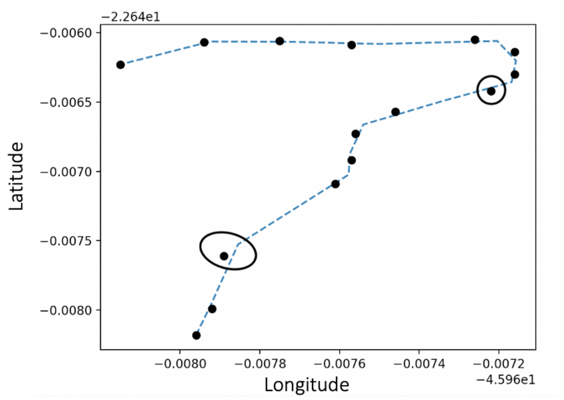


Fig 5.11: Kink in the road 1

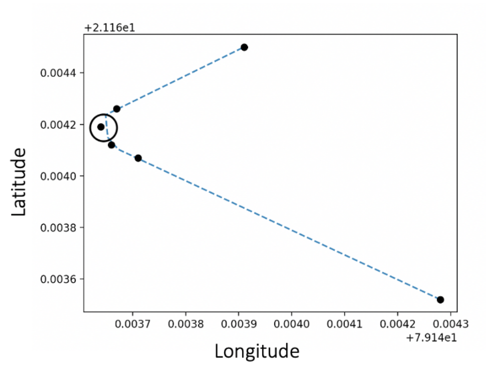


Fig 5.12: Kink in the road 2

Table 5.1: Sample results with the kink error rate

RoadID	Lat	Lon	error_rate
1	42.49075	-83.33225	1.70683107154
1	42.4919	-83.33607	0.18647938531
1	42.49276	-83.3388	0.10281089654
2	42.49075	-83.3232	0.13779231970
2	42.49071	-83.32604	0.11606565833
2	42.49067	-83.32872	0.18234617811
2	42.49067	-83.32948	0.44892143655
2	42.49069	-83.33023	0.21069185261
2	42.49074	-83.33098	1.70671522441

Table 5.2: Frankfurt and USA region

Considered Error	%detect(frank)	%detect(USA)
1 m	60	63
50 cm	76	69
40 cm	78	74
30 cm	81	75
20 cm	86	80
10 cm	91	83

5.3 Results and Discussion

Fig 5.10, shows detection of kink on the road using the proposed algorithm with threshold of 10cm. Table 5.1, shows result of roads having kink in them, roadId is the unique name for collection of geometrical line, the points which makes complete road is detected which have values more than 10 cm error rate. Lat, Lon are the latitude and longitude of the respective points.

In Table 5.2, result of the complete Frankfurt is presented on column 1, the selected bounding box is 50.33647 (north), 50.00887 (south), 8.83696 (east), 8.33939 (west), with 10cm error threshold the detections where around 91%, for 1 meter the detection where 60%. Result of the complete USA is shown in column 2, the selected bounding box is 49.5904 (north), 24.9493 (south), -66.9326 (east), -125.0011 (west), with 10cm error threshold the detections where around 83%, for 1 meter the detection where 63%. when the error threshold reduced to less than 10cm the noise increased, the optimal threshold after multiple iteration was found to be 10cm. Overall the identification of kink depends on the threshold value and the more the threshold is relaxed, the identification increased and hence the noise. Used linux machine (Intel(R) Xeon(R) CPU E5-2650 v4 @ 2.20GHz), Java 11 Application, JTS library for the Geometries manipulations. Database used is Oracle.

5.4 Conclusion

With help of real Map data, kink in the created road is identified.The proposed technique is efficient and is easy to implement. Result shows that error threshold is directly proportional to detection and so do noise. For frankfurt, USA detection was around 91% and 83%. The proposed methodology was tested on two region and results are shared.

In future, correction can be added in this approach by selecting neighbouring points of the detected one. with reference to these three point s, spline can be made and hence corrected new points can be added. Further GPS traces and Lidar data can be used to remove the kink.

Chapter 6

Approximating the Structure of a Curved Road using Polylines for Maps and Related Applications

6.1 Introduction

When approximating a curve with the help of polylines, there's always a challenge of what would be the optimal number of lines that can correctly describe the curve. Precision required from this kind of conversion is very subjective as it depends on the kind of application one is looking to build upon it. In the current scenario, proposed work involves finding paths in a geographical map where a primary requirement is to convert a curve representing any kind of road into polylines in order to aid in navigation. In this case an approach is required which can give some flexibility with respect to the precision of the polylines in approximating a given curve. Such flexibility should be available for tuning at run-time in order to be truly effective.

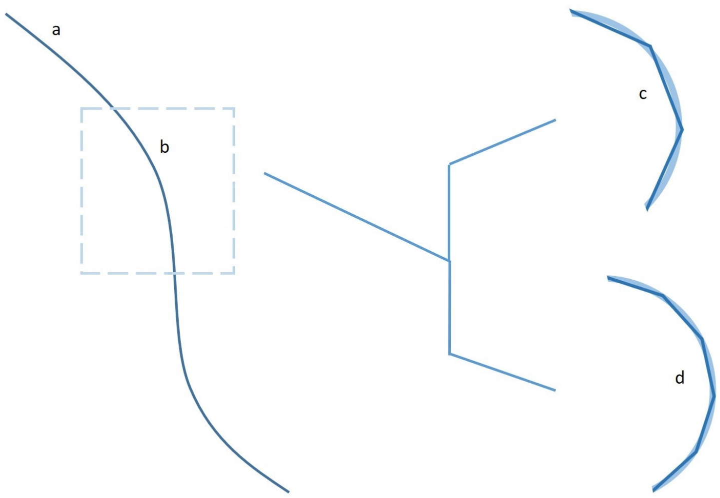


Fig 6.1: A part (b) of curve (a) can be converted to polylines. Using less lines (c) results in a less accurate approximation whereas using more lines (d) results in a more accurate approximation

The basic premise of this concept is that, digitally speaking, any curve is, at the minutest level, a collection of lines. Higher the number of lines, the higher the quality of curvature. Fig 6.1

describes that any curve can be subdivided into arcs and further these arcs can be approximated using lines. Using more lines results in a more accurate approximation while using less lines results in a less accurate approximation.

The main contributions of the proposed work are as follows -

1. It gives a new methodology for spline calculation.
2. The proposed methodology is faster and dynamic in nature.
3. It is tunable at runtime.
4. The efficiency of the proposed methodology improves with time due to caching.

6.2 Motivation

Proposed work revolves around finding useful navigation paths in satellite maps and therefore there is a requirement for converting irregular-shaped paths into polylines. The chosen method also needs to be easily implementable and the precision required is variable. The pre-existing methods used machine learning and other complicated approaches for this which are complicated and essentially prove to be overkill for such tasks. This is the motivation behind this research.

6.3 Proposed Methodology

Part 1: Core Idea

The primary idea is that any curve can be divided into sub-curves or curved sections which can be thought of as arcs of a circle. Given a geographical map, let's focus on a particular curved section of a road. In this curved section, sub-sections can be formed which are in the form of arcs. Given an arc, the radius of the circle which it belongs to can be found using one of the aforementioned methods.

The curve can be of any arbitrary curvature. The initial approach would be to divide it into arcs and for each arc the best-fitted lines will be identified in order to reconstruct it for using in our desired application. The precision of fitting will be customizable depending on the tuning factor. With that knowledge, proposed formula can be used which is described as follows.

After preprocessing the data, a part of the curve - an arc - with estimated radius 'R' is achieved based on one of the methods defined in literature. A hexagon can fit a circle using 6 lines with lengths equal to the radius of the circle, say 'R'. The circumference of such a circle will be definitely greater than 6R. It can be found exactly how much the circumference is greater than 6R and that is the initial challenge.

In first algorithm, rather than fixing the radius, it can be defined at run time by using the concept of tangent (tan) ratios. The tangent ratio in a triangle represents the ratio of the perpendicular side by adjacent side. As described in Fig 6.2, the tuning factor τ will be used to construct a right-angled triangle with R and the extrapolated center, say O, of the circle.

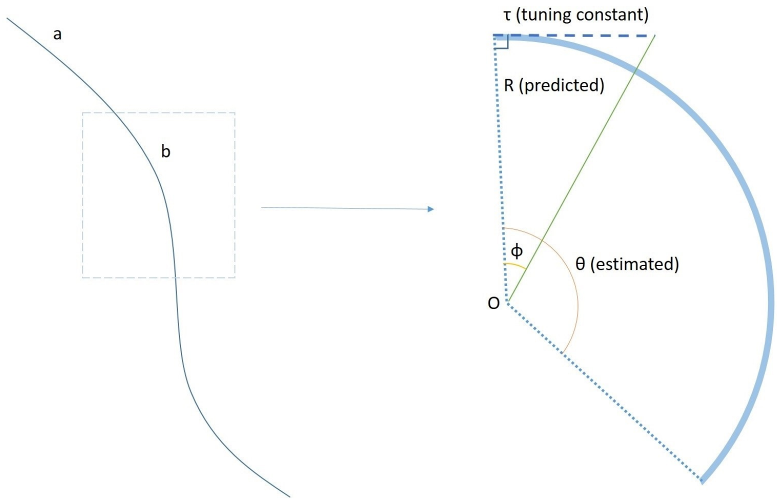


Fig 6.2: A part (b) of curve (a) is to be converted to polylines. For this a perpendicular τ of tunable length is projected from one end of the estimated radius R.

