CHALMERS
UNIVERSITY OF TECHNOLOGY

A study on gap acceptance in roundabouts in Sweden

Master's thesis in the Master's Programme Infrastructure and Environmental Engineering

VICTOR HANSSON

DEPARTMENT OF ARCHITECTURE AND CIVIL ENGINEERING
CHALMERS UNIVERSITY OF TECHNOLOGY

Gothenburg, Sweden 2025
www.chalmers.se

DEPARTMENT OF ARCHITECTURE AND
CIVIL ENGINEERING
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2025
www.chalmers.se

MASTER'S THESIS ACEX30

A study on gap acceptance in roundabouts in Sweden

Master's Thesis in the Master's Programme Infrastructure and Environmental Engineering

VICTOR HANSSON

Department of Architecture and Civil Engineering
Division of Geology and geotechnics
Urban Mobility Systems
CHALMERS UNIVERSITY OF TECHNOLOGY

Göteborg, Sweden 2025

A study on gap acceptance in roundabouts in Sweden

Master's Thesis in the Master's Programme Infrastructure and Environmental Engineering
VICTOR HANSSON

Examensarbete ACEX30
Institutionen för arkitektur och samhällsbyggnadsteknik
Department of Architecture and Civil Engineering
Division of Geology and geotechnics
Urban Mobility Systems
Chalmers University of Technology
SE-412 96 Göteborg
Sweden
Telephone: + 46 (0)31-772 1000

Department of Architecture and Civil Engineering
Göteborg, Sweden, 2025

A study on gap acceptance in roundabouts in Sweden

Master's thesis in Infrastructure and Environmental Engineering

VICTOR HANSSON

Department of Architecture and Civil Engineering
Division of Geology and geotechnics
Urban Mobility Systems
Chalmers University of Technology

ABSTRACT

Traffic levels and congestion is rising both globally and in Sweden and is causing issues related to time loss, health problems and pollution. Traffic analysis software is a powerful tool to help planning and optimization of the road network to tackle these issues and the accuracy of their predictions is of a vital importance.

This thesis aims to examine the use of a new method to collect data regarding gap acceptance parameters. The method consisted of recording video footage of three different roundabouts in the Greater Gothenburg area and implementing the AI-based video analysis software GoodVision to extract data from the recordings to then analyze the gap acceptance parameters.

Raff's method was used to calculate critical gap values at all roundabouts at 3.68 seconds, 3.81 seconds and 3.86 seconds respectively. Making up a combined critical gap value of 3.73 seconds. Comparative analysis was performed by using the traffic analysis software SIDRA Intersection and measuring the results of using the critical gap value based on the data collected at the location and the standardized critical gap value in the program. The results showed substantial differences, one scenario displayed an increased capacity reaching up to 25% when using critical gap values calculated from local data. Differences in results of this magnitude greatly affects the decision making in traffic planning and could ultimately be the difference maker between investing in new infrastructure or not.

Parameters affecting why a gap is accepted or rejected were analyzed by implementing machine learning classification algorithms. The results showed that the individual driving behavior of each driver had a higher impact on the decision of accepting or rejecting a gap than roundabout geometry or vehicle type in this thesis.

Key words:

Gap acceptance, Critical gap, Single-lane roundabout, Machine Learning, GoodVision, Sidra Intersection.

En studie om tidsluckor i cirkulationsplatser i Sverige

Examensarbete inom masterprogrammet Infrastruktur och miljöteknik

VICTOR HANSSON

Institutionen för arkitektur och samhällsbyggnadsteknik
Avdelningen för Geologi och geoteknik
Väg och trafik
Chalmers tekniska högskola

SAMMANFATTNING

Trafiknivåerna och graden av trängsel ökar både globalt och i Sverige, vilket leder till problem relaterade till tidsförluster, hälsorisker och miljöföroreningar. Trafikanalysprogramvaror utgör ett kraftfullt verktyg för planering och optimering av vägnätet i syfte att hantera dessa utmaningar, där noggrannheten i deras prognoser är av avgörande betydelse.

Syftet med detta examensarbete var att undersöka användningen av en ny metod för datainsamling avseende tidsluckeacceptans i cirkulationsplatser, med målsättningen att förbättra precisionen i resultaten från trafikanalysprogramvara. Den tillämpade metoden för insamling av tidsluckedata har bestått av att filma tre cirkulationsplatser i Östra Göteborg, varpå det AI-baserade videoanalysverktyget GoodVision har använts för att extrahera data från filmerna.

Raff's metod tillämpades för beräkning av kritiska tidsluckor och resultat blev 3.68 sekunder, 3.81 sekunder respektive 3.86 sekunder vid de tre cirkulationsplatserna. Det totalt sammanvägda värdet på den kritiska tidsluckan blev 3.73 sekunder. En jämförande analys genomfördes med hjälp av trafikanalysprogramvaran SIDRA Intersection, där de kritiska tidsluckorna baserade på lokala data ställdes mot programmets standardiserade värden. Resultaten indikerade en kapacitetsökning på upp till 25 procent vid användandet av kritiska tidsluckor baserade på lokala data.

Vidare analyserades de parametrar som påverkar varför en lucka accepteras eller avvisas genom implementering av maskininlärningsbaserade klassificeringsalgoritmer. Resultaten visade att det individuella körbeteendet hos enskilda förare hade större inverkan på beslutet att acceptera eller avvisa en tidslucka än såväl cirkulationsplatsens geometriska utformning som fordonstypen.

Nyckelord:
Tidsluckeacceptans, Kritisk tidslucka, Enfilig cirkulationsplats, Maskininlärning, GoodVision, Sidra Intersection

Contents

Preface

I would like to take the opportunity in this section to express my gratitude to everyone who participated in and supported the writing of this thesis. Firstly, I would like to give a major thanks to my examinator at Chalmers University of Technology, Kun Gao, who has been apart of this thesis since the beginning and helped shape the foundation. I would also like to give credit to my supervisors, Omkar Parishwad and Arsalan Najafi, for providing me with extensive support in both the writing and the technical aspects of this thesis.

Additionally, I would like to emphasize my gratitude to Norconsult AB which I have written this thesis in collaboration with and especially to my supervisor at the company, Viktor Sköldstedt. Which not only provided valuable knowledge and support, but also a welcoming environment to work in which made the whole thesis experience enjoyable.

1 Introduction

With increasing populations and rising levels of vehicle ownership, traffic congestion has become a daily challenge in a majority of cities worldwide and Sweden is no exception. Both in the Swedish capital, Stockholm and in the city which serves as the study area for this project, Gothenburg, the congestion has been deemed so severe that congestion tax has been implemented to counteract the rising levels of traffic (Transportstyrelsen, 2025). Traffic congestion contributes to multiple unfavorable consequences, including environmental pollution, loss of time and financial losses. Together with limitations in traffic infrastructure and poor traffic management, congestion is projected to extend even further in the near future (Çolak et al., 2016). Tackling the congestion issue requires thoughtful planning and accurate future predictions. Traffic simulation provides traffic planners and researchers with the possibility to test the implementation of different congestion mitigation measures in a replicated real-life digital model. This ultimately gives the potential to construct and evaluate long-term strategies for road network design and traffic infrastructure expansion.

1.1 Background

Traffic simulations depend on multiple different parameters and variables to provide accurate results, one of the most critical factors is gap acceptance. Gap acceptance refers to the time gap a driver needs to another vehicle in traffic to determine that it is safe to perform actions such as changing lanes or entering a roundabout (Guo et al., 2014). This distance varies significantly among different drivers and is related to several different factors, such as driving behavior and driving experience. It is also depending on outer circumstances, for instance the design and the dimensions of the roundabout. In traffic analysis software, such as SIDRA Intersection, the gap acceptance parameters affect the simulated result to a great extent and are therefore a crucial parameter to get as accurate as possible to real-life values.

1.2 Aim

This thesis aims to investigate the gap acceptance parameters, critical gap and follow-up headway, in Swedish roundabouts and to perform comparative result analysis between the usage of data collected in-field and the usage of standardized data when performing traffic simulations in SIDRA Intersection. The thesis also aims to research what parameters affect gap acceptance and to which extent they do.

1.3 Research questions

- What are the gap acceptance parameters in some specific roundabouts in Sweden and how do they compare to some standardized values?
- To what extent does the use of standardized values compared to the values obtained by in-field data collection effect the results when performing traffic analysis in SIDRA?
- Can AI-processing tools be beneficial to make traffic data collection more efficient?
- Which factors influence gap acceptance decisions, and how strongly does each factor contribute?

1.4 Limitations

The scope of this thesis is limited to studying and analyzing gap acceptance only in roundabouts. The project focuses exclusively on the standard one-lane roundabout. The thesis work will not include roundabouts with other crossing modes of transportation, such as crossing tram lines or bus-only lanes. The data collection will be conducted only at roundabouts in the Greater Gothenburg area due to practical reasons. The findings may not generalize to other contexts in Sweden.

2 Theoretical background

The following chapter serve as an overview for the theoretical foundations in traffic engineering, AI-based video analytics, and modeling software relevant to this study. The theoretical background has, aside from providing necessary and useful information about the research area, also assisted in setting the framework for the project work.

With increasing knowledge and findings about previous research and studies, the theoretical background has contributed to adaptations, improvements and limitations on the project work as a whole and has helped to establish a clear and stable foundation for the following parts of the thesis work.

2.1 Literature study

A literature review was conducted during the project work centered primarily around gathering knowledge about previous research work in the gap acceptance area. The literature review found research gaps in the fields of gap acceptance in Sweden, with very few previous studies conducted and even fewer in recent years. With the development and continuous development of artificial intelligence, the possibilities of implementing AI-tools in the traffic analysis are many. The literature review found that AI and traffic analysis is a field that is sparsely explored and that this is a research gap that will be aimed to be filled. The literature review also included obtaining information about other subjects, such as roundabouts as well as information about computer software used in the methodology, GoodVision and SIDRA Intersection.

The search for relevant literature was predominantly performed within the ScienceDirect database of peer-reviewed scientific articles. Search terms were combined with Boolean operators in Scopus to optimize the research process. Examples of search terms and Boolean operators included:

- Gap acceptance AND Roundabout
- Video analytics AND Traffic
- Critical gap OR Follow-up headway AND Traffic simulation

A lot of information has also been gathered through other sources, such as reports and guides from different administrations including both the Swedish transport administration, Trafikverket, and the U.S department of transportation. Notable is especially the extensive application guide "Roundabouts: An Informational Guide: Second Edition" which contributed with substantial amounts of expertise on roundabouts.

2.2 Roundabout

Roundabouts have a history stretching back already before the first cars were invented. Forms of traffic circles used by horse carriages and other modes of transports are known to have been around as early as the 18th century, for example at the world-renowned Arc de Triomphe in Paris (Murray, 2024). The common definition of the modern roundabout is widely considered to be a circular intersection where the traffic circulates, clockwise or anticlockwise depending on right-hand traffic or left-hand traffic, around a central inner circle and has the right of way over the traffic entering the inner circle through the arms of the roundabout. The earliest adaptation of this roundabout is perceived to have been in the United Kingdom in 1966 when the rule of yield-in was first implemented at all circular intersections in the country (Rodegerdts, 2010). Following this alteration, the roundabout gained increasing approval as a choice for traffic intersections in the coming years, not only in the United Kingdom, but also in other European countries and in the United States.

The key advantage of roundabouts compared to other non-signalized intersections is the substantial increase in safety. The design features of the roundabout to direct all traffic flow in the same direction eliminate the most critical part of the standard intersection, the left-turn. It greatly reduces the risk of head-on and broadside collisions which are the leading collisions for severe injury and fatal crashes (Rodegerdts, 2010). In a 2001 study which analyzed the crash and injury reduction after the installation of roundabouts at former stop sign and signal-controlled intersections in United States, the reduction of all collisions were found to be 38%. Crashes involving injuries were reported to be reduced even further with 76% and fatal collisions were estimated to be reduced by up to 90% (Retting et al., 2001).

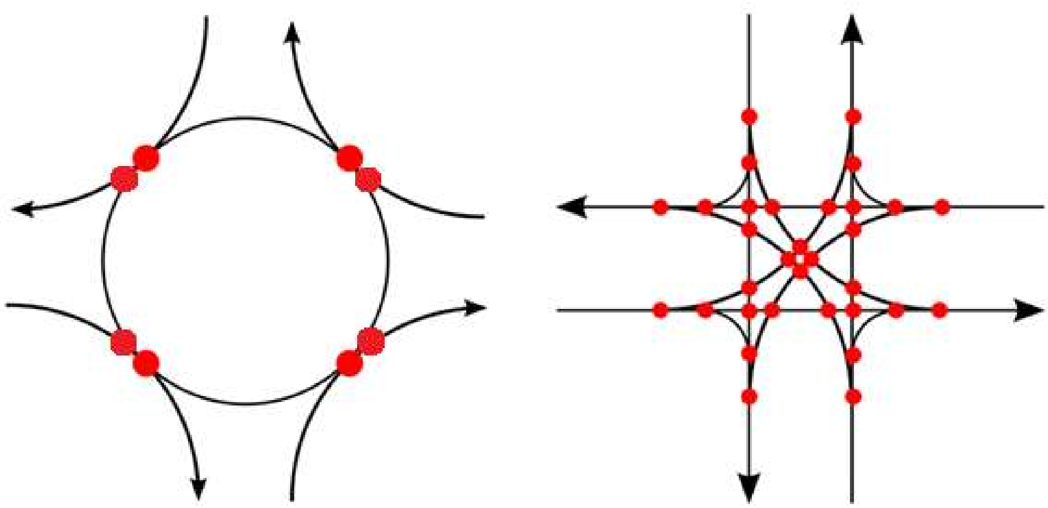

Figure 1: A comparison between the conflict points in a single-lane roundabout compared to a standard non signalized four-way intersection, modified from (Actam, 2008) CC BY 3.0

In a comparative analysis between a single-lane roundabout and a standard four-way intersection the conflict points in the roundabout are 75% less (Rodegerdts, 2010). The roundabout consists of eight total conflict points, four merging conflict points and four diverging conflict points. In the standard four-way intersection there is a total of 32 conflict points which includes eight diverging points and eight merging points, as well as 16 crossing points which is the type of conflict point with the highest risk of fatal or severe injury collisions.

The geometric property of the roundabout serves a crucial purpose to lower the speed both of incoming traffic and of traffic flowing in the inner circle of the roundabout (Rodegerdts, 2010). The speed control in the inner circle of the roundabout not only enhances safety but also, quite counter-intuitively, increases the capacity. The capacity of a roundabout is determined by the gap acceptance of the vehicles queuing to enter the inner traffic circle. A driver can only enter the roundabout when they determine that a gap between the vehicles circulating is large enough to safely maneuver in. With a high speed in the inner circulating traffic flow, the required gap would have to be of a greater distance to be considered an acceptable gap for the entering vehicle. Similarly, a lower speed for the inner circulating traffic flow allows for a lower distance to be considered an acceptable gap for the incoming traffic. A well-designed roundabout should include a balanced speed control that forms a consistent frequency of acceptable gaps without an unnecessarily large restriction in traffic flow speed (Rodegerdts, 2010).

2.3 AI-based video analytics software

AI-based video analytics software uses the recent development of artificial intelligence to automatically analyze video sequences without the necessity for manual work. The use of computer-based vision algorithms for the purpose of reducing manual work with extracting data from traffic videos has been the topic in recent studies. Alomari and Lebdeh presented a model based on machine learning algorithms and computer vision to detect and classify vehicles from videos which achieved a success rate of 93.2% (Alomari & Lebdeh, 2022).

GoodVision is a company which specializes in the AI-based video analytics software area. The company provides two different products for traffic analysis, one for real-time traffic monitoring and one for traffic data collection and analytics from camera recorded videos (GoodVision, 2025a). In the traffic data collection solution, video recorded footage from either fixed camera or flown drones is together with artificial intelligence used to automatically obtain data about traffic volumes and vehicle types with an estimated accuracy of 95-100% (GoodVision, 2025b). By defining the major and minor traffic flows together with the traffic geometry for an intersection, a road, or a roundabout, the AI-model has the possibility to present data for six different parameters (GoodVision, 2025c). The parameter in focus in this thesis is the "*Gap-Acceptance Analysis",* which gives data about rejected and accepted vehicle gaps in the traffic scene.