This results in an angle being created at the center, let's say τ ϕ, which will vary depending on the value of τ that is selected. At the point O, the two radii of the arc form an angle of θ among themselves.

Consider the following relation between R, τ and ϕ

$$\tan \phi = \tau / R \tag{6.1}$$

where, τ is the tuning factor and R is the estimated extrapolated radius of the curve, therefore the value of the subpart angle is as follows

$$\phi = \tan^{-1}(\tau / R) \tag{6.2}$$

Therefore, for a selected tuning factor τ, the corresponding value of ϕ can be calculated using the above formula. Note that the value of τ here actually approximates an arc of the curve, having the angle ϕ, as a straight line. This multiplicative factor can be extrapolated to find how many lines will be required to approximate the given arc which is essentially θ/ϕ; if this ismultiplied by τ it gives the total length of the projected path. The formula therefore is - number of lines according to τ (tuning factor) is θ/ϕ, where $\phi = \tan^{-1}$ (τ/R) and the formula for the total length would be

$$\tau\theta/\tan -1(\tau/R) \tag{6.3}$$

Hence as per the need in a given value of θ, to get the number of lines to form that curve, the formula (Eq 6.3) can be used.

The constant τ (greater than 0) can be adjusted depending on how accurate the result is required to be. For small calculations, it can be set to 1 and for more accurate ones it can be reduced to smaller values. For very small values of τ, tan-1 (τ/R) will be so small that the actual curve and τ will coincide almost exactly.

The second approach is to tune the angle at the center. As described in Fig 6.2, the tuning angle τ will be used to construct a right-angled triangle with R and the extrapolated center, say O, of the circle.

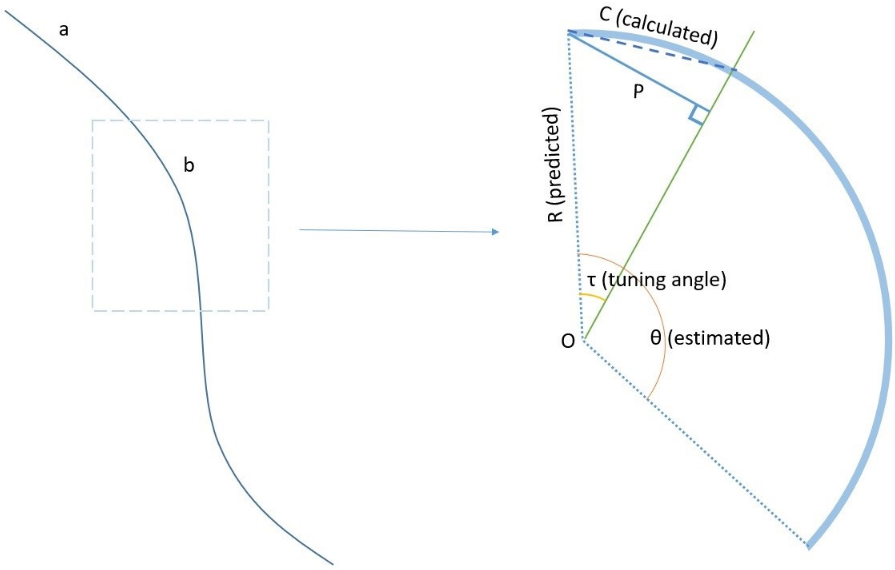


Fig 6.3: A part (b) of curve (a) is to be converted to polylines. For this a tunable angle τ is used to construct an isosceles triangle having its similar sides of length R

This results in Fig 6.3 is a line being created at the edge, let's call it C, which will vary depending on the value of τ that is selected.

Consider the triangle described by CO. In this triangle, a perpendicular P is drawn from one of the end points of C both of which lie on the circumference of the estimated circle. Let's say the angle made by C on R is x, then we have,

$$R\sin\tau = C\sin x \tag{6.4}$$

where τ is the tuning angle and R is the estimated extrapolated radius of the curve. Given that x = 90 - $\tau/2$ (isosceles triangle), using trigonometric identities we get,

$$\begin{aligned}
&R\sin\tau = C\cos(\tau/2),\\
&C = [R\sin\tau]/[\cos(\tau/2)],\\
&C = 2R\sin(\tau/2)\\
&\quad 2R\theta\sin(\tau/2)/\tau
\end{aligned} \tag{6.5}$$

Therefore, for a selected tuning angle τ, the corresponding value of C can be calculated using the above formula. Note that the value of C here actually approximates an arc of the curve, having the angle τ, as a straight line. This multiplicative factor can be extrapolated to find how many lines will be required to approximate the given arc which is essentially θ/τ; if this is multiplied by C it gives the total length ofthe projected path. The formula therefore is - number of lines according to τ (tuning angle) is θ/τ and the formula for the total length would be

Hence as per the need in a given value of θ, to get the number of lines to form that curve, the formula (Eq 6.3) can be used.

The angle τ (greater than 0) can be adjusted depending on how accurate the result is required to be. For very small values of τ, the actual curve and τ will coincide almost exactly.

Part 2: Pool of tuning factors

As explained in the previous section, the tuning factor (τ) can be set depending upon the requirements of the specific application. To enable a convenient selection of τ, a pool of values can be prepared for reference. Values from this pool could represent some average use-case scenarios and can serve as a starting point to implement this algorithm in highly variable applications, e.g. maps., depending on specific requirements.

Part 3: Algorithm for faster implementation

Following is the step-by-step approach to implement the proposed approach in map-related applications.

Step 0: For a selected arc-shaped subsection of the curve to be approximated, check to see if a similar curve has been approximated in the past by comparing its hash. If not, move to Step 1, otherwise stop.

Step 1: Select a value from the pool of values of the tuning factor (τ).

Step 2: According to the value of τ selected in the previous step, calculate the resultant approximation of the current subsection of the curve using the proposed formula (Eq 6.3)

Step 3: Check if the approximation is accurate enough for the requirement of the application. If yes, move to Step 4, else go back to Step 1 to select a different value of the tuning factor τ.

Step 4: Once an approximation is calculated which is accurate enough for the requirement, store the complete information of the calculation, viz. Value of the tuning factor and the relative coordinates of the vertices of the approximated spline, as key-value pairs to form a hash. Since a particular shape/type of curve repeats quite often in maps, this hash can be used directly if a match is found in the next iteration.

6.4 Observations and Inferences

Consider a full circle of radius (R) 7 units. In this case θ will be 360 degrees. From the formula for circumference (2πR) of a circle, its circumference comes out to be approximately 43.9822971503.

The circumference of this circle can be approximated using polylines with the help of the aforementioned technique by choosing a value for the tuning constant τ.

Lets take τ=1 , in this case the total length of polylines approximating the circumference would be -

$$\tau\theta/\tan-1(\tau/R) = 1{*}360/\tan-1(1/7) = 44.2$$
$$2R\theta\sin(\tau/2)/\tau = 2{*}7{*}360{*}\sin(1/2) = 43.9$$

It must be noted that Eq 6.3 fits the lines slightly outside the curve and is useful for cases where the polylines are required to be of a specific length, whereas Eq 6.5 fits the lines slightly inside the curve and is useful for cases where there are supposed to be a fixed number of lines across an arc of a given angle. In Eq 6.3, the value of τ is relatively very small as per the application because this constructs a total of 360 lines across the circumference whereas Eq 6.5 constructs only about 44 lines.

The results of Eq 6.3 can be made more accurate by taking $\tau = 0.5$ which gives the final value of 44.05. Table 6.1 shows the values of estimated total length of polylines for given values of τ and radius 7 units using Eq 6.3. Table 6.2 shows the values of the estimated total length of polylines for given values of τ and radius 7 units using Eq 6.5. The values of the trigonometric ratios are to be calculated in degrees since θ is taken in degrees.

As it can be seen, the accuracy goes on increasing with decrease in the value of the tuning factor τ which can be set as per the accuracy required in a given application. It can also be seen that as the value of the tuning factor is decreased from 1 to 0.1, there is a rise in the number of lines required to approximate the curve. In applications where such accuracy is required, a low value

Table 6.1: Relation between estimated total length of polylines and tuning factor τ as per Eq 6.3

Tuning factor τ	Length of polylines	Approx. lines for curve
1	44.279	44
0.5	44.057	88
0.4	44.030	110
0.2	43.994	219
0.1	43.985	439

Table 6.2: Relation between estimated total length of polylines and tuning factor τ as per Eq 6.5

Tuning factor τ	Length of polylines	Approx. lines for curve
4	43.9734	90
3	43.9773	120
2	43.9801	180
1	43.9817	360
0.5	43.9821	720

of the tuning factor can give higher number of lines resulting in the desired accuracy.

6.5 Simulation and Results

The following tools and technologies are used for the simulation environment.

Language: JAVA 1.8, PYTHON 3 (for plotting the graph).

Editor: INTELLIJ 2018 (Community Version).

Operating System: MacOS 10.16

This test has been run on many data sets with different curves and angles. The results obtained have been presented in Fig 6.4 to Fig 6.6.

Fig 6.4 shows the relation between the tuning factor (τ) and the number of lines required to approximate curves of various given radii and an angle of 60 degrees. Similar to Fig 6.5, it can be seen in Fig 6.6 too that, as the value of the tuning factor is decreased from 1 to 0.1, there is a rise in the number of lines required to approximate the curve. Applications requiring higher accuracy will surely benefit from a reduced value of the tuning factor whereas applications where it is required to have a very high throughput can benefit from an increased value of the tuning factor resulting in the desired performance.

Fig 6.5 shows the relation between the total length of the estimated set of polylines for a curve of a given radius and an angle of 60 degrees. It can be seen from this figure that as the value

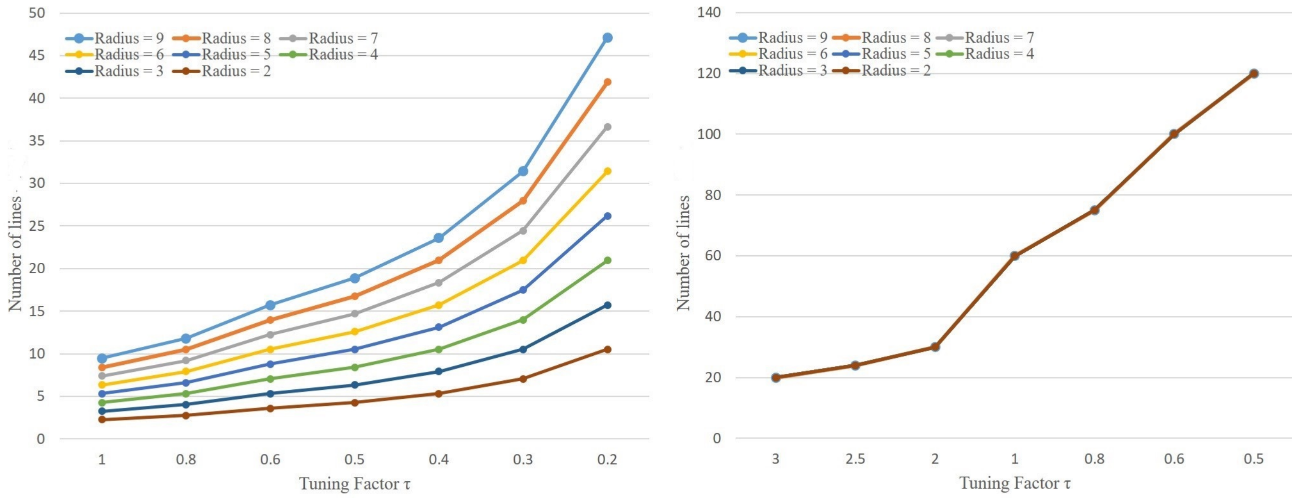


Fig 6.4: Relation between τ and the number of lines required for various radii ($\theta = 60$ degrees) as per Eq 6.3 and Eq 6.5

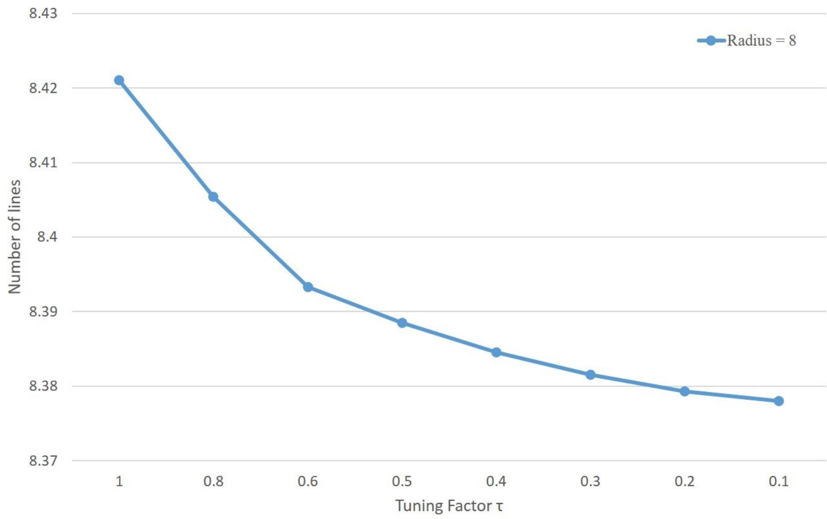


Fig 6.5: Relation between total length of polylines and τ (R = 8, $\theta = 60$ degrees)

of the tuning factor is decreased there is a decrease in the total length of the estimated set of polylines. The value also seems to tend to converge at some point for extremely small values of τ. This agrees with the idea that the estimated polylines form a polygon that will overlap the circle for extremely small values of τ.

As can be observed from Fig 6.6, that using values of τ smaller than a certain threshold (e.g. 0.1) do not result in significant improvement in the accuracy of the resulting polylines.

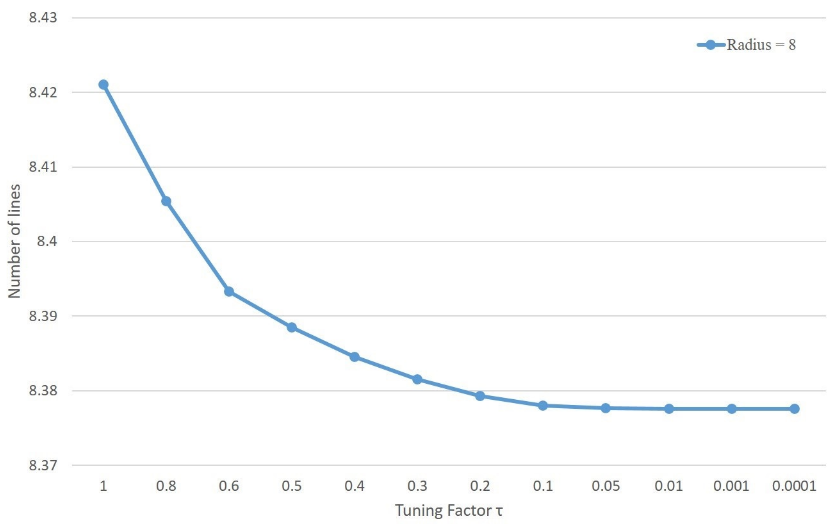


Fig 6.6: Extremely smaller values of τ, do not necessarily improve accuracy by a significant margin

6.6 Conclusion

Thus, using this formula it is possible to find the number of lines that can be approximately fitted into a given curve. Using the tuning factor τ, the user has full control over the accuracy of the calculation. This is especially useful in maps where accuracy and speed need to be balanced for a good user experience. The proposed approach, when used along with the proposed implementation methodology results in improved efficiency for converting curves to polylines for application in maps. There is scope for using this approach in real-life maps in future in order to achieve performance improvements in high-scalability conditions.

Chapter 7

Validating Roundabout Using Spline and Curvature for Map Data

7.1 Introduction

Splines are the mathematical representation of curves. In 2D map, roads is collection of multiple line segments, the curviness of road is represented using small segment of lines and hence we get more geometry for more curvy road. The curviness can be controlled by using any cubic function which should be continuous at the geometrical nodes. Chapter discuss about the identification and validation of a map feature (Roundabouts). The idea is to get the spline from the geometrical points if the roundabout and calculate the kappa value. reverse of the kappa value is the radius of curve for that point. All the points in the roundabout map feature is projected over the spline and for that particular point curvature value is calculated, which indeed provide us radius, then threshold is defined for 10 cm. Radius of roundabouts will be within the threshold value. Then this is cross validate with the validation technique where the bounding box of roundabout is calculated and hence the radius.

In chapter one authors have used the map data to detect the roundabouts using pure geometry. Machine learning is used to get the results on the fuzzy part of detection. It describe the detail process of map making using geometries. Image processing is widely used in identifying the features from the road [17, 18]. Probe data is another source of input in creating and detecting the features in the map. Zinoune et al. uses probe data for identification of roundabout. Other source of input may be the sensor, lidar data.

In this approach, splines are created using Map data (Geometries). Spline creation is widely used in field of Engineering for creation of smooth surfaces. Whole idea of spline creation is over the fitting of functions [19]. The best used function for smoothing of road is cubic. Many Medical and Engineering applications. Jo et al. have proposed the technique of creating the feature in map for autonomous vehicle using three step process, B-spline is used to smooth the sensor data [20]. Berglund et al. worked on obstacle on the road which is detected using the spline technique [21]. Many approaches uses this technique to smooth the trajectory, path, road lane detection, 3d image creation and many more [22, 23, 24, 25].

For solving the problem of roundabout verifications and detection, B-spline in the NURB library

Fig 7.1: Culdisec - special case of roundabout

is used. Once the values are extracted, projection of the points over the spline is done, there can be many technique to do the same but in the proposed work, the nearest point is selected from the set of splines point and then the curvature value is calculated. The radius is reciprocal of curvature and hence the cross verification is done on the basis of the idea that the curvature value will be smooth for the roundabouts over the span of complete spline points. Brandin et al. uses the idea of moving the vehicles on the roundabout and using the vehicular motion detection and stering angular change try to detect the roundabouts [13].

some of the special case like culdisec (Fig 7.1), which have the same entry and exist are exception case of roundabout, there are other similar case like pear drop or sharp acute double split roundabouts. Still we can get the spline out of it and our proposed technique create bounding box on that and get the radius, but there are some interesting roundabouts like in Fig 7.2, is some challenging cases, its more rectangular shaped roundabouts, but this happens mostly when the structure is very small, which is very rare case.