2.4 Gap acceptance theory

Gap acceptance is defined in the Highway Capacity Manual 2000 as "The process by which a minor-street vehicle accepts an available gap to maneuver" (Transportation Research Board, 2000). The accepted available gap, more commonly referred to as the critical gap $t_c\,(s)$, is the time gap between two vehicles in the major stream that a vehicle in the minor stream finds big enough to enter the major stream conflict zone. Gap acceptance also consists of one additional parameter, as a vehicle from the minor stream cannot enter the major stream directly after another vehicle from the same stream due to the physical length of the vehicle and the required safe stopping distance between the vehicles in the same headway, this variable is commonly referred to as the follow-up headway $t_f\,(s)$ (Chen & Hourdos, 2018). It is clear that both the critical gap t_c and the follow-up headway t_f is dependent on the driving behavior and judgement of each individual driver.

On highways, gap acceptance occurs regularly, both between vehicles in the same stream flow during overtaking, but also with vehicles merging while entering the main roadway, as well as vehicles diverging when exiting the main roadway. The traffic infrastructure on highways is specifically designed to enable all vehicles to maintain a high velocity, which of course also affects the gap acceptance parameters of the highway. The characteristic highway design layout with on-ramps and off-ramps with significant margins and parallel entry and exit angles removes one of the major barriers of gap acceptance in a traditional intersection. The difference in vehicle speed between the major stream vehicles and the waiting minor stream vehicles. By enabling sufficient acceleration before a minor stream vehicle needs to either reject or accept gaps to enter the major stream, the vehicle speeds are balanced, and the size of gaps required for acceptance is greatly reduced compared to the scenario where the minor stream vehicle starts from a standstill and requires both acceleration and reaction time in combination with a sufficiently sized gap.

Gap acceptance in roundabouts does have additional distinct differences from gap acceptance in highways. As previously described, see Chapter 2.2, a roundabout consists of an inner traffic circle, which traffic circulates around. The roundabout also includes a minimum of three legs connected to the inner traffic circle which allows incoming traffic to enter into the inner traffic circle. The traffic circulating around the inner circle has the right of way over the traffic from each leg which wants to enter the inner traffic circle and is therefore considered to be the major stream. The gap acceptance in roundabouts is to a large extent determined by the design dimensions for both the major stream and the minor stream. A large diameter of the inner traffic circle enables a higher vehicle speed for the major stream traffic, which makes for a greater difference in vehicle speed compared to the minor stream. A larger difference in vehicle speed forces the driver in the minor stream to require a larger gap to be able to react and to accelerate to the speed that matches the other vehicles in the major flow (Rodegerdts, 2010). This speed difference is especially clear in the maximum case, when the minor stream vehicle does not accept the initial gap presented when first arriving to the yield line and instead comes to a complete stop. Contrary to the design parameters of the central island, dimensions of the entering legs of the roundabout that can enhance the vehicle speed of the minor stream, such as a greater width of the entry lane and a larger angle between the approaching legs, instead lowers the time required for a gap to be accepted.

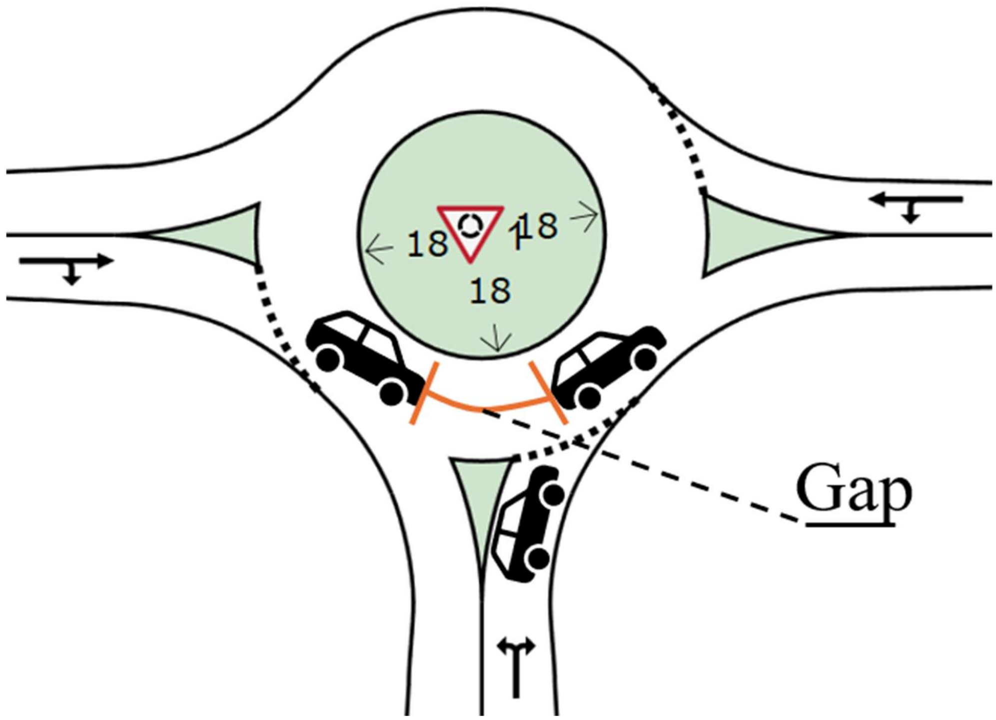

Figure 2: A visual representation of gap in roundabouts with the minor stream wanting to enter, which has to yield for the vehicles in the major stream of the roundabout and is accepting and rejecting the gaps offered between the vehicles

Critical gap and follow-up headway are generally assumed to be following a certain distribution with a variance and an average that can be estimated with several different methods combined with a sufficient amount of data (Guo et al., 2014). Already in 1969 Robert Ashworth presented a research article discussing how to analyze and interpret the data gathered from gap acceptance surveys. Ashworth addressed the issue of biased results derived from the hypothesis that every driver has a fixed critical gap acceptance (ASHWORTH, 1970). This would skew the results when drivers with a high critical gap threshold would only accept large critical gaps and reject the smaller ones, but drivers with a lower threshold would accept the first critical gap that was big enough, including both small and large gaps. To handle this matter Ashworth proposed a model for a gap acceptance curve which assumes the critical gap acceptance to follow a normal distribution and it being displaced by a value of s^2q, where s^2 is the variance in the distribution of critical gaps and where q *(veh/sec)* is the road volume in the major stream. By subtracting this s^2q value from the 50th percentile of the curve, the nonbiased critical gap mean would be possible to calculate given that the major road volume and the variance is known.

In 2025, a research study conducted in collaboration with SIDRA SOLUTIONS presented a method for carrying out roundabout capacity surveys which included the use of digital vehicle trajectories (Chun et al., 2025). This study shows international adoption of AI video analytics, but Swedish roundabout studies are lacking. The method consisted of collecting data by video recording roundabouts and processing the video with the assistance of an AI video-analytics platform called DataFromSky (DataFromSky, 2025). The case study was carried out by video recording with drones at a single-lane roundabout in Raleigh, NC, USA. With the implementation of DataFromSky the gap acceptance parameters critical gap and follow-up headway could be obtained. The mean critical gap was found to be 4.19 sec and 4.47 sec for the southbound and eastbound approaches respectively and 2.79 sec and 2.60 sec for the mean follow-up headway in the same directions.

2.5 Traffic modelling software

The earliest adaptations of traffic modelling software's were published as early as the 1950s when the Transport Research Road Laboratory, TRRL, presented the very first traffic intersection simulation in 1951 (Ross & Gibson, 1977). These initial versions of traffic simulation were designed to be simple and to primarily function manually without the requirement of computer involvement (Mousa, 2004).. Up until this point in history, macroscopic models with limited functionality and less details had been commonly used for traffic simulations which provided simulation results differentiating from the forecasted (Qiao et al., 2021). Following the advancement of computer performance, traffic simulation softwares saw quick enhancement in terms of accuracy and the range of possibilities offered during the following decades.

Signalised Intersection Design and Research Aid, more commonly known as the acronym SIDRA, is a traffic analysis software which was first developed and presented in the original version SIDRA 1 in the 1970s (Akcelik & Associates Pty Ltd, 2025a). During the lead of Dr Rahmi Akçelik, SIDRA has been part of leading research and evolution of traffic analysis for over 50 years. SIDRA Intersection is currently the most used software tool for roundabout capacity in the United States (Akcelik & Associates Pty Ltd, 2025b). SIDRA Intersection offers a wide variety of functions including the ability to model multiple vehicle types, designing complex road networks, including combinations of intersections or roundabouts and precise estimations of congestion influence on traffic flow. SIDRA Intersection includes a set of default critical gap values which is based on extensive data collection and research. These default critical gap values is what the critical gap values obtained by collecting data from Swedish roundabouts will be compared to and the reason for SIDRAs importance in this thesis

3 Methodology

The methodology consisted of five sequential phases: literature review, data collection, video-based data extraction using GoodVision, analysis through SIDRA Intersection and machine learning classification.

3.1 Data collection

The data collection was performed by videorecording roundabouts around the area of Gothenburg, the second largest city in Sweden, located on the country's Westcoast. The data collection process was initiated with an extensive preparation part. Due to the limitations of the camera equipment used in the data collection process, more thoroughly detailed in the subchapter 3.1.5 Camera equipment, it was vital to locate roundabouts with a surrounding that enabled sufficient video recording. To make it possible to get an overview of the complete roundabout in a single camera angle it was necessary to find sites where the elevation is high enough while still directly adjacent to the roundabout.

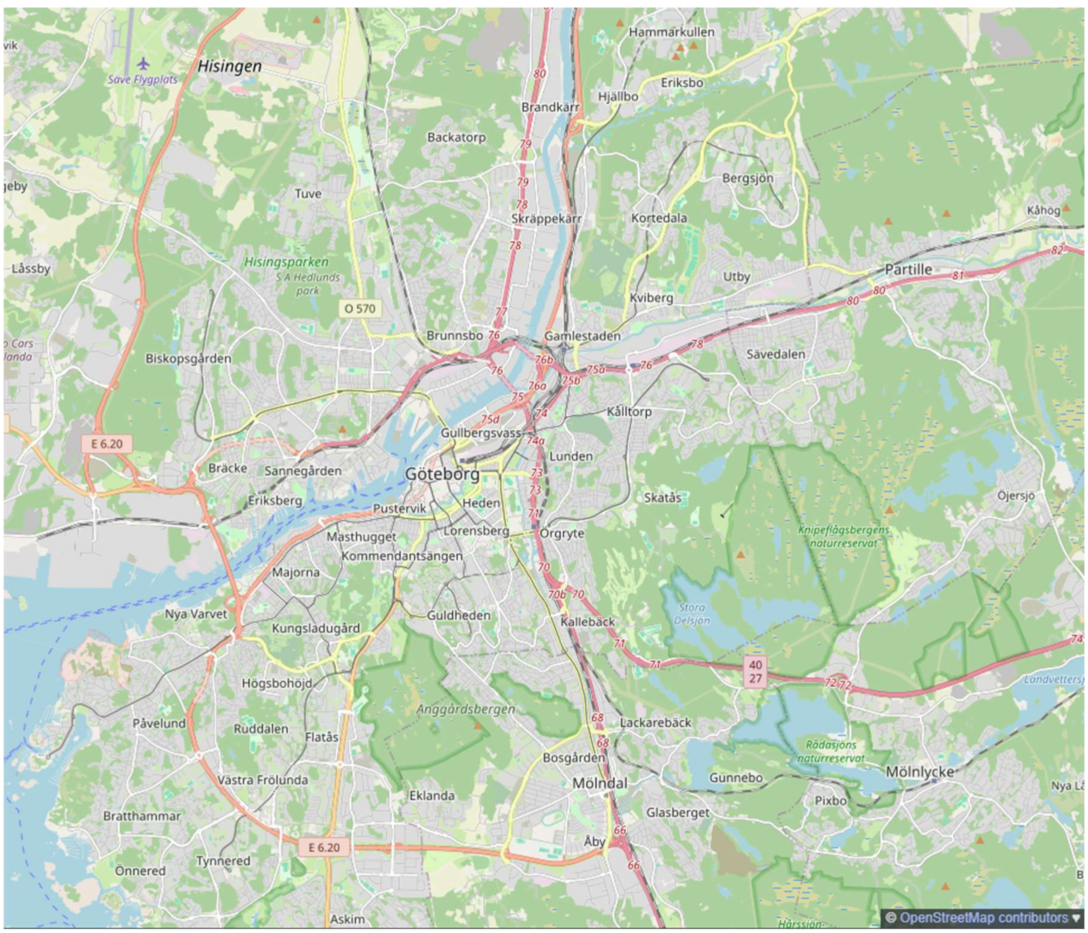

Figure 3: An overview map of Metropolitan Gothenburg, the research area in this project work, (available under the Open Database License (OpenStreetMap Foundation, 2025))

The locating process of roundabouts suitable for data collection was carried out by researching the Gothenburg area in OpenStreetMap (OpenStreetMap Foundation, 2025), in Google Street View (Google, 2025) as well as in-field observation by foot. The following locations were the ones chosen, see Figure 4.

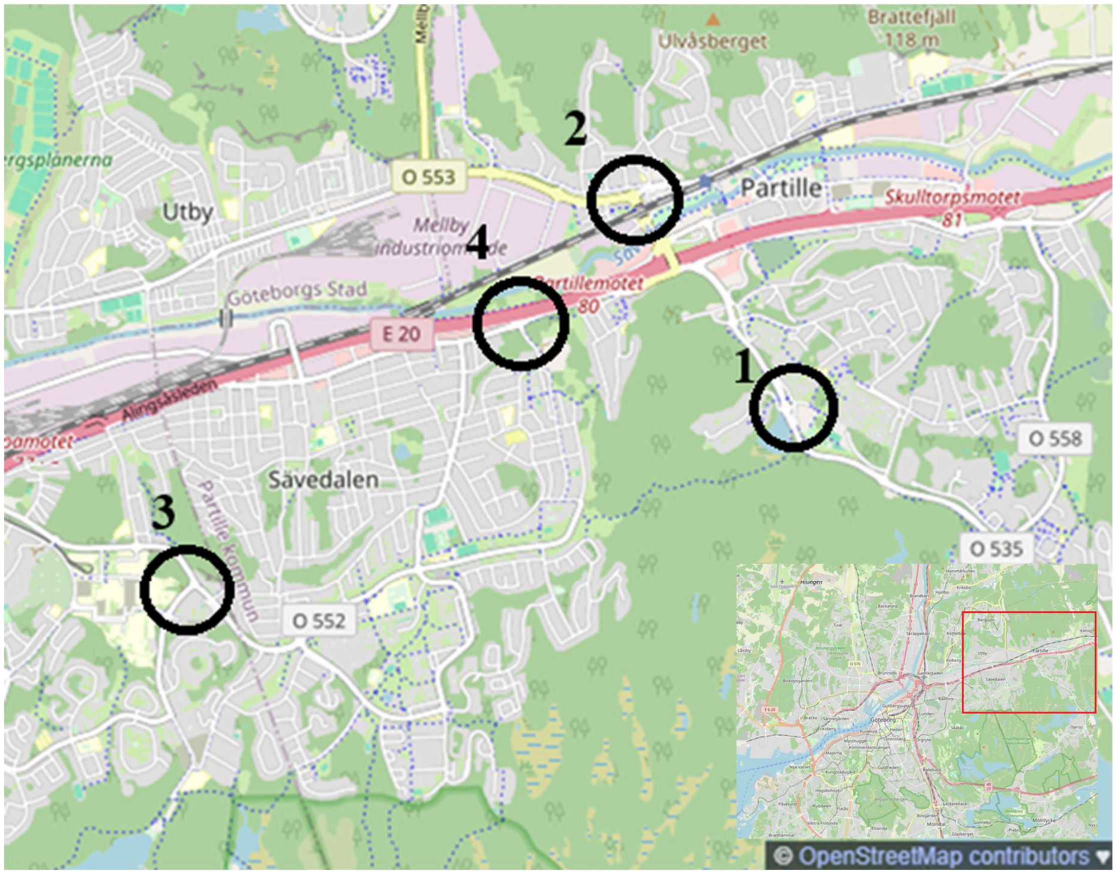

Figure 4: A map showcasing the location of the different roundabouts where the data were collected. A smaller map in the bottom right corner showing the areas location in the metropolitan Gothenburg area. (Available under the Open Database License, modifications done by the author (OpenStreetMap Foundation, 2025))

The data were collected at four different locations, all located in the eastern part of metropolitan Gothenburg. These locations all had an environmental surrounding able to provide a camera angle which could capture the entire roundabout in a single frame. Three of the data collection locations, number 1, 2 and 4, are situated in Partille Municipality and the remaining one, number 3, directly on the other side of the border in the neighbouring Gothenburg Municipality.

Table 1: A table presenting the location/address of the data collection spots

Location Number	Recording time	Location / Address
1	1h	Landvettervägen / Björndammsterassen/ Krondammsvägen
2	6h	Kung Göstas Väg / Utbyvägen / Lexbyvägen
3	1h	Östra Torpavägen / Smörslottsgatan
4	2h	Göteborgsvägen / Stora Ringvägen / Tillfällavägen

3.1.1 Location 1

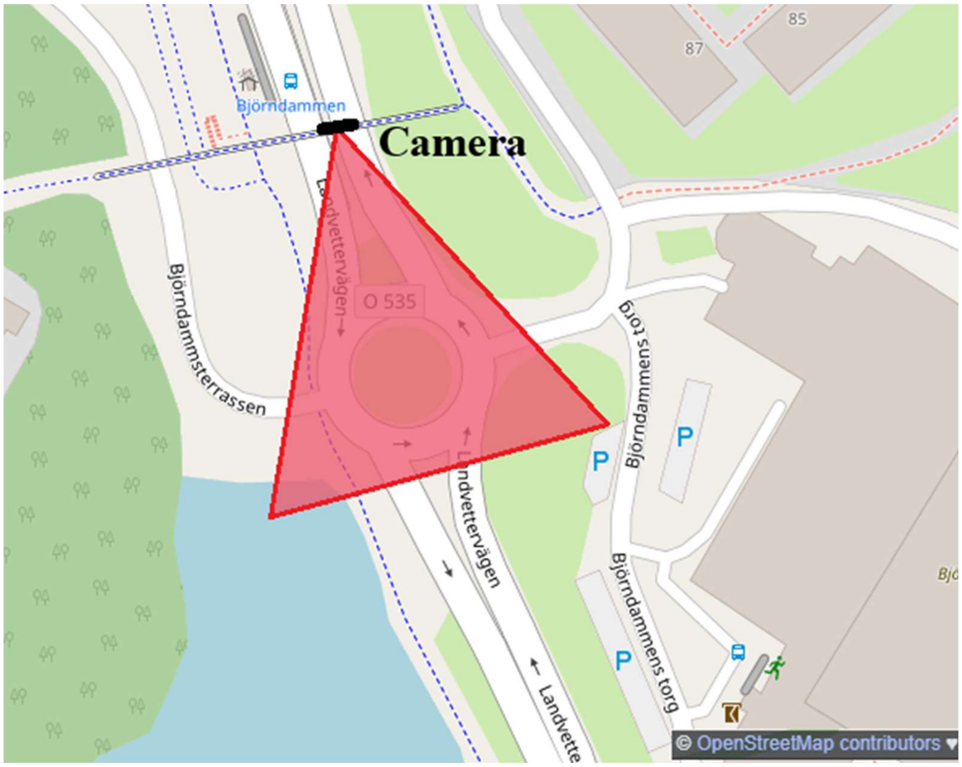

Figure 5: A detailed map of the first location used in the data collection. The black line represents the location of the camera and the redshaded triangle depicts the angle recorded by the camera. (Available under the Open Database License, modifications done by the author (OpenStreetMap Foundation, 2025))

The first location that was used in the data collection was the roundabout: Landvettervägen/ Björndammsterassen/ Krondammsvägen. This location was used in the very initial part of the project work and served mostly as an introduction to the methodology which was going to be used moving further. The site consisted of a four-legged single-lane roundabout where Landvettervägen is intersected by the two smaller residential roads Björndamssterassen and Krondammsvägen. Landvettervägen serves as the primarily route for commuters between the two suburban areas Landvetter and Partille in the eastern outskirt of the Metropolitan Gothenburg area. The nearby pedestrian overpass bridge was the main contributing factor for the decision to use this location, and it provided the possibility to capture the entirety of the roundabout into one camera shot.

Figure 6: An overview photo of the camera angle at the first data collection location: Landvettervägen/ Björndammsterassen/ Krondammsvägen

3.1.2 Location 2

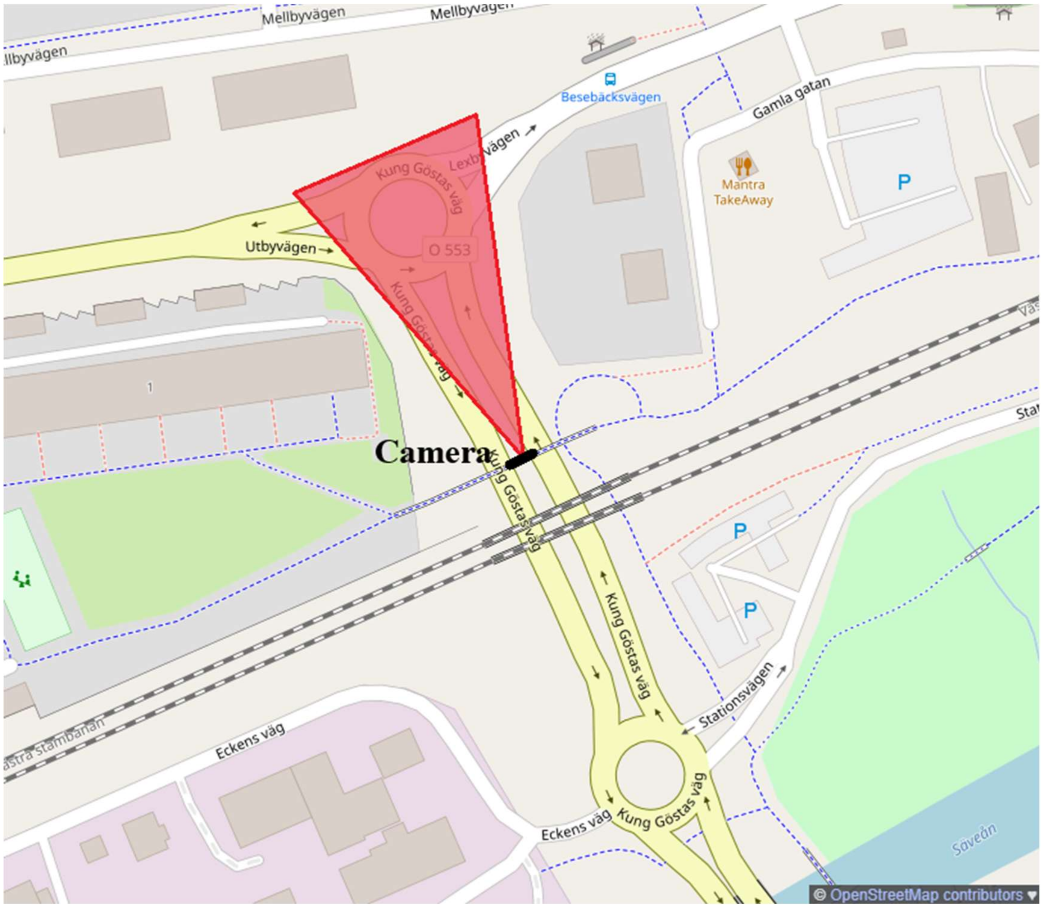

Figure 7: A detailed map of the second location used in the data collection. The black line represents the location of the camera and the redshaded triangle depicts the angle recorded by the camera. (Available under the Open Database License, modifications done by the author (OpenStreetMap Foundation, 2025))

The second data collection site were located at the roundabout: Kung Göstas Väg/ Utbyvägen/ Lexbyvägen and consisted of a three-legged single-lane roundabout. Kung Göstas Väg serves as the main passage for vehicles traveling between the north- and southside of the railway line Västra Stambanan which creates a barrier effect across Partille. Together with the direct proximity of the highway E20 this creates a continuous traffic flow at the site during rush hours. Utbyvägen serves as a connection for the suburban areas Utby and Bergsjön to both Partille center as well as linking it to Gothenburg center with the highway E20. Lexbyvägen provides a similar function, but for the considerably smaller population of the Lexby residential neighborhood. The heavy traffic flow between Kung Göstas Väg and Utbyvägen combied with the right of way for the circulating traffic and the yield-in rules of the roundabout causes frequent conflict and waiting for the traffic wanting to enter from Lexbyvägen. This process is very favorable for the purpose of gap acceptance data collection, since it produces a large quantity of continuous gaps for the drivers to determine whether they should accept or reject to enter the roundabout.

Along with the train line Västra stambanan there is a cycling and pedestrian bridge which offers a location with a high enough elevation to be able to capture the entire roundabout into one camera angle, although it is quite narrow to fit both the west and the east leg.

Figure 8: An overview photo of the camera angle at the second data collection location: Kung Göstas Väg/ Lexbyvägen/ Utbyvägen

3.1.3 Location 3

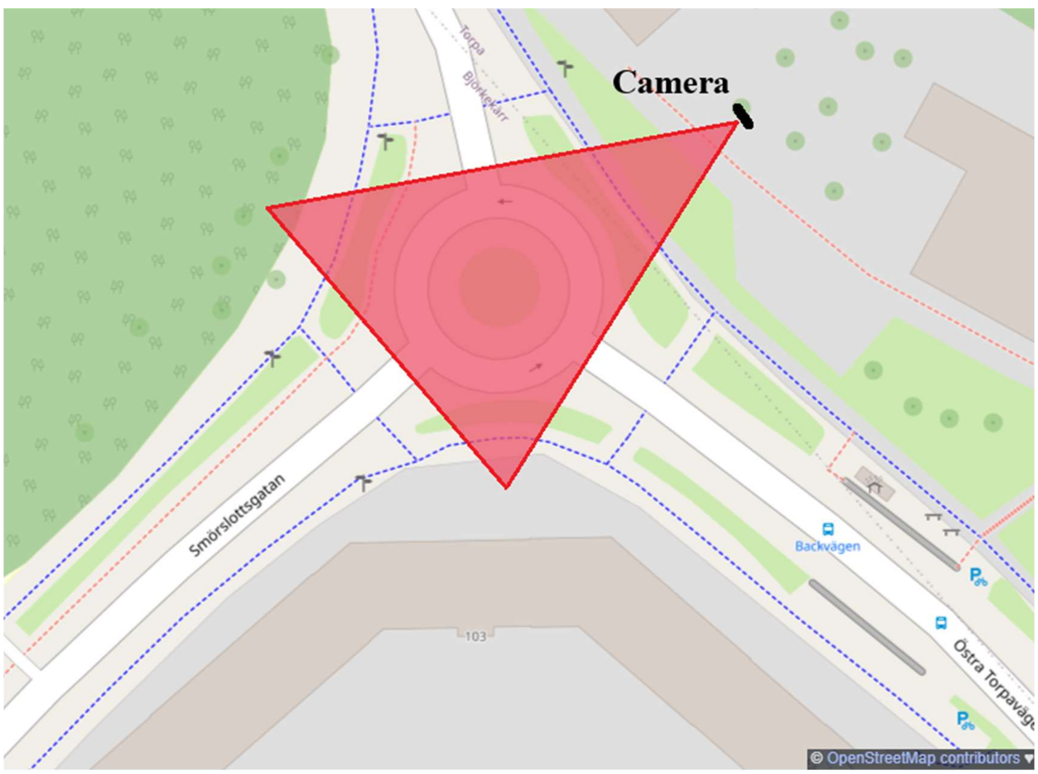

Figure 9: A detailed map of the third location used in the data collection. The black line represents the location of the camera and the redshaded triangle depicts the angle recorded by the camera. (Available under the Open Database License, modifications done by the author (OpenStreetMap Foundation, 2025))

The third data collection site were located at the roundabout: Östra Torpavägen/ Smörslottsgatan which is a three-legged single-lane roundabout. Östra Torpavägen provides a connection for the suburban residential areas Vallhamra and Sävedalen with the highway E20 and further towards central Gothenburg. Smörslottsgatan serves a similar function by connecting the residential areas of Björkekärr and Härlanda to Östra Torpavägen and further on as previously described. The roundabout experiences a quite even distributed flow of traffic between the north leg and the southwest leg as well as between the north leg and the southeast leg. The traffic flow between the southwest and the southeast leg is, however, considerably smaller since both the roads connect to residential areas. The terrain directly east of the roundabout consists of a steep slope which provides a site sufficiently elevated to enable the camera to capture the complete roundabout in a singular angle.

Figure 10: An overview photo of the camera angle at the third data collection location: Östra Torpavägen/ Smörslottsgatan

3.1.4 Location 4

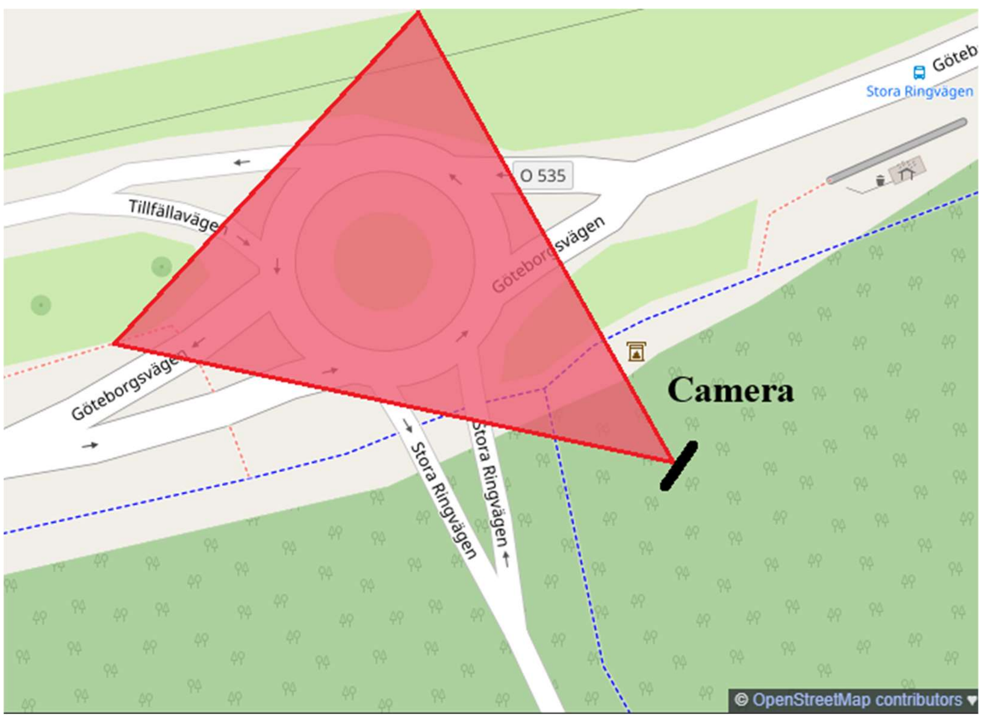

Figure 11: A detailed map of the fourth location used in the data collection. The black line represents the location of the camera and the redshaded triangle depicts the angle recorded by the camera. (Available under the Open Database License, modifications done by the author (OpenStreetMap Foundation, 2025))

The fourth and final data collection site were located at the roundabout: Göteborgsvägen/ Tillfällavägen/ Stora Ringvägen which is a four-legged single-lane roundabout. Göteborgsvägen is the main road in the central parts of Sävedalen and connects the area to the nearby highway E20. Tillfällavägen serves a similar purpose but works as a small version of a ringroad avoiding the most central and congested parts. Stora Ringvägen functions as a link for the residential areas of Vallhamra and Puketorp to the intersection and further on to Göteborgsvägen. A small slope directly adjacent southeast of the roundabout offered an elevated overview of the site and made it possible for the camera to capture the full roundabout in one angle.