Finally, another approach is used to verify the results with approximate calculations, the technique used is creating the manual bounding box over the feature and calculating the best feated geometry i.e eclipse or the circle, then radius is calculated and verified with the spline calculated technique. The result shows that this approach is easy and practically can be implemented with minimum infrastructure and has the success ration of more than 92% detection.

7.2 Proposed Methodology

The representation of roundabout in 2D form is collection of different lines as describe in Fig 7.3. Curviness of roundabout is defined by its curvature value. Rate of change of slope in theory

Fig 7.2: Rectangular shaped - special case of roundabouts

is the definition of curvature. In mathematical terminology that would be second derivative. Fig 7.3 shows the segment of roundabout having 2 lines, by definition the curvature can be calculated at common points and hence there will not curvature value for the first point. For circular curve, the rate of change of slope is constant and hence the radius which is reciprocal of curvature is fixed. For cubic curve the rate of change of slope will vary with each points and hence the radius is not constant. In Fig 7.4, roundabout is approximate circular geometry, Using B-spline Cubic function the value of curvature at different points are calculated and hence the radius.

For the above representation the roundabout is approximated with 14 geometry points and hence will have 14 curvature and radius values. Similarly the curvature value is calculated for other geometrical figures of roundabout.

When using the spline technique, the points are calculated from the parametric cubic equations. Hence we can have any value of points for that specific function.

Let the Geometrical points be represented by G(x,y) and smoothness of the curve be represented by F(x)=$ax^3 + bx^2 + cx + d$ then,

$$G(x, y) \propto F(x) \tag{7.1}$$

Increasing the points, will increase the smoothness of the curve. Once the B-spline cubic function is defined, predefined set of points can be used to fetch the curvature values.

Identification of roundabout using spline technique is as follows:
Step 1:

From the map data, detecting the loop. In the section phase 1 - deterministic approach

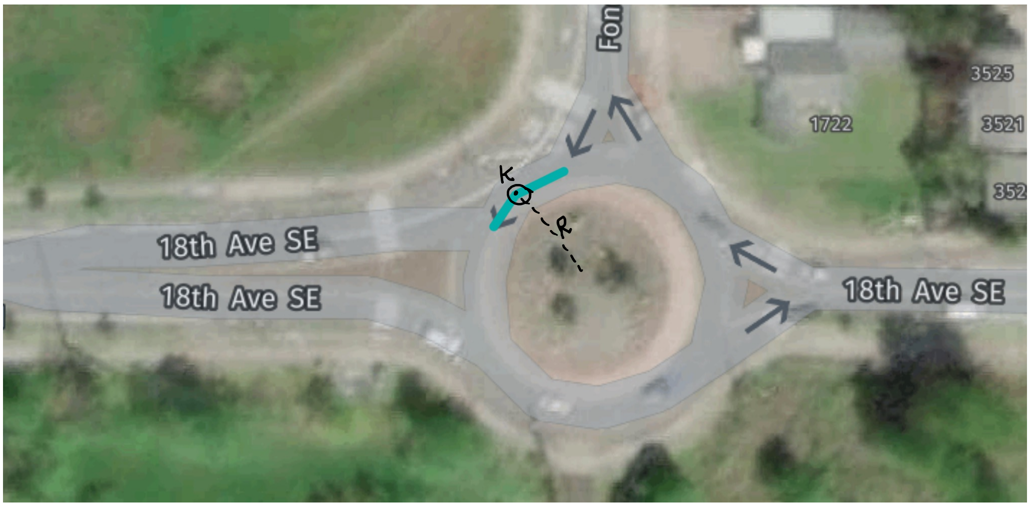


Fig 7.3: Roundabout representation in 2D map

the algorithm is defined.

Step 2:

Creating the spline for the looped set of points.

Assuming the points are correct then using four points creating cubic polynomial equation, calulating dervivatives for finding the heading and curvature values.

Let, A (x,y), B (x_1, y_1), C (x_2, y_2) and D (x_3, y_3), then,

$$\begin{aligned} y &= ax^3 + bx^2 + cx + d, \\ y_1 &= ax_1^3 + bx_1^2 + cx_1 + d, \\ y_2 &= ax_2^3 + bx_2^2 + cx_2 + d, \\ y_3 &= ax_3^3 + bx_3^2 + cx_3 + d \end{aligned} \tag{7.2}$$

Selecting four geometrical points as A, B, C, D. we have considered the common equation and hence the choice of cubic function is best for spline creation. Single function has to pass through all the four point to make the curve smooth.

Solving these equation using Matrix Technique, where

$$\begin{bmatrix} x^3 & x^2 & x & 1 \\ x_1^3 & x_1^2 & x_1 & 1 \\ x_2^3 & x_2^2 & x_2 & 1 \\ x_3^3 & x_3^2 & x_3 & 1 \end{bmatrix} \times \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} y \\ y_1 \\ y_2 \\ y_3 \end{bmatrix} \tag{7.3}$$

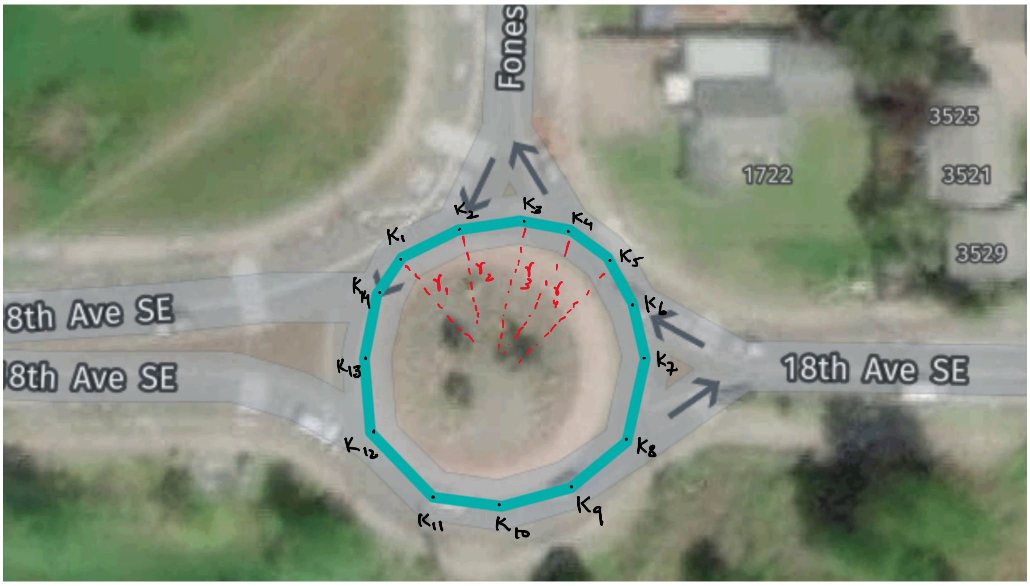


Fig 7.4: Roundabout 2D representation with kappa and radius representation

$$X \times A = Y \tag{7.4}$$

where, A is the constant matrix, X is the matrix having values of x coordinate and Y is the matrix having values of y coordinate.
Using the Inverse Matrix technique

$$A = Y \times X^{-1} \tag{7.5}$$

Using gaussian elimination [76], solve the matrix to get the value of the constants a, b, c, d.

As Y is function of x, y can be denoted as f(x), hence the equation can be represented as,

$$f(x) = ax^3 + bx^2 + cx + d \tag{7.6}$$

First derivative: (slope)

$$f'(x) = 3ax^2 + 2bx + c \tag{7.7}$$

Second derivative: (Bending)

$$f''(x) = 6ax + 2b \tag{7.8}$$

For calculation of Heading, North is considered the base and hence the angle made with reference to Y axis is considered, its value is in range 0 to 360 degree.

For calculation of curvature (signed),

$$\kappa = f''(x)/((1 + f'(x))^2)^{3/2} \tag{7.9}$$

and hence the radius is inverse of curvature, which is given by,

$$R = 1/\kappa \tag{7.10}$$

It is observe that the absolute difference of the curvature radius in the roundabouts is in the range 5m-10m.

Assuming the points are not correct

Using the B-spline technique, calculate knot vectors and finally get the curvature by calculting κ value.

$$\begin{gathered} N_{i,1}(t) = \begin{cases} 1, & t_i \le t < t_{i+1} \\ 0, & \text{otherwise} \end{cases} \\ N_{i,k}(t) = \tfrac{t - t_i}{t_{i+k-1} - t_i} N_{i,k-1}(t) + \tfrac{t_{i+k} - t}{t_{i+k} - t_{i+1}} N_{i+1,k-1}(t) \end{gathered} \tag{7.11}$$

where,
$t_i (0 \le i \le n + k)$ are called knot values
$t_i = 0$ if $i < k$;
$t_i = i - k + 1$ if $k \le i \le n$;
$t_i = n - k + 2$ if $i > n$

Step 3:

Calculate the relative difference of curvatures having tolerance threshold of 10 meters.