Figure 12: An overview photo of the camera angle at the fourth data collection location: Göteborgsvägen/ Tillfällavägen/ Stora Ringvägen

3.1.5 Camera equipment

Camera specifics:

- **Camera model name:** Sony Alpha A7 IV
- **Camera objective:** 28-70mm F3.5-5.6 OSS
- **Camera stand name:** Manfrotto Element MII

The video recording was carried out with the system camera Sony Alpha A7 IV which was able to fulfill the quality requirements by GoodVision with a large margin. The steadiness and stability of the footage was a key parameter when extracting the data to ensure that the vehicle trajectories matched up, both with each other and with the traffic infrastructure in the recording to be able to draw the lines in GoodVision as accurately as possible.

Figure 13: Overview picture of the camera equipment used during the data collection

3.2 GoodVision

For the process of extracting the data collected in the video-recorded traffic footage the AI-based video analytics software GoodVision was selected. An overviewing description of GoodVision has been presented in previous parts of the report, see Section 2.3 AI-based video analytics software. The program offers a broad selection of functions for video analytics. The methodology in this project work will be using the *GoodVision Video Insights Platform* and primarily the functionalities regarding the vehicle trajectory and vehicle classification.

3.2.1 Video uploading and input data

When uploading the desired video recording into the *GoodVision Video Insights Platform* the format and quality of the footage needs to be validated before the process is initiated. After a successful uploading of the desired video files, the software also requires input in terms of geolocation, date and time of recording and if the footage is recorded with a drone or with a fixed camera. The user of the software also has the possibility to select what classes in the vehicle classification they want to include in the processing. This project work has been using the standardized vehicle classification in GoodVision which consists of the following categories: Cars, buses, motorcycles, vans, light trucks and heavy trucks. The standardized vehicle classification also includes bicycles and pedestrians, but these have both been excluded from this project work since it focuses solely on the vehicles travelling on the roadway and not the surrounding pavement and cycleways.

The option is also given to specify where the center point of the trajectory should be positioned in terms of height of the vehicle. The options range between the halfway point, 1/4 height, 1/8 height and all the way down to the bottom point of each vehicle, see Figure 14. This selection shows some differences when displaying trajectory between vehicle classes that have a significant difference in height, such as cars and heavy trucks. If the camera angle is proportionally low it would be preferable to use the bottom point as the center for the vehicle trajectory, as this could help clarify in what lane each vehicle has been travelling. In the data collection for this project work the camera angle has, however, been sufficient enough to instead use the center point of the car as the center point of the trajectory. This selection has the advantage of making the trajectory match better with the road network as well as making it easier to track background vehicles with vehicles in the foreground blocking the camera view.

3.2.2 Vehicle trajectory

After finishing the selection of required and optional inputs, the software processes the uploaded video and presents the result with an overlay of trajectories in different colours over a still image, see Figure 15. Each line represents one individual vehicle, and each colour categorizes them in the different vehicle classes. The colour scheme represents the following vehicle classes:

- Light Green: Car
- Dark Blue: Bus
- Light Blue Motorcycle
- Turquoise: Van
- Beige: Truck
- Brown: Heavy Truck

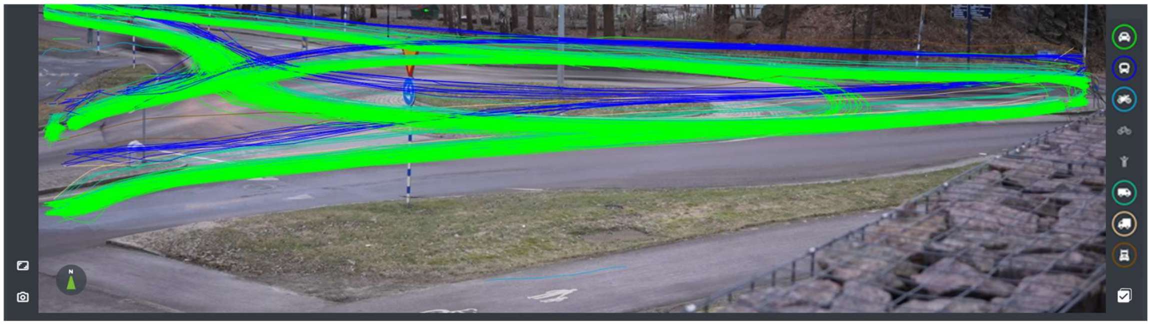

Figure 14: An example of the resulting vehicle trajectories after the video has been processed with GoodVision. The right-side legend depicts the different vehicle classes and their respective color.

3.2.3 Traffic flow data and OD-matrices

Once the vehicle trajectory is in place, it is possible to describe the traffic scene to enable multiple additional features. In this project work, where all traffic scenes consist of roundabouts, the scene description have been consisting of defining all entries and exits to the inner traffic circle. By defining these lines, the software produces vehicle lists where each vehicle gets an ID, a vehicle class and a timestamp on when they crossed this line. This makes it possible to extract multiple different statistics and metrics about the traffic conditions.

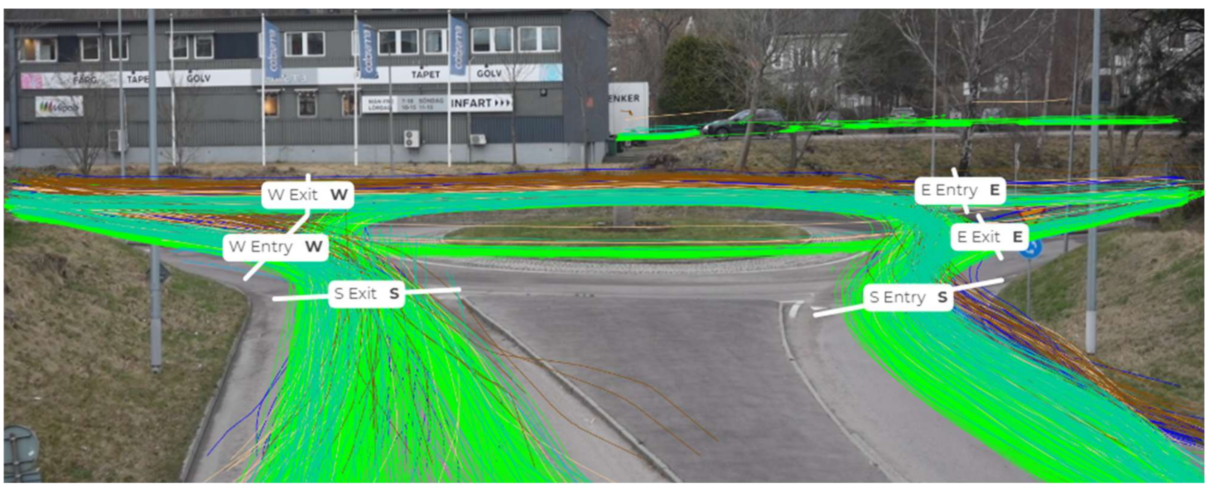

Figure 15: An example of how a traffic scene can be described with the help of the line function in GoodVision

Table 2: An example of a vehicle list produced by GoodVision including an individual ID, vehicle class, time gap between the passing of different vehicles and an exact timestamp for each vehicle crossing q defined line in the traffic scene

ID	CLASS	TIME GAP (s)	ENTRY TIME
LMv7912nbMWRt9T4f2EMkj	CAR	0	2025-03-27 14:00:08.789
9od8NahfLdhhvxLrM8mvX8	CAR	2.745	2025-03-27 14:00:11.534
HouVHDBioNyutgL9YRRYt3	TRUCK	36.279	2025-03-27 14:00:47.813
DL6hi95a3zfyBZXbpoeMkR	CAR	2.23	2025-03-27 14:00:50.043
GvfqG8qqoxTipYsJFwYSq6	CAR	38.888	2025-03-27 14:01:28.931
4aYLDuW7Vrx2R4YUQLnnBW	CAR	11.622	2025-03-27 14:01:40.553

For the upcoming traffic simulation, it is of great interest to know the traffic flow which enters each leg of the roundabout and where it exits. Origin-Destination (OD) matrices are therefore created by matching each ID of vehicles crossing one of the entry lines with an identical ID passing any of the corresponding exit lines in the roundabout. Combined with the vehicle classification which is connected to every ID it is also possible to produce OD-matrices for each separate vehicle class.

Table 3: Corresponding O-D matrices to the traffic scene depicted previously (see Figure 15) for both total traffic volume and for cars separately.

Origin (column) - Destination (row) matrix - TOTAL			
ORIGIN/DESTINATION	**EXIT E**	**EXIT S**	**EXIT W**
ENTRY E	---	222	67
ENTRY S	307	---	982
ENTRY W	45	1278	---

Origin (column) - Destination (row) matrix - CAR			
ORIGIN/DESTINATION	**EXIT E**	**EXIT S**	**EXIT W**
ENTRY E	---	198	60
ENTRY S	276	---	811
ENTRY W	41	1106	---

3.2.4 Gap acceptance data extraction

The method for extracting gap acceptance data utilizes the same line-function that was used for gathering data about traffic flow, but with a different layout when describing the traffic scene. The layout consists of three different lines, Major, Minor Wait and Minor In. The Major line defines where the major traffic flow, which has the right of way, passes the merging point with the minor flow. This line is placed at the point where the gap to enter the major flow is considered to be closed and it is no longer possible for a vehicle from the minor flow to enter the major flow. In this project work, where every traffic scene is a roundabout, the Major line is placed from the central island to the splitter island crossing the circulatory roadway.

Figure 16: An example of the described scene for gap acceptance data extraction with the Major line highlighted and the incorporated vehicle trajectories.

The Minor In line is placed in line with the yield-in lines in the existing infrastructure. In case of absence of these lines or worn-down conditions of these, the video is examined manually to determine where the vehicles can be considered to have entered into the roundabout. The Minor Wait line is placed on a distance from the Minor In line where vehicles crossing are assumed to have been starting to wait for an acceptable gap to enter. This distance is approximately one vehicle length but differs depending on what vehicle class is waiting so a bit of buffer distance is commonly included. A vehicle movement filter is applied in GoodVision between the Minor Wait and Minor In line to guarantee that only vehicles travelling in the designated direction are included.

Figure 17: An example of the described scene for the gap acceptance data extraction with the Minor In line and Minor Wait line with the incorporated vehicle trajectories.

Once the traffic scene is described, vehicle reports for every line, consisting of the same parameters as the vehicle lists used in the traffic flow data, see Table 4, is produced in GoodVision and gives the essential data required to calculate accepted and rejected gaps. The separate vehicle reports for each line is combined into one Excel file where the timestamps are matched against each other.

Table 4: The combined vehicle list of both the Major line, Minor Wait line and Minor In line used to calculate accepted and rejected gaps.

CLASS	ENTRY TIME (Major)	CLASS	ENTRY TIME (Minor waiting)	EXIT TIME (Minor in)
CAR	2025-04-01 15:00:05.582	**CAR**	2025-04-01 15:00:13.607	2025-04-01 15:00:15.188
CAR	2025-04-01 15:00:21.975	**CAR**	2025-04-01 15:00:16.788	2025-04-01 15:00:18.347
CAR	2025-04-01 15:00:23.950	**CAR**	2025-04-01 15:00:18.880	2025-04-01 15:00:20.191
CAR	2025-04-01 15:00:27.644	**CAR**	2025-04-01 15:00:34.416	2025-04-01 15:00:36.333
CAR	2025-04-01 15:00:29.925	**CAR**	2025-04-01 15:00:37.259	2025-04-01 15:00:39.062
CAR	2025-04-01 15:00:32.889	**CAR**	2025-04-01 15:00:50.870	2025-04-01 15:00:51.771
VAN	2025-04-01 15:01:23.393	**BUS**	2025-04-01 15:01:35.275	2025-04-01 15:01:38.184
CAR	2025-04-01 15:01:32.327	**VAN**	2025-04-01 15:01:40.416	2025-04-01 15:01:42.561
CAR	2025-04-01 15:02:24.241	**CAR**	2025-04-01 15:01:42.972	2025-04-01 15:01:44.687
CAR	2025-04-01 15:02:27.104	**CAR**	2025-04-01 15:01:45.563	2025-04-01 15:01:47.283
CAR	2025-04-01 15:02:42.600	**CAR**	2025-04-01 15:01:47.797	2025-04-01 15:01:49.381
CAR	2025-04-01 15:03:01.921	**CAR**	2025-04-01 15:02:02.820	2025-04-01 15:02:03.921
CAR	2025-04-01 15:03:09.194	**VAN**	2025-04-01 15:02:23.715	2025-04-01 15:02:30.991
CAR	2025-04-01 15:03:36.184	**CAR**	2025-04-01 15:02:31.678	2025-04-01 15:02:33.597
CAR	2025-04-01 15:03:52.571	**CAR**	2025-04-01 15:02:34.297	2025-04-01 15:02:36.032

With a simple subtraction between the vehicle registrations at the Major line, the gaps between the vehicles can be calculated, these values are the same as the gaps offered for the vehicles waiting to enter from the Minor flow, see Table 5.

Table 5: The calculated gap between the vehicles in the Major flow, which is equal to the gaps offered for the vehicles in the Minor flow to either accept or reject

MAJOR: ENTRY TIME	MAJOR: TIME GAP BEFORE VEHICLE	MAJOR: TIME GAP AFTER VEHICLE
15:00:05.582	00:00:00.000	00:00:16.393
15:00:21.975	00:00:16.393	00:00:01.975
15:00:23.950	00:00:01.975	00:00:03.694
15:00:27.644	00:00:03.694	00:00:02.281
15:00:29.925	00:00:02.281	00:00:02.964
15:00:32.889	00:00:02.964	00:00:50.504
15:01:23.393	00:00:50.504	00:00:08.934
15:01:32.327	00:00:08.934	00:00:51.914
15:02:24.241	00:00:51.914	00:00:02.863
15:02:27.104	00:00:02.863	00:00:15.496
15:02:42.600	00:00:15.496	00:00:19.321
15:03:01.921	00:00:19.321	00:00:07.273

By matching the timespan between which a vehicle first crosses the Minor Wait line and then subsequently the Minor In line, with what gaps were offered at the Major flow during the same time frame, it is possible to determine what gaps were accepted and what gaps were rejected. Gap acceptance events were determined by matching timestamps between major and minor streams. Rejected and accepted gaps were classified algorithmically using timestamp ranges.

To reduce the manual work and make the matching process more effective the COUNTIF function in Excel is used to search the list of offered gaps between the timestamp at the Minor Wait line and the timestamp at the Minor In line. The function then recognizes and counts the timestamps at the Major line which is within the range of the lower and upper limit and presents the number as: number of vehicles that have been rejected. If the value presented is 0, no conflict has been found between minor flow and major flow vehicles. If the value presented is 1, the vehicle in the minor stream flow has waited for one vehicle to pass before entering and the gap to the next car in the major stream flow is therefore the accepted gap. If the value presented is 2 or greater, the vehicle from the minor stream flow has waited for multiple cars in the major flow to pass and every gap between the cars in the major flow has been rejected, except for the last one which has been accepted.

For each minor vehicle, the number of rejected major vehicles determined the count of rejected gaps, with the subsequent accepted gap being recorded. So, 2 number of vehicles, rejected give one accepted gap and one rejected. 3 number of vehicles rejected, gives one accepted gap and two rejected and so forth. By using the IF-function in Excel with the condition >0 for the values of number of vehicles rejected together with the complementing timestamps, all gaps accepted and rejected during one timeframe can then be presented in one table, see Table 6.

Table 6: The calculation of accepted and rejected gaps by comparing the timestamps from Minor Wait and Minor In to the gaps offered at the Major line

CLASS	MINOR STREAM VEHICLE APPROACHING	MINOR VEHICLE ENTERING MAJOR	NUMBER OF VEHICLES REJECTED	LAST REJECTED VEHICLE TIMESTAMP	MAXIMUM REJECTED GAP (s)	AVERAGE REJECTED GAP (s)	ACCEPTED GAP (s)
VAN	15:02:23.715	15:02:30.991	2	15:02:27.104	2.863	2.863	15.496
CAR	15:12:29.433	15:12:33.284	1	15:12:29.967			7.563
CAR	15:17:25.154	15:17:34.189	3	15:17:30.560	2.237	2.081	16.849
VAN	15:17:53.355	15:17:59.178	1	15:17:55.609			49.616
CAR	15:19:58.392	15:20:02.839	1	15:19:58.684			10.043
CAR	15:21:30.483	15:21:35.253	1	15:21:31.185			33.304
CAR	15:25:15.876	15:25:22.526	1	15:25:17.866			12.128
CAR	15:25:53.718	15:26:10.501	5	15:26:06.034	3.719	2.747	7.553
CAR	15:27:16.684	15:27:21.746	1	15:27:18.523			11.975
CAR	15:28:02.657	15:28:06.878	1	15:28:03.742			8.902
CAR	15:38:33.640	15:38:38.128	1	15:38:34.376			7.682
CAR	15:40:16.940	15:40:26.604	2	15:40:22.393	3.777	3.777	22.457
CAR	15:48:24.855	15:48:30.115	1	15:48:24.936			6.153
CAR	15:56:45.561	15:56:57.057	3	15:56:51.480	2.566	2.510	7.155
CAR	15:57:17.810	15:57:21.419	1	15:57:18.011			4.601

3.3 Critical gap calculation

After data extraction in GoodVision and further processing of the data in Excel, the dataset is presented in a binary list, consisting of values either accepted or rejected. Which method to implement for calculating critical gap from datasets of accepted and rejected values to get the most accurate result, is a thoroughly analyzed and debated topic as previously discussed, see Chapter 2.4 Gap acceptance theory. In this project work, the method chosen is Raff's method.

Raff's method is a well proven method for the calculation of critical gap and is often favored for its simplicity and straightforward implementation. Raff's method is based on a cumulative distribution of the accepted and rejected gaps, and it proposes that the intersection point between the two graphs is the value for the critical gap. The method assumes that the critical gap should be at the point where the cumulative probability of accepted gaps $F_a(t)$ shorter than the critical gap, is equal to the cumulative probability of rejected gaps, $F_r(t)$ longer than the critical gap. The method can be mathematically expressed as follows, see Equation (1):

$$1 - F_r(t) = F_a(t) \qquad (1)$$

Raff's method presents a clear visual representation of critical gap calculation, and the pattern of the intersected cumulative graph lines have given Raff's method the additional name, the Threshold method. In Raff's method, all gaps, either rejected or accepted, are thought to be independent and no consideration is taken to what decision was made previously by the same driver. The leading motive for choosing Raff's method for this thesis is its simplicity and stability, even when working with relatively small datasets.

3.4 SIDRA Intersection

After performing the calculation for critical gap with the data from all the three different locations, the new critical gap values were implemented into SIDRA Intersection to compare the results from the analysis with the base values. The standardized critical gap value in SIDRA Intersection is 4.00 seconds in single-lane roundabouts. The analysis was performed by describing the site of each roundabout in the software. The site description included parameters such as island diameter, entry angle and entry radius, lane width and lane length among many other design features. The site input also includes details about traffic volumes and traffic movement, which was filled with local traffic volume data obtained by the O-D matrices constructed in GoodVision.

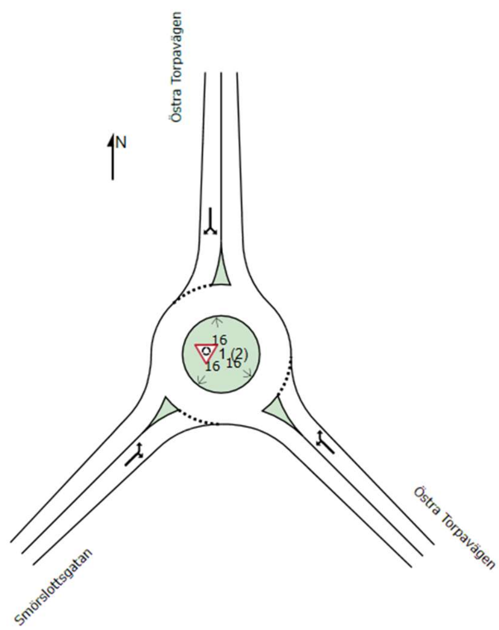

Figure 18: An example of one of the sites modelled in SIDRA Intersection

3.5 Classification algorithms in machine learning

Classification algorithms in machine learning are used to develop a better understanding of what parameters and variables are important when the decision of accepting or rejecting a gap is made. The algorithms use an existing dataset with different features and outcomes and tries to predict the outcome of additional unseen data based on the knowledge learned from the existing data. So, in the case of this thesis, the goal was to feed the machine learning algorithm with data about accepted and rejected gaps to enable the algorithm to predict if future gaps would be accepted or rejected based on multiple different parameters. To enable the model to understand and process the existing dataset, some adaptations were made with the help of the programming software Python. For the full code used, see Appendix. All time values for gaps, both accepted and rejected, were grouped into one combined column named "Offered gaps" and one additional column was added labeled "Decision" which included a binary system, where a "1" indicated that the gap was accepted and a "0" indicated that the gap was rejected. Other parameters that were also added to research the importance of these features were: at which time of the day the gap was registered, at which roundabout it was and on which day it was.

Table 7: Python modified dataset based on the data extracted from GoodVision

Time of the day	Roundabout	Class	Date	Minor stream vehicle approaching	Minor stream vehicle entering major stream	Number of vehicles in major rejected	Last rejected vehicle passage timestamp	Maximum rejected gap	Offered gap	Decision
Afternoon	A	CAR	2025-03-27	11:45.4	11:52.8	1	11:49.5	0	6.286	1
Afternoon	A	CAR	2025-03-27	12:08.7	12:18.9	2	12:14.1	2.561	2.561	0
Afternoon	A	CAR	2025-03-27	12:08.7	12:18.9	2	12:14.1	2.561	7.221	1
Afternoon	A	CAR	2025-03-27	22:24.2	22:31.5	2	22:27.9	2.907	2.907	0
Afternoon	A	CAR	2025-03-27	22:24.2	22:31.5	2	22:27.9	2.907	4.984	1
Afternoon	A	CAR	2025-03-27	22:39.7	22:47.5	2	22:44.1	2.504	2.504	0
Afternoon	A	CAR	2025-03-27	22:39.7	22:47.5	2	22:44.1	2.504	3.495	1
Afternoon	A	CAR	2025-03-27	23:54.2	23:59.9	2	23:56.3	1.775	1.775	0
Afternoon	A	CAR	2025-03-27	24:08.4	24:14.8	2	24:11.4	1.544	1.544	0
Afternoon	A	CAR	2025-03-27	26:45.1	26:48.0	1	26:45.4	0	3.652	1
Morning	B	HEAVY TRUCK	2025-03-27	16:30.5	16:56.4	7	16:45.2	3.479	2.544	0
Morning	B	CAR	2025-03-27	17:02.7	17:10.1	2	17:08.7	4.699	4.699	0
Morning	B	CAR	2025-03-27	17:02.7	17:10.1	2	17:08.7	4.699	4.544	1
Morning	B	CAR	2025-03-27	17:11.4	17:21.4	3	17:20.5	4.456	2.796	0

3.5.1 Random forest

The first classification algorithm used was Random forest. Random forest algorithm is based on the concept of decision trees. Decision trees are models that follow a flowchart-like structure. The model includes a starting node, also called root node, and reaching out from the node is different branches with featured values. A data point follows the model from the root node through the branch that matches the criteria in the node down to the bottom node where the model produces the prediction to classification. Random forest is known as an ensemble machine learning method because it combines a multitude of decision trees into one “forest” (Ho, 1995). The final classification of the random forest is then classification result that is chosen by the greatest number of trees. The Random forest algorithm was chosen for its capability of handling both numeric and categorical data and because of its high robustness against noisy data and outliers.

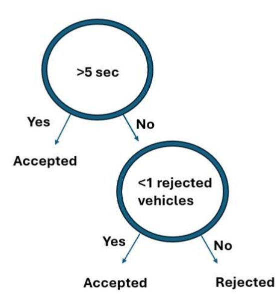

Figure 19: A basic model of a decision tree

3.5.2 K-Nearest neighbors

The K-nearest neighbor algorithm is a straightforward and easy to implement algorithm. By plotting existing data points on a graph with featured parameters on the x- and y-axis the model classifies new data points depending on what is the class of the nearest neighbors. As an example for this thesis, accepted and rejected gaps could be plotted on a graph with the time value of "offered gaps" on the x-axis and "number of previous vehicles rejected" on the y-axis. The data points would then be spread out with a certain pattern, and the algorithm would place new data points among these existing ones. The algorithm would then predict if the new value were accepted or rejected based on the nearest neighbor, if the algorithm checks the 5 nearest neighbors and 3 are accepted and 2 rejected, the prediction would be that new value is accepted. The number of neighbors that the algorithm considers when predicting is decided by the user, thereby the K-value in the name. It is of great importance to choose an optimal K-value to ensure accuracy of the model since a too large or too small could skew the results in opposite directions. The KNN algorithm was primarily chosen because of its high interpretability, meaning that the model is easily understood and the predictions made by the model is explainable.

3.5.3 Support vector machine

The Support vector machine algorithm is a commonly used algorithm within machine learning classification tasks. The model draws a line, or a hyperplane in a multidimensional case, through the dataset that aims to maximize the distance between the different data points. The aim of implementing this line or hyperplane is then, very straightforward, to divide the data set into different classifications based on the chosen parameters. The support vector machine algorithm was used because the algorithm is effective when the dataset includes multiple dimensions, such as numerical data and text classification data. Additionally, the support vector machine algorithm also has the ability to separate non-linear boundaries by implementing curved lines or by adding additional dimensions which can be very powerful in certain datasets.

3.5.4 Naïve bayes classifier

Naïve bayes classifier is a probabilistic classification algorithm which is based upon Bayes theorem. Bayes theorem consists of two parameters, C which represents the class and X which represents the features. So, $P(C \mid X)$ indicates the probability of the class given the certain features. $P(X \mid C)$ represents the probability of the feature dependent on the class. Combined with the probability of the class $P(C)$ and the probability of the feature $P(X)$, Bayes theorem is defined mathematically as follows, See equation (2).

$$P(C \mid X) = \frac{P(X \mid C)P(C)}{P(X)} \tag{2}$$

In Naïve bayes classifier, Bayes theorem is implemented by assuming that the features are independent given the class, meaning that the classification from one parameter is unrelated to the other parameters. This quite unrealistic or "naïve" assumption is what have given the model its name. The reasoning for implementing the Naïve bayes classifier algorithm is because of its simplicity and character as a fast baseline model. The naïve assumptions of feature independence in the model makes the model very quick and high performing in circumstances when the datasets are small.

4 Results

This chapter presents the outcomes of data collection, gap extraction using GoodVision, critical gap estimation with Raff's method, SIDRA Intersection comparisons, and classification model performance.

4.1 Data collection and extraction

4.1.1 Location 1

The first location of the data collection at the roundabout: Landvettervägen/ Björndammsterassen/ Krondammsvägen, had a very low success rate in terms of gathering data points. The location was chosen very early during the project work because of its favorable position in terms of providing a good camera angle and the large traffic flow on Landvettervägen. The remaining two legs of the roundabout, Björndammsterassen and Krondammsvägen, provided too small traffic flows and the conflicts at the roundabout were very low, leading to an insufficient number of accepted and rejected gaps at this location. Concerns were raised about this already while observing the roundabout during the video recording and it was also later confirmed when attempting to extract data in GoodVision from this location. The location was therefore excluded from further research during this project work and the few data points that were collected were deemed irrelevant for the thesis. The time spent both video recording and working on GoodVision was, however, not completely pointless. Since this was the very first location used, it served as a good learning experience to understand both the camera recording and to gather valuable knowledge and practice on how to use GoodVision most effectively.

4.1.2 Location 2

The second location was the roundabout: Kung Göstas Väg/ Utbyvägen/ Lexbyvägen. The data collection at this location produced data points at a much higher rate than the previous location. The roundabout had a large traffic flow between the south leg and the west leg which created a constant interruption for the traffic entering the roundabout from the east leg. This provided a substantial number of gaps for the incoming traffic from the east to accept or reject, producing plenty of data points. This location was the one that generated the highest number of data points per hour, and it was thus the primarily used location throughout the whole data collection and data extraction process. Of the 10 total hours of video recording, 6 hours were recorded at this roundabout. In total it yielded 840 data points, which consisted of 251 accepted gaps and 589 rejected gaps. The proportion of the gaps which were accepted made up 30% of the total number of gaps. For more detailed information on all data points gathered, see Table 8. In the table, all data points are categorized based on the time value as well as if they were rejected or accepted. The table also provides columns that depict how large the proportion of values were in the certain time range compared to the total number of data points. The table features also the cumulative probability of each time range, which was used further when calculating the critical gap with Raff's method.

Table 8: The gap acceptance data collected at the roundabout: Kung Göstas Väg/ Utbyvägen/ Lexbyvägen

Critical gap data: Kung Göstas Väg							
Time gap (s)	**Accepted gaps**	**Rejected gaps**	**Probability of accepted**	**Probability of accpeted**	**Cumulative probability accepted**	**Cumulative probability rejected**	**Cumulative probability rejected reversed**
0-1	0	0	0%	100%	0%	0%	100%
1-1.5	0	12	0%	100%	0%	2%	98%
1.5-2	0	120	0%	100%	0%	22%	78%
2-2.5	1	172	1%	99%	0%	52%	48%
2.5-3	2	124	2%	98%	1%	73%	27%
3-3.5	17	55	24%	76%	8%	82%	18%
3.5-4	13	41	24%	76%	13%	89%	11%
4-4.5	33	22	60%	40%	26%	93%	7%
4.5-5	35	16	69%	31%	40%	95%	5%
5-5.5	23	13	64%	36%	49%	98%	2%
5.5-6	21	6	78%	22%	58%	99%	1%
6-6.5	35	4	90%	10%	72%	99%	1%
6.5-7	17	2	89%	11%	78%	100%	0%
7-7.5	28	1	97%	3%	90%	100%	0%
7.5-8	13	0	100%	0%	95%	100%	0%
8-9	13	1	93%	7%	100%	100%	0%
0-9	251	589	30%	70%			

4.1.3 Location 3

The third location, the roundabout: Östra Torpavägen/ Smörslottsgatan, saw a slight decrease in terms of producing data points compared to Location 2, although the performance was very similar. The roundabout had a more even distribution in terms of traffic flow between the three legs of the roundabout, which also lead to a more even distribution of accepted and rejected gaps between the three legs of the roundabout. The total number of collected data points at Location 3 was 232 gaps, comprised of 82 accepted gaps and 150 rejected gaps. The proportion of the gaps which were accepted was 35%, which was slightly higher than at Location 2

Table 9: The gap acceptance data collected at the roundabout: Östra Torpavägen/ Smörslottsgatan

Critical gap data: Östra Torpavägen							
Time gap	**Accepted gaps**	**Rejected gaps**	**Probability acceppted**	**Probability rejected**	**Cumulative probability accepted**	**Cumulative probability rejected**	**Cumulative probability rejected reversed**
0-1	0	0	0%	100%	0%	0%	100%
1-1.5	0	0	0%	100%	0%	0%	100%
1.5-2	0	22	0%	100%	0%	15%	85%
2-2.5	2	47	4%	96%	2%	46%	54%
2.5-3	2	32	6%	94%	5%	67%	33%
3-3.5	1	19	5%	95%	6%	80%	20%
3.5-4	3	12	20%	80%	10%	88%	12%
4-4.5	4	8	33%	67%	15%	93%	7%
4.5-5	10	2	83%	17%	27%	95%	5%
5-5.5	9	5	64%	36%	38%	98%	2%
5.5-6	7	1	88%	13%	46%	99%	1%
6-6.5	4	0	100%	0%	51%	99%	1%
6.5-7	12	1	92%	8%	66%	99%	1%
7-7.5	9	1	90%	10%	77%	100%	0%
7.5-8	9	0	100%	0%	88%	100%	0%
8-9	10	0	100%	0%	100%	100%	0%
0-9	82	150	35%	65%			

4.1.4 Location 4

The fourth data collection location at the roundabout: Göteborgsvägen/ Tillfällavägen/ Stora Ringvägen, was the only roundabout that didn't consist of three legs after Location 1 was excluded. Although this roundabout included four legs, it produced data points at an almost identical rate as Location 3 and at a slightly lower rate compared to Location 2. The traffic flow in this roundabout was mainly incoming from the east leg and exiting with a quite even distribution among the remaining three legs, leading to most accepted and rejected gaps being registered at the north and the west entries. This location was the location that had the highest proportion of accepted gaps in relation to total number of gaps.