Let κ_x is the first curvature, κ_m is the consecutive the next curvature value. The relative difference in the value is

$$d_r = \frac{|r_x - r_m|}{r_x} \qquad [As, r_x = \frac{1}{\kappa_x}, r_m = \frac{1}{\kappa_m}] \tag{7.12}$$

$$d_r = \frac{|\kappa_m - \kappa_x|}{\kappa_m} \tag{7.13}$$

for m points in the roundabout, average change in the curvature is

$$\frac{\sum_{n=1}^{m} d_n}{m - 1} \le 10 \tag{7.14}$$

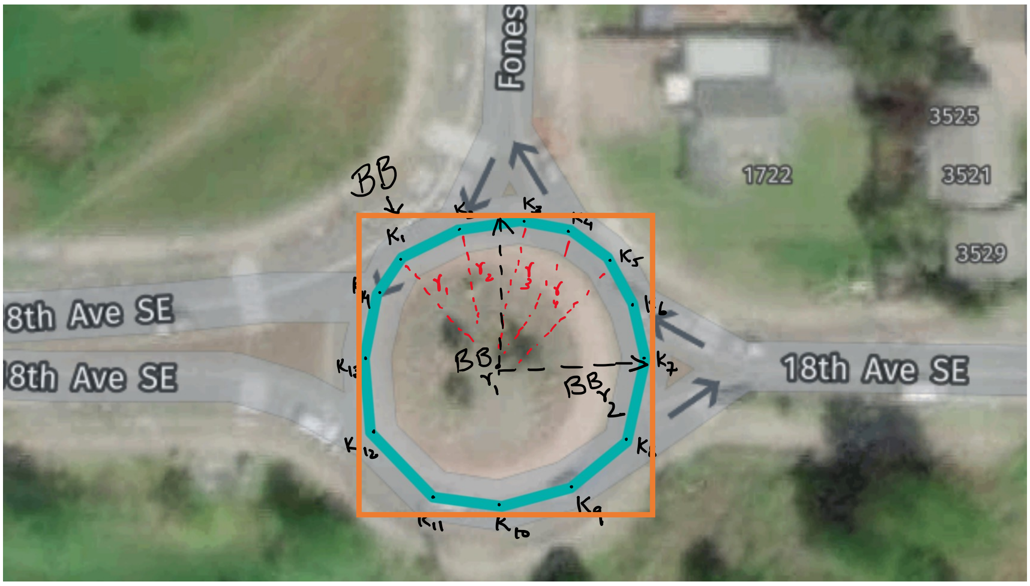


Fig 7.5: Roundabout representation in Bounding Box

Step 4:

Validation (by Bounding box technique)

- Get the bounding box of the closed figure:
  This can be done by identifying the polygon using the JTS library
- Identify the fitted geometrical shape from it (circle or eclipse)
- Get the radius:
  For circle there will not be any change, but for Eclipse we have 2 radius(average them to get the single radius.)
- Threshold with some tolerance value (R +- Tolerance value)

In the Fig 7.5, the orange line enclosing the roundabout is the bounding box(BB). There are two radii, width and height variations (BB_r1, BB_r2), and geometrical variations like square and rectangle. Get the bounding box of the roundabout, assume that the bounding box is the best fitted approximated cover for the map-feature, with respect to 2D geometry this will be eclipse or circle.
IF eclipse,

$$\frac{(x-x_0)^2}{a^2} + \frac{(y-y_0)^2}{b^2} = 1 \tag{7.15}$$

where, a and b will be length and breadth of the bounding box then the average radius should be between

$$\frac{a}{2} - 10 \leq r \leq \frac{b}{2} + 10 \qquad where, a < b \tag{7.16}$$

Fig 7.6: Europe geo-boundary [4]

IF Circle, then the radius should be between the $R_c \pm 10$, where R_c is radius of the circle or the half sides of bounding box

7.3 Experimental Setup and Results

Used linux machine (Intel(R) Xeon(R) CPU E5-2650 v4 @ 2.20GHz), Java 1.8 Application, JTS library for the Geometries manipulations.

Target region is described in the Fig 7.6, Europe is further divided into Eastern Europe(EEU) and Western Europe(WEU) [4]. In WEU, total links traversed are 59320831. Firstly all the roundabout links are traversed and then for each roundabouts, a sequence of geometries from collection of links are assembled into one array of Coordinates, below is the sample geojson representation of complete roundabout.

{“id”:“o123”,
‘geometry”:“type”:“LineString”, “coordinates”: [[13.6105, 46.58293], [13.61054, 46.58297], [13.61053, 46.58303], [13.61045, 46.58307], [13.61037, 46.58306], [13.61033, 46.58304], [13.6105, 46.58293]],
“Type”:“Roundabout”,
1 “type”:“Feature”}

Applied proposed bounding box technique to get results, Admin wise data is laoded and anal-

ysed, for example
AdminName: **Villach-Land**
Country: **Österreich**

Table 7.1: Roundabout details for Villach-Land

Arc length	Lower bound	Upper bound	Ratio
54.15	2.23	22.23	1.0
76.76	7.79	27.79	1.0
99.05	13.35	33.35	0.84
118.02	17.24	37.24	0.94
58.47	3.89	23.89	1.0
116.78	16.68	36.68	0.9375
103.78	12.79	32.79	1.0
163.22	28.36	48.36	0.926
101.38	13.35	33.35	1.0
77.48	7.79	27.79	1.0
55.51	3.34	23.34	1.0
41.31	0.0075	20.007	1.0

Table 7.1 shows the roundabouts details in the Villach-Land, Österreich. Each row defines the properties of roundabout, arc length is the total circumference of roundabout, lower bound is 10 less than radius of bounding box, upper bound is 10 more than the radius and ratio is curvatures which are in the range of bounds verses total number of geometrical coordinates of the roundabout. For the complete region it is found that approximately 97% of roundabouts have valid curvature values. Similarly more than 4000 admin areas were parsed in the WEU region such as Baden, Eisenstadt, Weiz,Liezen, Güssing, Jennersdorf, Kirchdorf an der Krems, Spittal an der Drau...etc. for few regions Fig 7.7 like Spittal an der Drau, it is 99.2% roundabouts are within range. Total cumulative for WEU Detection approximately is 92.12.

Similarly for EEU, Total traversal links are 56871879. Some of the admin like Malacky, Námestovo, Pezinok, Liptovský Mikuláš, Zlaté Moravce, Trenčín, Žiar nad Hronom, Nové Mesto nad Váhom, Spišská Nová Ves...etc. for few regions in Fig 7.8 like Zlaté Moravce the roundabouts were 100% in range. Total cumulative for EEU Detection approximately is 89.47.

7.3.1 Analysis of Arc in Roundabout

In Fig 7.9 , Roundabout circumference(arcLength) Analysis is done for WEU and EEU. For WEU, 39.7% of total roundabouts are having arclength less than equal to 80 meters. 34.1% are having arc length netween 80 to 110 meters. more than 73% of runabouts have less than 110 meter circumference. There are very less roundabouts having arclength more than 200 meters. For EEU in Fig 7.10, 36.2% of total roundabout are having arc length less than equal to 80 meters, 35.3% are having in between 80 to 110 meters, 12.8% are having between 110 to 140

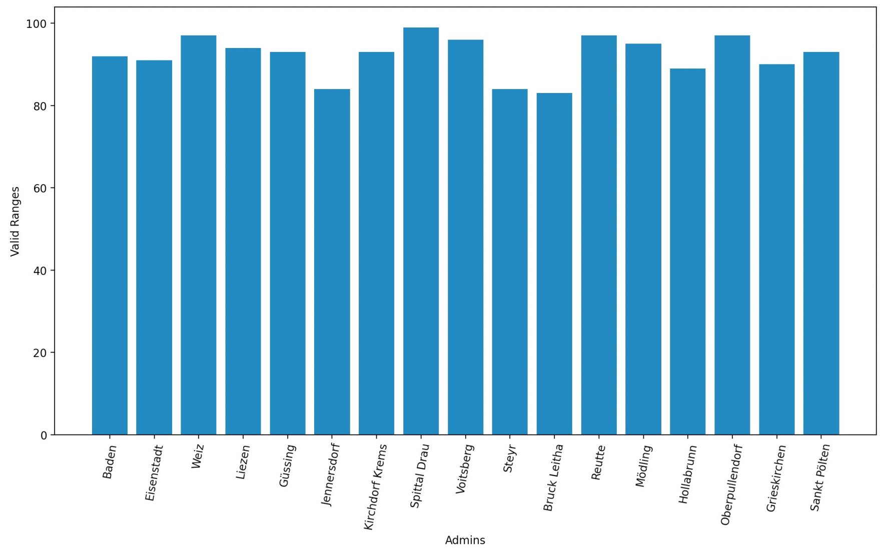


Fig 7.7: Sample valid range of roundabouts in WEU Admins

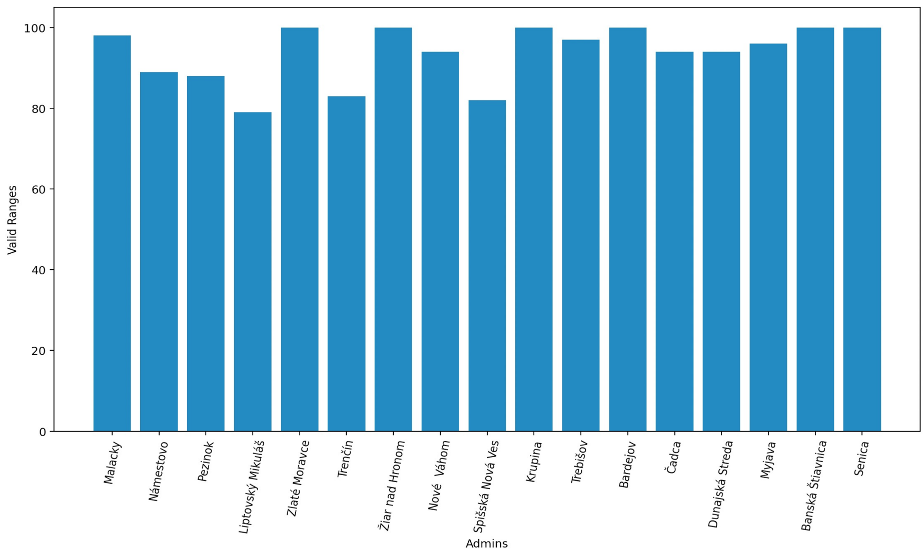


Fig 7.8: Sample valid range of roundabouts in EEU Admins

meters and remaining are very less, to the extend that they can be ignored. Overall majority of roundabouts have arclength less than equal 140 meters, which cover more than 84% of total roundabout in EEU.

7.3.2 Analysis of Radius in Roundabout

With reference to radius of the bounding box, the analysis have been done on complete Europe. Observers results are as follows, For WEU, in Fig 7.11 majority of roundabouts lies within the radius of 40 meters, 86.7% of roundabout have less than equal to 30 meter of radius and cumulatively 91.74% of total roundabout have radius less than equal to 40 meters. For EEU, the results are almost similar with reference to Fig 7.12 more than 95% of total roundabouts have radius less than equal to 40 meters.

7.3.3 Comparison and Discussion

In chapter one, authors have used deterministic technique to identify roundabouts from the loops of road, the identification for the deterministic approach is 80%, and remaining 20% is detected using non deterministic technique, With proposed technique the validation of roundabout in the deterministic technique is increased to more than 90%, as this technique is also deterministic in nature and validate the roundabouts using spline techniques, so this has improved the deterministic validation of roundabout to 10%.

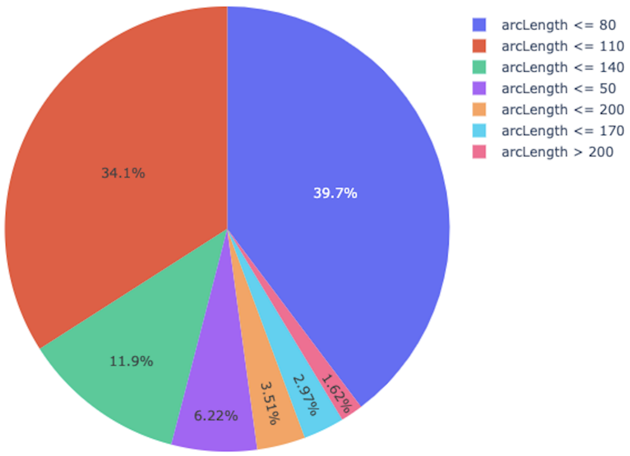


Fig 7.9: Percentage wise representation of roundabout on basis of circumference for WEU

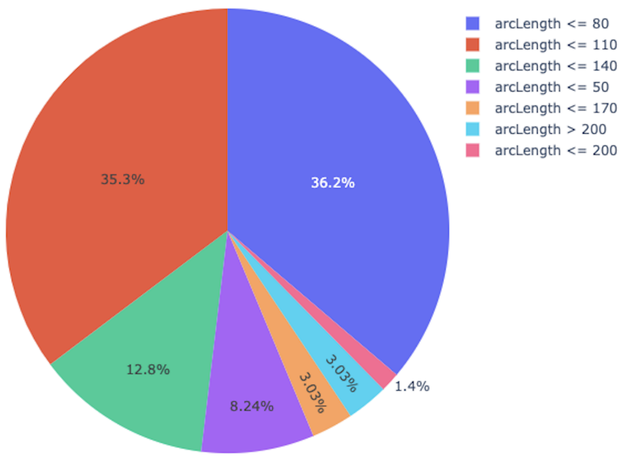


Fig 7.10: Percentage wise representation of roundabout on basis of circumference for EEU

In crux, Roundabouts from the region in Western and Eastern Europe are tested, It is observed that the more the threshold is increased, more is the false detection, on trail and error basis threshold at 10 is selected, which provide the optimum result.

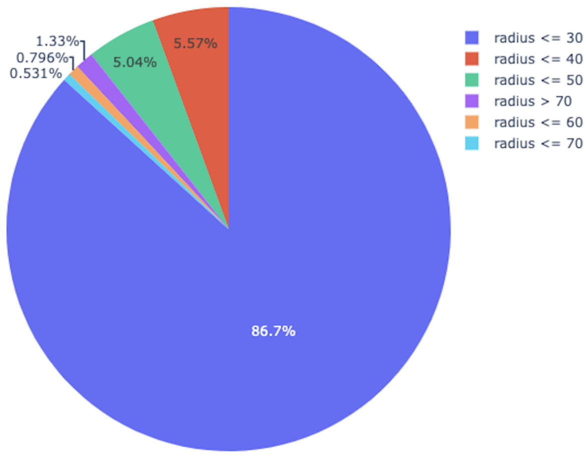


Fig 7.11: Percentage age wise representation of roundabout on basis of radius for WEU

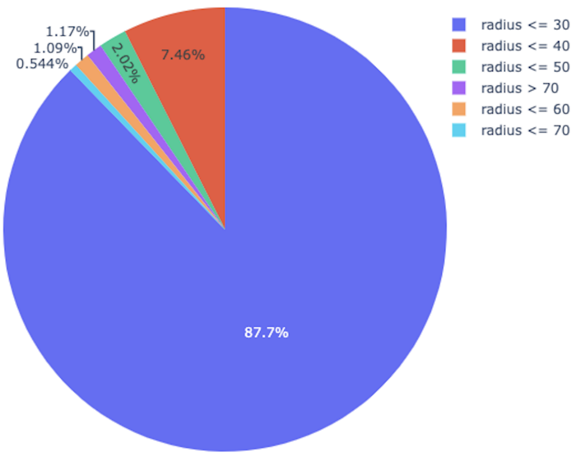


Fig 7.12: Percentage wise representation of roundabout on basis of radius for EEU

Case WEU: Western Europe

On an average best detection are available for the roundabouts in the range from 30 to 60 meters. For the western europe average for this case is 96.36. As the roundabout size increases the detection is decreased, thats due to shape points or number of geometry representing the roundabout. In general the complete WEU detection rate is 92.5

Case EEU: Eastern Europe

For eastern Europe, the case is almost similar with the average detection for the cases in the range 30-60 is 96.5. Eastern Europe complete detection is around 91.9%. Further the deterministic approach described in first chapter along with this approach has improved the result to 10% for deterministic technique.

Table 7.2: Result WEU

Diameter (Meter)	**Detection (limit-10)**
0-30	99.2
30-40	94.3
40-50	96.5
50-60	98.3
60-70	86.7
above 70	80

Table 7.3: Result EEU

Diameter (Meter)	**Detection (limit-10)**
0-30	97.5
30-40	93.6
40-50	97.2
50-60	98.7
60-70	85.7
above 70	78.7

7.4 Conclusion and Future Scope

The verification done in the letter for identifying the roundabout uses spline technique with 92.2% accuracy rate for Europe. With reference to Singh et al., improvement of 12% is achieved in the deterministic technique. Further three point technique can be implemented to get the curvature, with the assumption that points lie on circle and to the extend circle is special case of spline. Savitzky et al. filter can smooth the curve and provide the curvature values.

Chapter 8

Conclusion and Future Work

In conclusion, through this study, it has been demonstrated that just by using the domain knowledge and surface geometry it is possible to correctly solve the problem of identifying roundabouts. The proposed approach involved splitting up the practical scenarios into two - a deterministic one and a non-deterministic one. Using these approaches the required man-hours were significantly reduced and that resulted in improved automation in more than 98% of the cases where the technique was implemented. This methodology has a lot of scope and can be utilized to further automate the identification of roundabouts for maps in multiple countries. Our second work focus on junction inside roundabout, we implemented different models to achieve an accuracy of 81%. In the future, some other attributes, like the speed limit on junction links, can be considered, which may enhance the model's performance in identifying the junctions in roundabouts. We may add sensors and other lidar data to conclude the results with more accuracy.

Our third work, is with respect to kinks in the roads, Result shows that error threshold is directly proportional to detection and so do noise. For frankfut, USA detection was around 91 and 83%. The proposed methodology was tested on two region and results are shared. In future, correction can be added in this approach by selecting neighbouring points of the detected one. with reference to these three point s, spline can be made and hence corrected new points can be added. Further GPS traces and Lidar data can be used to remove the kink.

In our fourth work , we propose a new formulae to approximate the curve. Using this formula it is possible to find the number of lines that can be approximately fitted into a given curve. Using the tuning factor τ, the user has full control over the accuracy of the calculation. This is especially useful in maps where accuracy and speed need to be balanced for a good user experience. The proposed approach, when used along with the proposed implementation methodology results in improved efficiency for converting curves to polylines for application in maps. There is scope for using this approach in real-life maps in future in order to achieve performance improvements in high-scalability conditions.