Table 10: The gap acceptance data collected at the roundabout; Göteborgsvägen/ Tillfällavägen/ Stora Ringvägen

Critical gap data: Göteborgsvägen							
Time gap	**Accepted gaps**	**Rejected gaps**	**Probability accepted**	**Probability rejected**	**Cumulative probability accepted**	**Cumulative probability rejected**	**Cumulative probability rejected reversed**
0-1	0	1	0%	100%	0%	1%	100%
1-1.5	0	10	0%	100%	0%	7%	93%
1.5-2	1	31	3%	97%	1%	26%	74%
2-2.5	2	35	5%	95%	3%	48%	52%
2.5-3	1	38	3%	97%	4%	72%	28%
3-3.5	1	17	6%	94%	5%	83%	17%
3.5-4	5	10	33%	67%	9%	89%	11%
4-4.5	8	7	53%	47%	17%	94%	6%
4.5-5	13	4	76%	24%	28%	96%	4%
5-5.5	12	2	86%	14%	39%	97%	3%
5.5-6	14	2	88%	13%	52%	99%	1%
6-6.5	8	2	80%	20%	60%	100%	0%
6.5-7	8	0	100%	0%	67%	100%	0%
7-7.5	6	0	100%	0%	72%	100%	0%
7.5-8	13	0	100%	0%	84%	100%	0%
8-9	17	0	100%	0%	100%	100%	0%
0-9	109	159	41%	59%			

4.1.5 Total data

The total data collection, including all the data from Location 2, 3 and 4 can be found summarized in the table below, see Table 11. The total data consisted of 1361 total gaps, 455 accepted gaps and 906 rejected gaps. The final distribution was nearly exactly double the number of rejected gaps compared to accepted gaps, or 33% accepted gaps to 67% rejected gaps.

Table 11: The total gap acceptance data collected at all roundabouts

Critical gap data: Total							
Time gap (s)	**Accepted gaps**	**Rejected gaps**	**Probability accepted**	**Probability rejected**	**Cumulative probability accepted**	**Cumulative probability rejected**	**Cumulative probability rejected reversed**
0-1	0	1	0%	100%	0%	0%	100%
1-1.5	0	22	0%	100%	0%	3%	97%
1.5-2	2	174	1%	99%	0%	22%	78%
2-2.5	6	255	2%	98%	2%	50%	50%
2.5-3	7	197	3%	97%	3%	72%	28%
3-3.5	18	92	16%	84%	7%	82%	18%
3.5-4	21	63	25%	75%	12%	89%	11%
4-4.5	46	37	55%	45%	22%	93%	7%
4.5-5	60	22	73%	27%	35%	95%	5%
5-5.5	46	22	68%	32%	45%	98%	2%
5.5-6	43	9	83%	17%	55%	99%	1%
6-6.5	48	6	89%	11%	65%	99%	1%
6.5-7	39	3	93%	7%	74%	100%	0%
7-7.5	43	2	96%	4%	83%	100%	0%
7.5-8	38	0	100%	0%	92%	100%	0%
8-9	38	1	97%	3%	100%	100%	0%
0-9	455	906	33%	67%			

In the following table the statistics for all the gaps that were offered are presented, see Table 12. The table shows that the majority of the gaps that were offered had a smaller value than the critical gap values that were calculated, with the mean value for offered gaps being 3.08s. Roughly two thirds of all offered gaps had a value below 4 seconds.

Table 12: The distribution of all offered gaps at all roundabouts in the data collection

Frequency distribution	Probability of occurrence	Number of occurrences
0-1 sec	14.6%	199
2-2.5 sec	19.1%	261
2.5-3 sec	15.0%	204
3-3.5 sec	8.1%	110
3.5-4 sec	6.2%	84
4-4.5 sec	6.1%	83
4.5-5 sec	6.0%	82
5-5.5 sec	5.0%	68
5.5-6 sec	3.8%	52
6-6.5 sec	4.0%	54
6.5-7 sec	3.1%	42
7-7.5 sec	3.3%	45
7.5-8 sec	2.8%	38
8-8.5 sec	2.4%	33
8.5 -9 sec	0.4%	6
Total	100.0%	1361

4.2 Critical gap calculations

The critical gap was calculated by using Raff's method combined with the data presented in the previous chapter, see Chapter 4.1. Raff's method utilizes the cumulative distribution of both the accepted gaps and the rejected gaps to find the value where a driver is equally likely to accept or reject the gap. By plotting the cumulative distributions in one graph where the cumulative probability of rejection is reversed, Raff stated that the intersection point between the lines is equal to the critical gap. The logic for reversing the cumulative probability of rejection is to have a 100% likelihood of rejection when the values are the lowest and for the percentage to decrease when the value for the time gap is increasing.

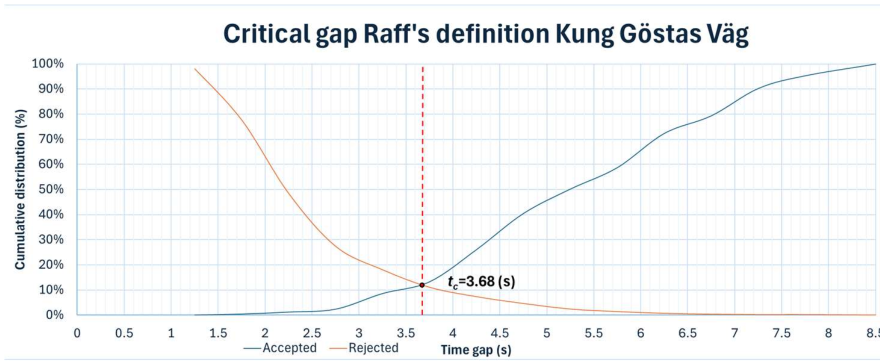

Figure 20: Critical gap calculated with Raff's definition for the gap acceptance data collected at the roundabout: Kung Göstas Väg/ Utbyvägen/ Lexbyvägen.

The critical gap calculation for the data collected at the roundabout: Kung Göstas Väg/ Utbyvägen/ Lexbyvägen, is presented in the graph above, see Figure 20. The lines in the graph are plotted from the data in the columns: "Cumulative probability accepted" and "Cumulative probability rejected reversed" in the table "Critical gap data: Kung Göstas Väg", see Table 8. The intersection point between the lines, and thereby also the critical gap, were determined to be at the time gap value: t_c = 3.68s.

The method was repeated for the same columns from the table for Östra Torpavägen and Göteborgsvägen, see Table 9 and Table 10. The results from these datasets are presented in the graphs below, see Figure 21 and Figure 22. The final calculated values for the critical gap were calculated for the roundabout at Östra Torpavägen to: t_c = 3.86s, and at Göteborgsvägen to: t_c = 3.81s.

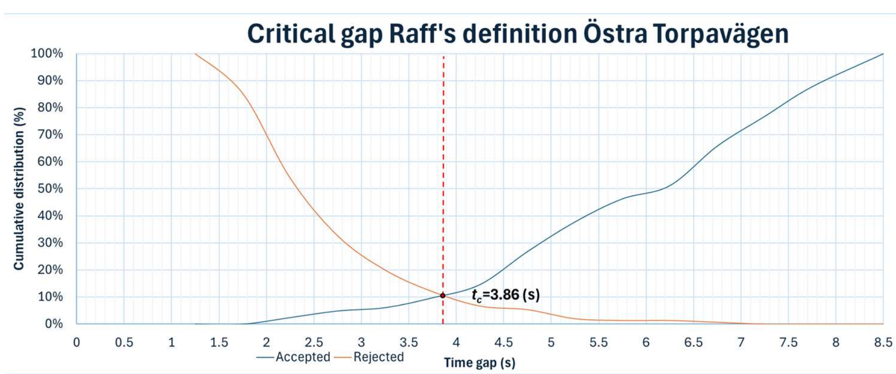

Figure 21: Critical gap calculated with Raff's definition for the gap acceptance data collected at the roundabout: Östra Torpavägen/ Smörslottsgatan.

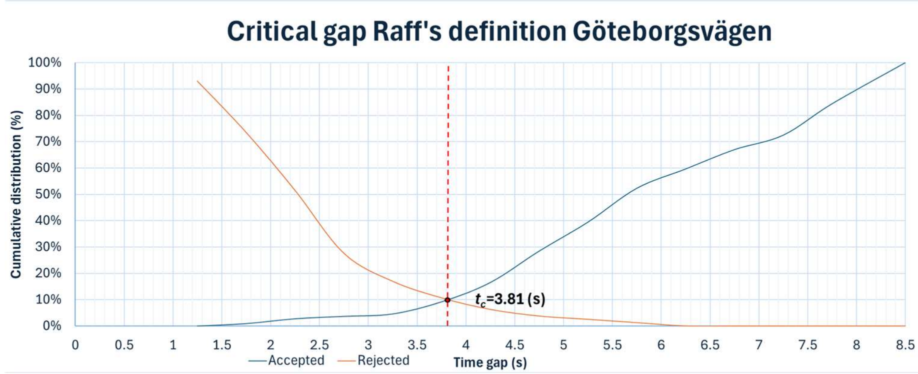

Figure 22: Critical gap calculated with Raff's definition for the gap acceptance data collected at the roundabout: Göteborgsvägen/ Tillfällavägen/ Stora Ringvägen.

Raff's method was also applied to the total gap acceptance data which combined the data from all roundabouts into one dataset, see Table 11: The total gap acceptance data collected at all roundabouts, to create an overall value for the critical gap. The results are presented in the graph below, see Figure 23, and the overall critical gap value for the total data were calculated to be: t_c = 3.73s.

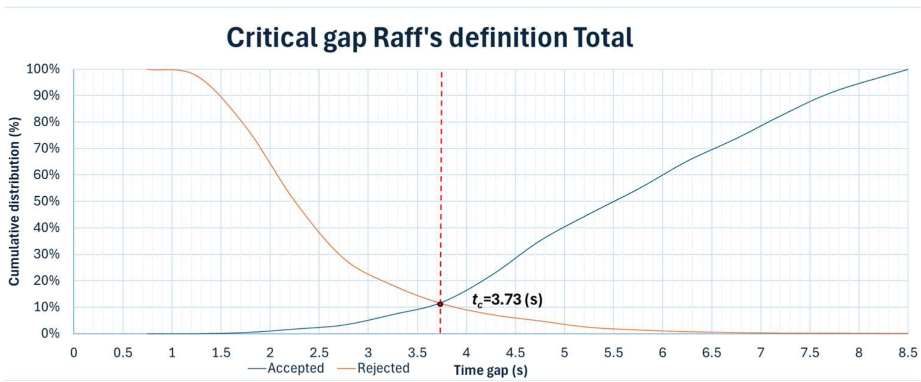

Figure 23: Critical gap calculated with Raff's definition for the combined gap acceptance data collected at all roundabouts

Table 13: Summarized data table

Location	Critical Gap (s)	Accepted %	Rejected %	N (gaps)
Kung Göstas Väg	3.68	30%	70%	840
Östra Torpavägen	3.86	35%	65%	232
Göteborgsvägen	3.81	41%	59%	268
Combined	3.73	33%	67%	1361

4.3 SIDRA Intersection

The critical gap values calculated for each location, see Section 4.2, were used during the analysis in SIDRA Intersection and compared to the default value for critical gap, t_c= 4.00s. For the results presented in this chapter, only the analysis with the largest difference in the simulated result is shown. The location with the largest difference in the simulated result was, logically, the location with the largest difference between the calculated critical gap and the default critical gap, Kung Göstas Väg with t_c= 3.68s. In the results presented it is clear that the different values provide substantial differences for a multitude of parameters. Especially notable is that the degree of saturation is reduced from 0.90 while using the default value for critical gap, down to 0.75 with the calculated critical gap, indicating that the actual congestion levels at this location are significantly lower than that expected with standard values. Other parameters of interest can be observed in the table below, see Table 14. Queue lengths were substantially reduced, implying that default parameters overestimate congestion.

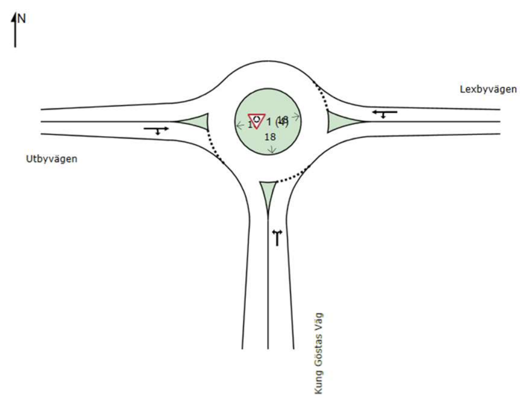

Figure 24: An overview of the location, as displayed in SIDRA Intersection

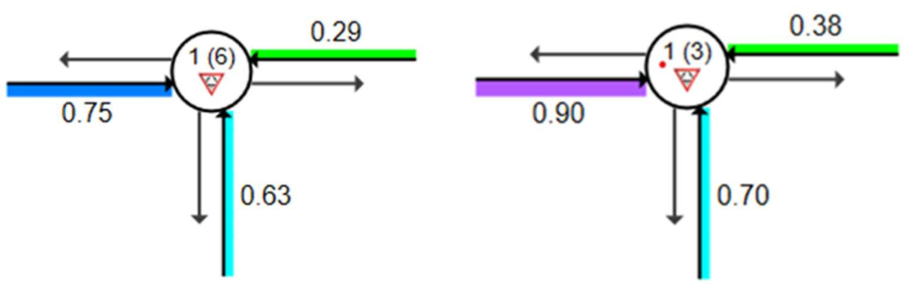

Figure 25: A comparison of the degree of saturation for the different approaches to the roundabout. The scene to the left displays the degree of saturation with the calculated critical gap and to the right the analysis with the default values

Table 14: The simulated results for different parameters with the calculated critical gap and the default value.

Parameter	Demand flow (veh/h)	Effective capacity (veh/h)	Degree of saturation	Back of queue, average (m)	Back of queue, 95th (m)
Input critical gap: 3.68 s	2181	2908	0.75	27.5	62.6
Default critical gap: 4.00 s	2181	2434	0.90	57.5	142.8

4.4 Classification algorithms

Four different machine learning classification algorithms were used in this thesis, Random forest, K-nearest neighbour, Support vector machine and Naïve bayes. The training- and testing accuracies of these algorithms are presented below, see Figure 26. The training accuracy represents how well the algorithm predicted the classification on the training dataset, which was provided to algorithm during the training and learning phase and which consisted of 80% of the total dataset. The testing accuracy represents how well the algorithm predicted the classification of the testing dataset, which was the remaining 20% of the dataset which the algorithm did not train on and which is commonly referred to as “the unseen data”. The results clearly presents that the Random forest algorithm had the highest percentage for both the training accuracy as well as the testing accuracy. The KNN- and the SVM-algorithms saw nearly identical accuracies for both testing and training data. The Naïve bayes classifier performed the lowest accuracy for both the training and the testing data.