Finally we propose a basic mathematical technique to validate roundabouts, verification done for identifying the roundabout uses spline technique with 92.2% accuracy rate for Europe. With reference to prior work, improvement of 12% is achieved in the deterministic technique.

Further three point technique can be implemented to get the curvature, with the assumption that points lie on circle and to the extend circle is special case of spline. Savitzky-Golay filter can smooth the curve and provide the curvature values. Authors also worked on security aspect of communications

List of Publications

1. Rakesh Singh, Prashant Singh Rana and Neeru Jindal, *"A novel approach for detecting roundabouts in maps based on analysis of core map data"*, Multimedia Tools and Applications, Springer, 79(41):30785-30811, 2020. [SCI, IF 3.6]

2. Rakesh Singh, Prashant Singh Rana, Neeru Jindal, *"Approach for Detecting Junctions in Roundabout Using Map Data"*, IETE Journal of Research, Taylor & Francis, 1-13, 2023. [SCI, IF 1.5]

3. Rakesh Singh, Prashant Singh Rana and Neeru Jindal, *"A Novel Approach To Identify Kink In 2D Map Using The Spline Technique on Real Map Data"*, Multimedia Tools and Applications, Springer, 1-15, 2023. [SCI, IF 3.6]

4. Rakesh Singh, Prashant Singh Rana, Neeru Jindal, *"An Approach For Validating Roundabout Using Spline and Curvature for Map Data"*, Wireless Personal Communication, Springer, 1-14, 2023. [SCIE, IF 2.2]

5. Rakesh Singh, *"Secure map data storage using encoding by algorithm pool"*. US Patent No - 11165574, 2 November 2021. [Granted]

6. Rakesh Singh, *"Dynamic Geometry Using Virtual Spline For Making Maps"*. US Patent No - 20230154073, 18 May 2023. [Published]

References

[1] Prokop Jehlicka, Daniel Rolf, Hilko Hofmann, and Björn Karge. Trail interpolation, July 2 2019. US Patent 10,337,869.

[2] Gregory M Zaverucha. Approximating polylines by curved paths. In *IEEE International Conference Mechatronics and Automation, 2005*, volume 2, pages 758–763. IEEE, 2005.

[3] Thanh Phuong Nguyen and Isabelle Debled-Rennesson. Decomposition of a curve into arcs and line segments based on dominant point detection. In *Scandinavian Conference on Image Analysis*, pages 794–805. Springer, 2011.

[4] Wikipedia contributors. Europe — Wikipedia, the free encyclopedia, 2021. [Online; accessed 5-November-2021].

[5] Markus Gerke. Automatic quality assessment of road databases using remotely sensed imagery. *PhD thesis, Leibniz Universität Hannover, Germany, No. 261; also in: Deutsche Geodätische Kommission, Reihe C, No. 599*, 2006.

[6] Nicolas Boichis, Jean-Marc Viglino, and Jean-Pierre Cocquerez. Knowledge based system for the automatic extraction of road intersections from aerial images. *International Archives of Photogrammetry and Remote Sensing*, 33(B3):27–34, 2000.

[7] N Boichis, JP Cocquerez, and S Airault. A top down strategy for simple crossroads automatic extraction. *International Archives of Photogrammetry and Remote Sensing*, 32:19–26, 1998.

[8] Maria Elisabeth De Gunst. Knowledge-based interpretation of aerial images for updating of road maps. *Ph.D. thesis, Delft University of Technology, the Netherlands*, 1996.

[9] Bruce W Robinson, Lee Rodegerdts, Wade Scarborough, Wayne Kittelson, Rod Troutbeck, Werner Brilon, L Bondizio, Ken Courage, Michael Kyte, and John Mason. Roundabouts: An informational guide. Technical report, United States. Federal Highway Administration, 2000.

[10] Z 3. Chai. Design of rural roundabout squares. *Technology of Gansu, China*, 23(3):176–178, 2007.

[11] Xinyi Li and Wen Zhang. Automatic object-oriented roundabouts extrction from high resolution multispectral images. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, 42:779–784, 2017.

[12] Laura García Cuenca, Javier Sanchez-Soriano, Enrique Puertas, Javier Fernandez Andrés, and Nourdine Aliane. Machine learning techniques for undertaking

roundabouts in autonomous driving. *Sensors*, 19(10):2386, 2019.

[13] Magnus Brandin. Roundabout detecting arrangement, April 16 2019. US Patent 10,259,322.

[14] Mario Muñoz-Organero and Ramona Ruiz-Blázquez. Detecting different road infrastructural elements based on the stochastic characterization of speed patterns. *Journal of Advanced Transportation*, 2017, 2017.

[15] Marco Di Vaio, Paolo Falcone, Robert Hult, Alberto Petrillo, Alessandro Salvi, and Stefania Santini. Design and experimental validation of a distributed interaction protocol for connected autonomous vehicles at a road intersection. *IEEE Transactions on Vehicular Technology*, 68(10):9451–9465, 2019.

[16] C Setchell and EL Dagless. Vision-based road-traffic monitoring sensor. *IEE Proceedings-Vision, Image and Signal Processing*, 148(1):78–84, 2001.

[17] Alireza Khotanzad and Yaw Hua Hong. Invariant image recognition by zernike moments. *IEEE Transactions on pattern analysis and machine intelligence*, 12(5):489–497, 1990.

[18] Anant Choksuriwong, Hélène Laurent, and Bruno Emile. Comparison of invariant descriptors for object recognition. In *IEEE International Conference on Image Processing 2005*, volume 1, pages I–377. IEEE, 2005.

[19] Clément Zinoune, Philippe Bonnifait, and Javier Ibañez-Guzmán. Detection of missing roundabouts in maps for driving assistance systems. In *2012 IEEE Intelligent Vehicles Symposium*, pages 123–128. IEEE, 2012.

[20] Kichun Jo and Myoungho Sunwoo. Generation of a precise roadway map for autonomous cars. *IEEE Transactions on intelligent transportation systems*, 15(3):925–937, 2013.

[21] Tomas Berglund, Andrej Brodnik, Håkan Jonsson, Mats Staffanson, and Inge Soderkvist. Planning smooth and obstacle-avoiding b-spline paths for autonomous mining vehicles. *IEEE Transactions on Automation Science and Engineering*, 7(1):167–172, 2009.

[22] P. Sudhakara, V. Ganapathy, and K. Sundaran. Transportation navigation planning of an aerial vehicle in a wind turbulent atmosphere using spline interpolation technique. In *2018 International Conference on Power, Energy, Control and Transmission Systems (ICPECTS)*, pages 64–67, 2018.

[23] C. Huang, L. Wang, Z. Zhang, R. Shun-cheung Yeung, A. Bensoussan, and H. Shuhung Chung. A novel spline model guided maximum power point tracking method for photovoltaic systems. *IEEE Transactions on Sustainable Energy*, 11(3):1309–1322, 2020.

[24] T. André Dos Santos Barros, P. J. Dos Santos Neto, M. V. De Paula, A. B. Moreira,

P. S. Nascimento Filho, and E. Ruppert Filho. Automatic characterization system of switched reluctance machines and nonlinear modeling by interpolation using smoothing splines. *IEEE Access*, 6:26011–26021, 2018.

[25] M. Elbanhawi, M. Simic, and R. Jazar. Randomized bidirectional b-spline parameterization motion planning. *IEEE Transactions on Intelligent Transportation Systems*, 17(2):406–419, 2016.

[26] Jean-Baptiste Bordes and Michel Roux. Detection of roundabouts in satellite image. *ISPRS, Workshop on Topographic Mapping from Space*, 36:1, 2006, February.

[27] H Laurent Choksuriwong, A and B Emile. Comparison of invariant descriptors for object recognition. In *IEEE International Conference on Image Processing*, volume 1, pages I–377. IEEE, 2005.

[28] M Ravanbakhsh and CS Fraser. Road roundabout extraction from very high resolution aerial imagery. *International Archives of Photogrametry and Remote Sensing*, pages 19–26, 2009.

[29] Fernando Jorge. Detecting roundabout manoeuvres using openstreetmap and vehicle state. *Masters thesis in systems, control and mechatronics at Chalmers*, 2012.

[30] Min Zhao, David Kathner, Meike Jipp, D Soffker, and Karsten Lemmer. Modeling driver behavior at roundabouts: Results from a field study. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pages 908–913. IEEE, 2017.

[31] Tove Hels and Ivanka Orozova-Bekkevold. The effect of roundabout design features on cyclist accident rate. *Accident Analysis & Prevention*, 39(2):300–307, 2007.

[32] Mai-Britt Herslund and Niels O Jørgensen. Looked-but-failed-to-see-errors in traffic. *Accident Analysis & Prevention*, 35(6):885–891, 2003.

[33] Hilko Hofmann and Prokop Jehlicka. Probe based identification and validation of roundabout junctions, June 6 2017. US Patent 9,672,759.

[34] Mohammed AH Ali and Musa Mailah. Path planning and control of mobile robot in road environments using sensor fusion and active force control. *IEEE Transactions on Vehicular Technology*, 68(3):2176–2195, 2019.

[35] Carlos Cisneros, Luis Morales, Manuel Silvera, and Fernando Campos. Design of mini roundabouts with adaptive traffic lights to improve vehicular travel times and roundabout capacity in metropolitan cities. In *2022 Congreso Internacional de Innovación y Tendencias en Ingeniería (CONIITI)*, pages 1–6. IEEE, 2022.