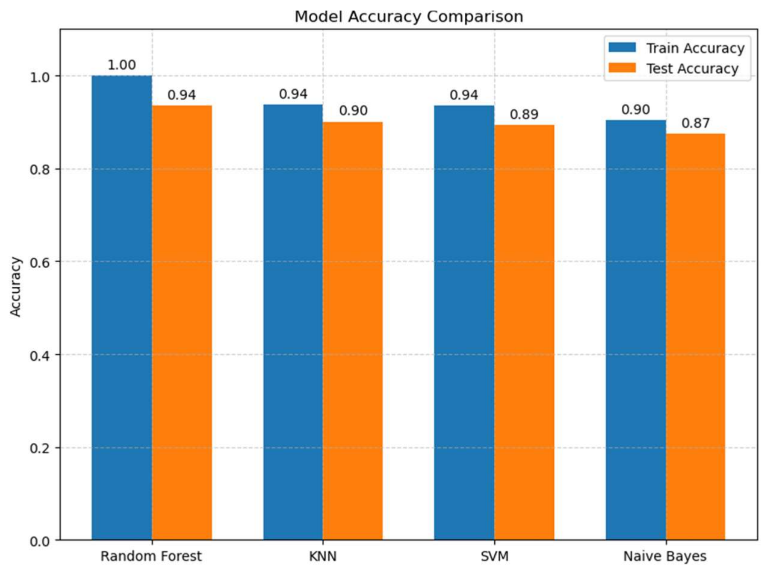

Figure 26: A presentation of the training- and testing accuracies of the classification algorithms

4.4.1 Random forest

The Random forest algorithm also provides results on how great the importance of each feature used in the algorithm is for impacting if a gap is accepted or rejected. All values included in the dataset, as well as the corresponding feature importance score is presented in the diagram below, see Figure 28. Offered gap was by far the most predictive feature, followed by number of vehicles rejected. Temporal and location factors had minor influence on the classification.

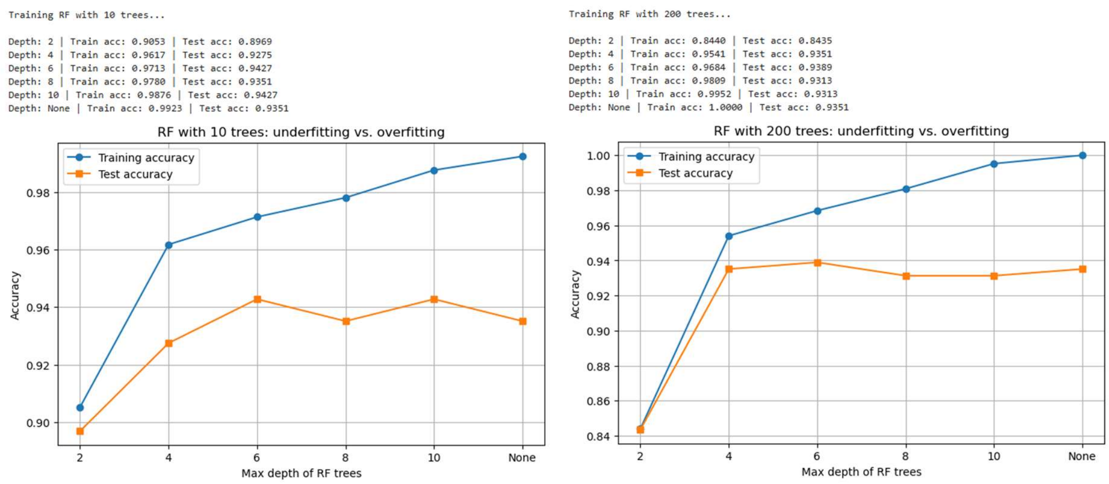

Figure 27: Diagrams presenting the difference in accuracy of the Random forest algorithm depending on how many decision trees and the maximum depth of the trees in the Random forest

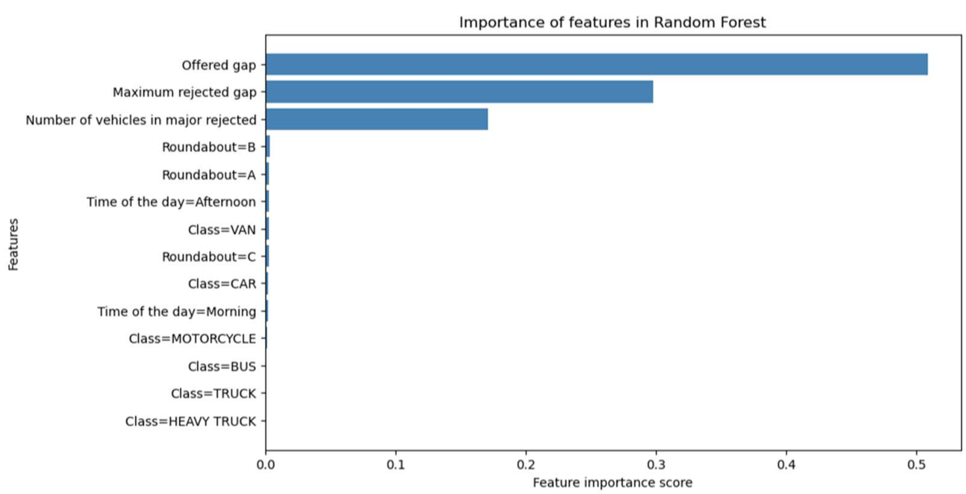

Figure 28: A diagram presenting the feature importance score of all values included in the dataset that was used in the machine learning classification algorithms.

4.4.2 K- Nearest neighbour

In the analysis from the K-Nearest neighbour, the impact of the values of "Offered gap" and "Number of vehicles in major rejected" were compared with each other. Both features had a high importance score in the Random forest algorithm and was therefore considered to be of high interest. By adapting the placement of the test point, some interesting results where shown. The results depicted in the graphs below, shows that the same value for an offered gap: 3 seconds, were predicted by the algorithm to be accepted by the waiting vehicle if the vehicle had not rejected any previous gaps, see Figure 29. This means that the vehicle accepts the first gap presented, but if a vehicle has already rejected two gaps, the algorithm instead predicts that the offered gap of 3 seconds will instead be rejected. The red-coloured dots are values indicating class 1: accepted gaps and the blue-coloured dots indicate class 0: rejected gaps. The testing point is shown in green colour and with circles showing which neighbors are affecting the prediction. The number of neighbors considered as well as the predicted class of the testing point is presented in the title above each chart. KNN highlighted that driver behavior changes depending on whether it is the first gap or after multiple rejections.

```
KNN Training Accuracy: 0.9378
KNN Test Accuracy: 0.9008
```

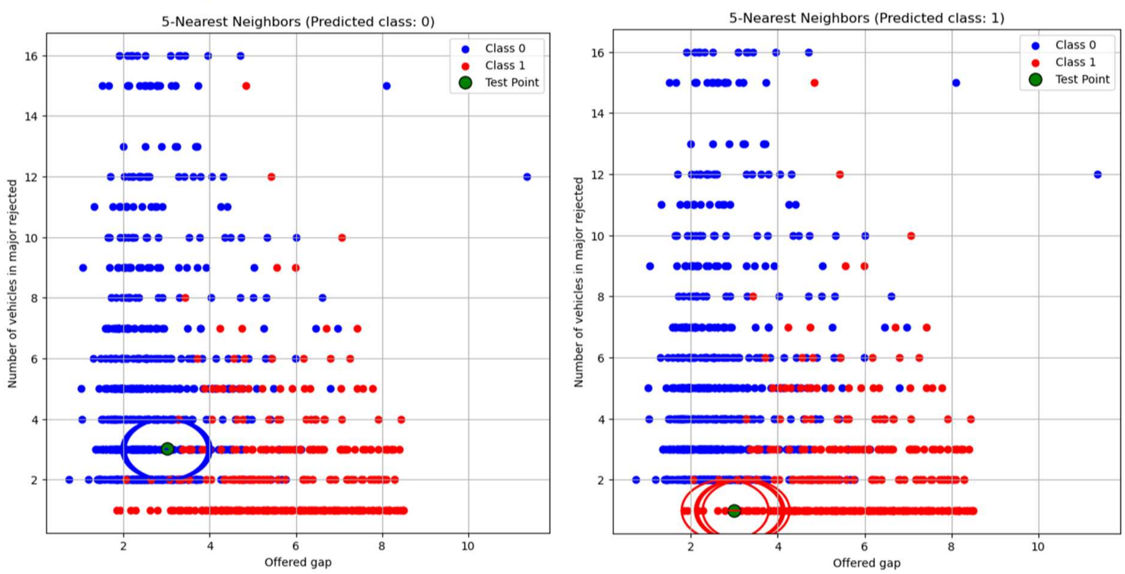

Figure 29: The predicted acceptance or rejection by the algorithm if the 3 seconds of offered gap is the first gap presented or if the waiting vehicle have already rejected two vehicles.

The reasoning for a gap having a higher likelihood of being accepted if it is the very first gap presented to the driver compared to a later one, could intuitively be assumed to be because the driver that accepts the very first gap is approaching the roundabout at a higher velocity than the one which is already waiting. This would eliminate the necessity of accelerating when entering the roundabout, allowing the physical possibility to enter with a smaller time gap. The trend does, however, also proceed when comparing higher values of "Number of vehicles in major rejected". When the algorithm predicts acceptance or rejection of a offered time gap of 4 seconds, the results differ if the waiting vehicle have rejected two gaps previously or if the driver have rejected four gaps previously, see Figure 30.

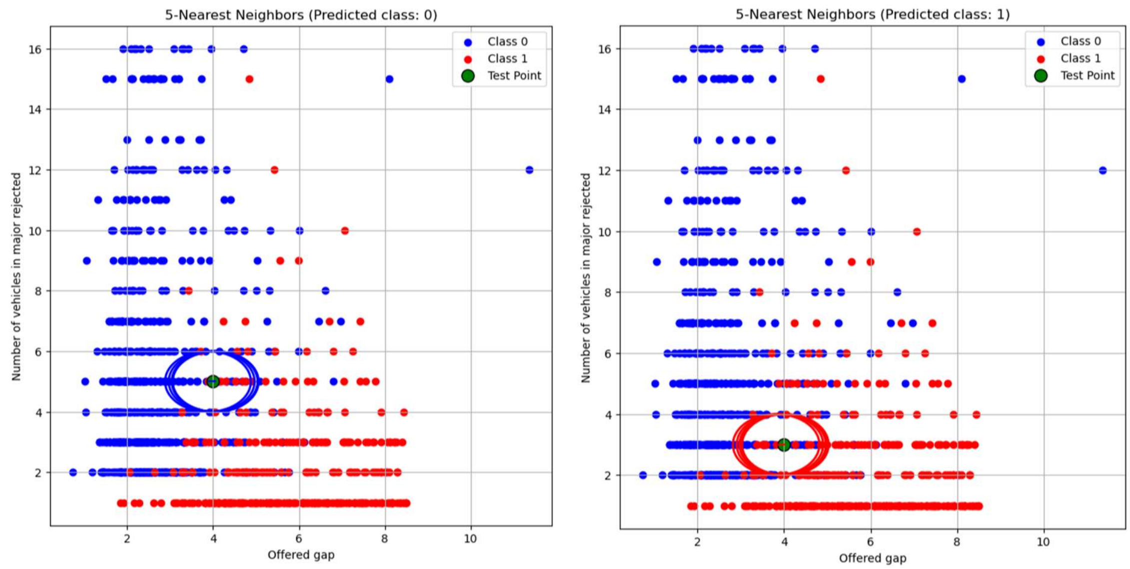

Figure 30: A comparison of the K-nearest algorithm predicting acceptance or rejection of an offered gap of 4 seconds depending on if the vehicle has already rejected two or four gaps.

4.4.3 Support vector machines

Following on the results from the feature importance score in Random forest, see Section 4.4.1, the relation between "Offered gap" and "Vehicles in major rejected" as well as "Offered gap" and "Maximum rejected gap" was researched further. The results from the Support vector machine algorithm are depicted in the following diagrams, see Figure 31 and Figure 32. In the results, the training data is shown in red and green dots, indicating if the gap was accepted or rejected and the testing data is shown as the black crosses. The resulting classification divided by the algorithm is shown as the red and green areas, where the red area is where the algorithm predicts the gap to be rejected and the green area where the algorithm predicts the gap to be accepted.

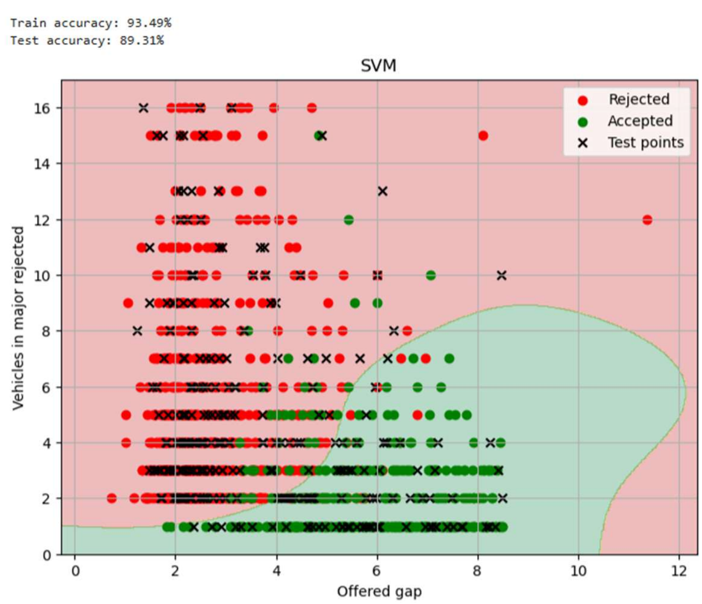

Figure 31: The classification divide predicted by the SVM-algorithm in the case of "Offered gap" and "Vehicles in major rejected"

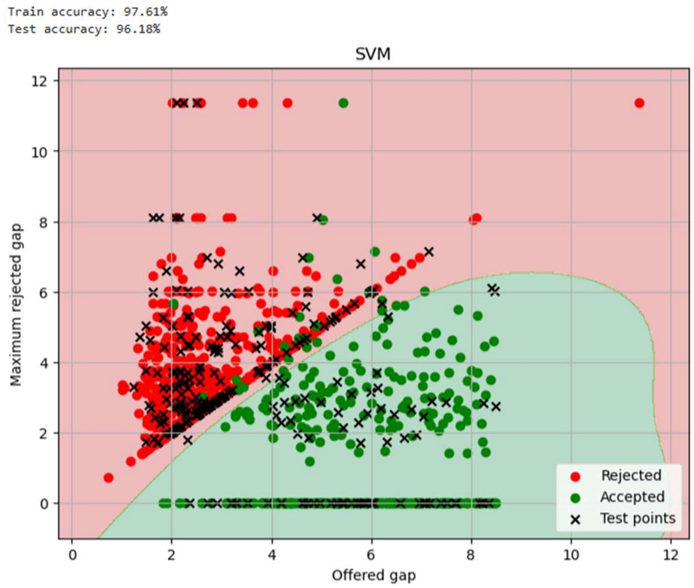

Figure 32: The classification divide predicted by the SVM-algorithm in the case of "Offered gap" and "Maximum rejected gap"

4.4.4 Naive Bayes classifier

Similarly to the case of the SVM-algorithm the Naïve Bayes classifier was also used to research the relation between "Offered gap" and "Vehicles in major rejected" as well as "Offered gap" and "Maximum rejected gap". By using the same parameters for both algorithms, a comparison could be made to determine what algorithm produced the best prediction for these datasets. The results from the Naïve Bayes classifier are shown in the following diagrams, see Figure 33 and Figure 34. Like the results from the SVM-algorithm, the training data is shown in red and green dots, indicating if the gap was accepted or rejected and the testing data is shown as the black crosses. The prediction done by the algorithm is shown as the red and green areas, where the green color indicates an accepted gap, and the red area indicates a rejected gap. SVM and NB confirmed that offered gap and number of rejections dominate classification, though with lower accuracy than RF.