[36] Yao Zhang, Jiangyan Zhang, and Bin Dong. An optimal management scheme for connected vehicles merging at a roundabout. In *2022 6th CAA International Conference on Vehicular Control and Intelligence (CVCI)*, pages 1–6. IEEE, 2022.

[37] Ahmad Shatnawi, Abderraouf Drine, Mohammad Al-Zinati, and Qutaibah Althebyan. Simulation study of speed control at congested arms of roundabouts. In *2021*

22nd International Arab Conference on Information Technology (ACIT), pages 1–5. IEEE, 2021.

[38] Ashaa Supramaniam, Muhammad Aizzat Zakaria, Baarath Kunjunni, Mohamad Heerwan Bin Peeie, A Fakhri Ab Nasir, and Muhammad Izhar Ishak. Estimation of electric vehicle turning radius through machine learning for roundabout cornering. In *2021 IEEE 19th Student Conference on Research and Development (SCOReD)*, pages 329–332. IEEE, 2021.

[39] Salma Eleuch, Nadia Khouja, Tomaso Erseghe, and Fethi Tlili. Feature-based vehicle tracking at roundabouts in visual sensor networks. In *2020 17th International Multi-Conference on Systems, Signals & Devices (SSD)*, pages 167–172. IEEE, 2020.

[40] Mahendren Sutharsan, Shehan Rajakaruna, SY Jayaweera, JACM Jayaweera, and Subramaniam Thayaparan. Vision-based adaptive traffic light controller for single intersection. In *2020 5th International Conference on Information Technology Research (ICITR)*, pages 1–6. IEEE, 2020.

[41] Mersha Bemnet Wondimagegnehu and Getachew Alemu. Short time traffic flow identification and prediction at road junctions using recurrent multilayer perceptron. In *2017 IEEE AFRICON*, pages 15–20. IEEE, 2017.

[42] Antoni Buades and R Grompone von Gioi. Visual system inspired algorithm for contours, corner and t-junction detection. In *2016 6th European Workshop on Visual Information Processing (EUVIP)*, pages 1–6. IEEE, 2016.

[43] Tsin Hing Heung, Tin Kin Ho, and Yu Fai Fung. Coordinated road-junction traffic control by dynamic programming. *IEEE Transactions on Intelligent Transportation Systems*, 6(3):341–350, 2005.

[44] TK Ho. Fuzzy logic traffic control at a road junction with time-varying flow rates. *Electronics letters*, 32(17):1625–1626, 1996.

[45] Ben Waterson and Simon Box. Quantifying the impact of probe vehicle localisation data errors on signalised junction control. *IET Intelligent Transport Systems*, 6(2):197–203, 2012.

[46] Murat Ekinci and Barry T Thomas. Road junction recognition and turn-offs for autonomous road vehicle navigation. In *Proceedings of 13th International Conference on Pattern Recognition*, volume 3, pages 318–322. IEEE, 1996.

[47] Alberto Broggi, Pietro Cerri, Stefano Debattisti, Maria Chiara Laghi, Paolo Medici, Daniele Molinari, Matteo Panciroli, and Antonio Prioletti. Proud—public road urban driverless-car test. *IEEE Transactions on Intelligent Transportation Systems*, 16(6):3508–3519, 2015.

[48] Mehdi Ravanbakhsh, Christian Heipke, and Kian Pakzad. Knowledge-based road junction extraction from high-resolution aerial images. In *2007 Urban Remote Sens-*

ing Joint Event, pages 1–8. IEEE, 2007.

[49] Xiaoguang Yang, Xiugang Li, and Kun Xue. A new traffic-signal control for modern roundabouts: method and application. *IEEE Transactions on Intelligent Transportation Systems*, 5(4):282–287, 2004.

[50] Dung Van Than and Tegoeh Tjahjowidodo. A direct method to solve optimal knots of b-spline curves: An application for non-uniform b-spline curves fitting. *PLoS One*, 12(3):1–24, 2017.

[51] Jörg Peters. Efficient one-sided linearization of spline geometry. In *Mathematics of Surfaces*, pages 297–319. Springer, 2003.

[52] Mariusz Zygmunt, Tadeusz Gargula, and Przemysław Klapa. Reduction of circular arcs in european cadastral systems—the proposal of a solution referring to the recommendations of the inspire data specification on cadastral parcels. *ISPRS International Journal of Geo-Information*, 9(1):27, 2020.

[53] Dong Liu, Danping Gu, Danny Smyl, Jiansong Deng, and Jiangfeng Du. B-spline-based sharp feature preserving shape reconstruction approach for electrical impedance tomography. *IEEE transactions on medical imaging*, 38(11):2533–2544, 2019.

[54] Shang Gao, Guiyun Tian, Xuewu Dai, Xuefeng Jiang, Deren Kong, Yan Zong, and Qiuji Yi. A b-spline method with ais optimization for 2-d iot-based overpressure reconstruction. *IEEE Internet of Things Journal*, 7(3):2005–2013, 2019.

[55] Jiguang Dai, Tingting Zhu, Yang Wang, Rongchen Ma, and Xinxin Fang. Road extraction from high-resolution satellite images based on multiple descriptors. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:227–240, 2020.

[56] David C Andrade, Felipe Bueno, Felipe R Franco, Rodrigo Adamshuk Silva, João Henrique Z Neme, Erick Margraf, William T Omoto, Felipe A Farinelli, Angelo M Tusset, Sergio Okida, et al. A novel strategy for road lane detection and tracking based on a vehicle's forward monocular camera. *IEEE Transactions on Intelligent Transportation Systems*, 20(4):1497–1507, 2018.

[57] Pengpeng Sun, Xiangmo Zhao, Zhigang Xu, Runmin Wang, and Haigen Min. A 3d lidar data-based dedicated road boundary detection algorithm for autonomous vehicles. *IEEE Access*, 7:29623–29638, 2019.

[58] Xiaohui Li, Zhenping Sun, Dongpu Cao, Zhen He, and Qi Zhu. Real-time trajectory planning for autonomous urban driving: Framework, algorithms, and verifications. *IEEE/ASME Transactions on mechatronics*, 21(2):740–753, 2015.

[59] Dequan Zeng, Zhuoping Yu, Lu Xiong, Zhiqiang Fu, Peizhi Zhang, and Hongtu Zhou. *dbo* trajectory planning and *hahp* decision-making for autonomous vehicle

driving on urban environment. *IEEE Access*, 7:165365–165386, 2019.

[60] Matthias Schreier, Volker Willert, and Jürgen Adamy. Compact representation of dynamic driving environments for adas by parametric free space and dynamic object maps. *IEEE Transactions on Intelligent Transportation Systems*, 17(2):367–384, 2015.

[61] Weichao Wang, Lei Jiang, Shiran Lin, Hui Fang, and Qinggang Meng. Imitation learning based decision-making for autonomous vehicle control at traffic roundabouts. *Multimedia Tools and Applications*, 81(28):39873–39889, 2022.

[62] Ercan Avşar and Yağmur Özinal Avşar. Moving vehicle detection and tracking at roundabouts using deep learning with trajectory union. *Multimedia Tools and Applications*, 81(5):6653–6680, 2022.

[63] Jelle De Bock and Steven Verstockt. Road cycling safety scoring based on geospatial analysis, computer vision and machine learning. *Multimedia Tools and Applications*, 82(6):8359–8380.

[64] Alexander Gribov. Approximate fitting a circular arc when two points are known. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 2, pages 15–16. IEEE, 2017.

[65] Ji-Hwei Horng and Johnny T Li. A dynamic programming approach for fitting digital planar curves with line segments and circular arcs. *Pattern Recognition Letters*, 22(2):183–197, 2001.

[66] Wenhua Wan and José A Ventura. Segmentation of planar curves into straight-line segments and elliptical arcs. *Graphical models and image processing*, 59(6):484–494, 1997.

[67] Sheng-Feng Qin, David K Wright, and Ivan N Jordanov. On-line segmentation of freehand sketches by knowledge-based nonlinear thresholding operations. *Pattern Recognition*, 34(10):1885–1893, 2001.

[68] Johan Ludvig Heiberg et al. *The thirteen books of Euclid's Elements*, volume 1. The University Press, 1956.

[69] B.G. Heiberg, J. L & Teubneri. ed. euclid. the thirteen books of euclid's elements of geometry. *(English translation by Richard Fitzpatrick). ISBN 978-0-6151-7984-1*, 2008.

[70] Jeremy Avigad, Edward Dean, and John Mumma. A formal system for euclid's elements. *The Review of Symbolic Logi*, 2(4):700–768, 2008.

[71] Burke. Paul. Circlesphere. *http://paulbourke.net/geometry/circlesphere/*, 2010.

[72] Hyungjun Park. Choosing nodes and knots in closed b-spline curve interpolation to point data. *Computer-Aided Design*, 33(13):967–974, 2001.

[73] Erastus H Lee and George E Forsythe. Variational study of nonlinear spline curves.

SIAM review, 15(1):120–133, 1973.

[74] Rakesh Singh. Roundabout junctions dataset, 10.21227/n7m1-ft80. 2020.

[75] James Arvo. *Graphics gems II.* Elsevier, 2013.

[76] Joseph F Grcar. Mathematicians of gaussian elimination. *Notices of the AMS*, 58(6):782–792, 2011.