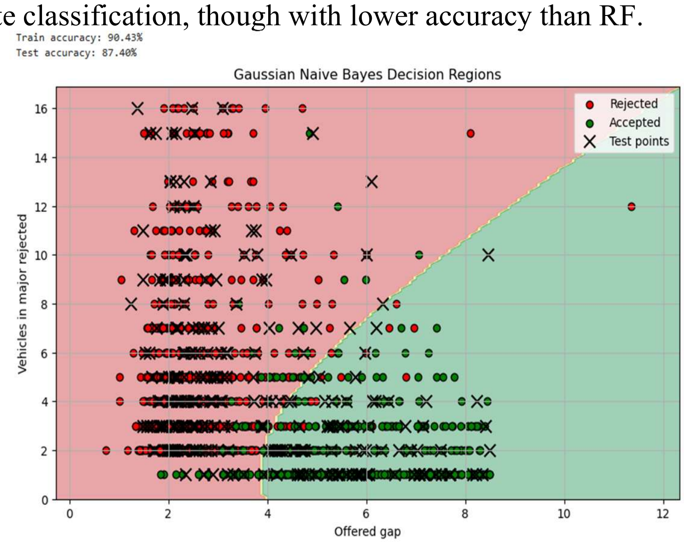

Figure 33: The classification results from the Naïve bayes classifier for the parameters "Offered gap" and "Vehicles in major rejected"

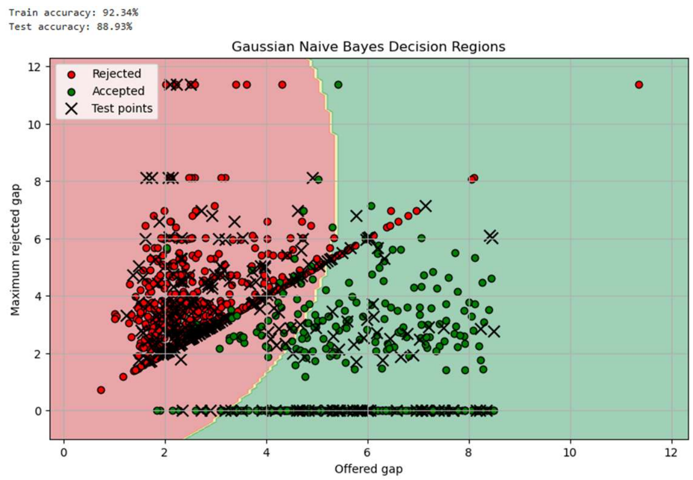

Figure 34: The classification results from the Naïve bayes classifier for the parameters "Offered gap" and "Maximum rejected gap

5 Discussion

In this chapter, the findings gathered from this study are discussed and linked to the research questions which this thesis aimed at answering. It includes the results from the calculation of the critical gaps, the comparison of implementing the different parameters in analysis as well as the result on parameters impacting Gap acceptance and the reasoning behind it.

5.1 Gap acceptance parameters and the impact on analysis software

The results from the data collection, see Section 4.1, presented gap acceptance parameters from three different roundabouts in Sweden. The roundabouts showed very similar results to each other in a multitude of aspects. Approximately half of all the gaps offered at all roundabouts were below 3 seconds and had a acceptance ratio of 2.3% with a corresponding rejection ratio of 97.7%. Only two out of a total of 197 gaps offered below 2 seconds were accepted, both times resulting in hard braking from the vehicle driving inside the roundabout to avoid collision. Such exceptions, though rare, suggest that relying on very small gaps may compromise safety, supporting the need for conservative critical gap estimates.

On the other side of the spectrum, less than 2% of all rejected gaps was larger than 6 seconds and only three gaps in total were larger than 7 seconds. All three gaps over 7 seconds were observed closer in the video recordings because of their clear points as outliers. At all of the occasions, it was found that the vehicles circulating inside the roundabout did not use the indicators while exiting the roundabout and therefore left the waiting vehicle with a much shorter time to react to newly presented gap.

The effect of drivers not using indicators while exiting the roundabout has the possibility to skew the results of the critical gap calculation to be larger than the actual value, both like in the case just described, but also by creating uncertainty in the traffic environment leading to more caution and larger thresholds for accepting gaps. After discovering that the largest rejected gaps were caused by the neglect to use turn signals, the video recordings for the rejected gaps between 5 and 7 seconds were also studied. No further instances of this situation occurring were found while researching the rest of the video recordings and it could therefore be expected to have a very low effect on the total data collection as well as the critical gap calculations, but further studies on this effect could be of great interest.

At all roundabouts, “the turning point”, in this case meaning the time range where the gaps switched from being majority rejected to being majority accepted, were shown to be above 4 seconds. This is found interesting in the context of the critical gap being calculated to be lower than 4 seconds at all roundabouts. The reason for this is, as previously discussed, see Section 2.4, because all drivers are not offered the same gaps. This becomes a problem when drivers with a high threshold to accept gaps is presented with a gap large enough that a driver with a low threshold would have accepted it. On the contrary, a driver with a low threshold might only be presented with large gaps at the roundabout and when accepting that gap, the data collected would indicate that also this driver had a high threshold to accept gaps. This dilemma skews the data of gap acceptance and is the reasoning for calculating critical gap with Raff’s method rather than simply assuming that the critical gap is equal to the value of the average accepted gap. This confirms that Raff’s method remains more reliable than average observed gap acceptance, especially in heterogeneous driver populations.

The values of the critical gaps calculated with Raff's method from the three roundabouts were 3.68s, 3.81s and 3.86s respectively. Making up a total value of critical gap to 3.73s. Comparatively, the standard value for critical gap in SIDRA Intersection is 4.00s. There is a great quantity of potential reasons for the critical gaps calculated on Swedish data being lower than that of SIDRAs standard value which is based on Australian data. One of the main supposed factors could be that of the geometric dimensions of the roundabouts. The roundabouts used for the data collection in this thesis work had a diameter of the central island ranging from 16-18m which is a relatively small value compared to Australia where the desirable diameter for designing roundabouts is at least 20m for the lowest speed limits (Austroads, 2023). Larger diameters of the central island does as previously mentioned, see Section 2.4, enable the vehicles to travel at a higher speed leading to more caution and therefore a higher threshold for accepting gaps. Other contributing factors for the lower values of critical gap could be related to the difference in vehicle types, not only the difference between light and heavy traffic, but also the effect of Sweden's rise of electric vehicles. Electric vehicles does provide a faster acceleration than their non-electric counterparts, petroleum- and diesel vehicles (Wang et al., 2025). The increase in acceleration provides a vehicle waiting to enter a roundabout with greater conditions and possibilities for accepting smaller gaps. More than 110 000 or approximately 38% of newly registered cars in Sweden during 2023 were electric vehicles meaning that they have a substantial impact on the evolution of gap acceptance parameters in Sweden and could be of great interest to research further (SCB, 2025)

When inputting the location-based critical gap values into SIDRA Intersection and comparing them to analyzing with the standardized value it was possible to see some significant impacts, see Section 4.3. For the roundabout which had the largest traffic flow during the recorded hours, and which is presented in the results, SIDRA calculated an increase in effective capacity from approx. 2400 veh/h to 2900 veh/h or roughly 25% increase. This affected the analysis greatly and reduced the degree of saturation from 0.90 to 0.75 at the specific roundabout which made it pass the recommended values for a sufficient roundabout according to Swedish Traffic Authority (Trafikverket, 2014). When analyzing the other recorded roundabouts with their corresponding traffic volumes the impact of changing the critical gap value varied. The results show that calculating a critical gap value from data collected at the location and implementing it in SIDRA could have a large impact on the analysis, but it is not certain. Hence, local calibration of critical gap values is essential; using standardized values risks overestimating congestion and underutilizing capacity. The method can serve as a very useful tool to help improve traffic analysis, but it will be up to every engineer to make a logical decision when to invest the time and when the investment will not be worth the difference. When more data gets collected, the easier the decision will be, more data will already be available to adjust the critical gap to similar roundabouts and the effects on certain parameters will be clearer.

As mentioned in the methodology chapter, see Section 3.2, the data collection in this thesis was effectivized by using GoodVision for extracting data from the video recordings. Previously gap acceptance data have been extracted by manually observing and measuring recorded video, leading it to be very time consuming and easy to make errors or inconsistent measuring. The use of GoodVision in this thesis is considered to have been greatly timesaving, enabling a larger data size to be gathered and in the end leading to a more accurate result. The future use of GoodVision or similar softwares is seen as very beneficial and can provide crucial assistance in the future of gap acceptance research.

5.2 Parameters impact on accepted and rejected gap

The importance of different parameters regarding why a gap is accepted or rejected was analyzed by implementing four different machine learning classification algorithms. Random forest, K-nearest neighbor, Naïve Bayes and Support vector machine were all used in analysis. The results from the importance of features showed that the size of the gap, as well as the history of each individual driver in terms of size of previous rejected gaps and number of previous rejected gaps were the outstanding most important parameters. The results indicated that the individual driving behavior of each driver was significantly more important than the difference in roundabout geometry, the difference in vehicle type and the difference in time of the day. The results are to be observed with a certain caution since the difference between the three roundabouts were not very large in terms of geometrical properties it is logical that these parameters did not have a significant impact on the decision of acceptance or rejection. For the parameters concerning different vehicle types there is also some concern about a too small size of data for certain vehicle types to have a large impact on the full results. One conclusion that could be drawn from the results were that driving behavior did have a substantial impact on deciding whether to accept or reject a gap and the relationship between the different parameters were analyzed with both Naïve Bayes and SVM.
As seen in the results, see Section 4.4.3 and Section 4.4.4, both the Naïve Bayes classifier and the SVM is attempting to present a clear boundary between an accepted and a rejected gap. For the results from the support vector machine algorithm it can be observed, see Figure 33, an almost linear trend between "maximum rejected gap" and "offered gap" between 2 and 6 seconds. The algorithm indicates that no vehicle has previously rejected a value of above 6.5 seconds and then later accepted a gap of any value. The reasoning for this stagnation while moving further along the x-axis and even sharp decline when reaching offered gaps over 11.5 seconds is the limitations of the SVM-algorithm. Because the algorithm aims to draw a line between accepted and rejected gaps as far from each other as possible, the low amount of data points while reaching the outer range of the data set makes for some quite illogical predictions. As in this case it is expecting an offered gap of 12 seconds to be rejected, regardless of the previous history of the driver. This must be considered when evaluating the SVM-algorithm, but it does provide good performance and indication when presented with a large enough data set. Since only three roundabouts of similar geometry were included, the low importance of geometric features may reflect dataset homogeneity rather than universal insignificance.

When observing the same parameters with the Naïve Bayes classifier the same linear trend is not presented. The algorithm shows a minimum accepted gap of just above 2 seconds and then proceeds with a more exponential growth until it predicts the maximum rejected value to be approximately 5 seconds. The algorithm is also affected by one clear outlier at the top of the diagram, see Figure 34, where one vehicle have rejected a gap of over 11 seconds and then proceeded to accept a gap of approximately 5.5 seconds. This outlier could potentially also be a vehicle which experienced the exiting flow effect.

When instead comparing the other parameter "vehicles in major rejected" to "offered gap" it can be observed in the Naïve Bayes classifier, see Figure 33, that the algorithm predicts the minimum accepted gap to be just below 4 seconds. The algorithm predicts that all vehicles that reject two gaps or less will accept a presented gap which is larger than 4 seconds. For vehicles which reject three gaps or more the algorithm predicts an almost linear trend where a greater number of rejected gaps indicates a larger threshold. For example, the algorithm predicts that a vehicle which has already rejected 5 gaps will require a gap of at least 5 seconds to accept it. The results from the algorithm prediction could also be interpreted as the average vehicle that has rejected more than three gaps have a critical gap of over 4 seconds.

For the SVM-algorithm it can be observed, see Figure 31, a prediction that only the vehicles accepting the initial or the second gap presented will accept offered gaps below 4 seconds. The algorithm then predicts a sharper threshold at around 5-5.5 seconds for all vehicles that rejects between two to five gaps. Like previous, the algorithm show some quite nonsensical predictions in the other parts of the dataset, for example predicting that offered gaps of 0 seconds will be accepted if it is the initial gap and that some gaps of above 10 seconds will be rejected. However, KNN and SVM converged in predicting a behavioral threshold around 4–5s after repeated rejections, indicating fatigue/impatience effects in waiting drivers.

5.3 Further research

Gap acceptance is a broad subject and there is a lot of different areas which could be subject to further research. It would be of great interest to further research the possibility of using AI-based softwares in traffic analysis as a whole and specifically in the gap acceptance area. The use of a static camera in this thesis limited the possible locations to gather data from, but the use of, for example a drone, could help to solve this issue and broaden the possibilities greatly. This thesis found that the single greatest importance feature was behavioral variability between different drives and it could therefore be of high interest to research these parameters further. Collecting data from roundabouts with very high levels of congestion could be relevant to finding a form of lower limit in terms of behavioral change in relation to critical gap and maybe include to what extent it affects traffic safety as well. As briefly mentioned in the discussion chapter the rapid growth of the proportion of electrical vehicles in Swedish traffic environment could lead to an interesting future study on a comparison between critical gaps in EV and traditional combustion engine vehicles.

6 Conclusion

To conclude this thesis, it was found that the critical gap parameters in three separate Swedish roundabouts were 3.68s, 3.81s and 3.86s respectively. Across three Swedish roundabouts, critical gap values were consistently lower than the standardized SIDRA value, reinforcing the importance of context-specific calibration. They were all found to be lower than the standardized value in SIDRA Intersection of 4.00s, which they were to be compared to. While no single factor could be isolated, the study identified several plausible influences, from roundabout size to driver behavior and vehicle technology, that requires further investigation. The three different roundabouts used for the data collection in this thesis had very similar geometric properties and no clear distinction could be made regarding this parameter's impact on gap acceptance.

The data collection was performed by recording roundabouts with a static camera and analyzing the video with the AI-based software GoodVision. This methodology was greatly successful and sped up the data collection process severely. This new methodology of AI-based video analytics for traffic data collection has the possibility to replace traditional manual methods and enables larger and more extensive future studies in the field.

Researching and collecting data from every location when performing analysis in a software like SIDRA Intersection will not be sustainable and time-justified in comparison to the difference in result every time. The findings from this thesis do, however, show that it has the potential to have significant impact on the results in certain circumstances and that collecting data and calculating critical gaps with the presented method could turn out to be a very favorable decision when making intersection analysis both in the area of research and in a working environment.

References

Actam. (2008). *Roundabout intersection diagram*. Wikipedia.

Akcelik & Associates Pty Ltd. (2025a). *Brief Version History of SIDRA INTERSECTION*. https://www.sidrasolutions.com/software/sidra-intersection/features/introduction/version-history

Akcelik & Associates Pty Ltd. (2025b). *What is SIDRA INTERSECTION?* https://www.sidrasolutions.com/software/sidra-intersection/features/introduction/what-is-sidra-intersection

Alomari, A. H., & Lebdeh, E. A. (2022). Smart Real-Time Vehicle Detection and Tracking System Using Road Surveillance Cameras. *Journal of Transportation Engineering, Part A: Systems*, *148*(10), 04022076. https://doi.org/10.1061/JTEPBS.0000728

ASHWORTH, R. (1970). The Analysis and Interpretation of Gap Acceptance Data. *Transportation Science*, *4*(3), 270–280. http://www.jstor.org/stable/25767570

Chen, R., & Hourdos, J. (2018). Evaluation of the roundabout capacity model in HCM6 edition and HCM 2010 on a multilane roundabout. *Transportation Research Record*, *2672*(15), 23–34. https://doi.org/10.1177/0361198118794287

Chun, G., Rouphail, N. M., & Akçelik, R. (2025, January). Roundabout Capacity Survey Method Using Vehicle Trajectory Data and Gap Acceptance Cycles. *Roundabout Capacity Survey Method Using Vehicle Trajectory Data and Gap Acceptance Cycles*.

Çolak, S., Lima, A., & González, M. C. (2016). Understanding congested travel in urban areas. *Nature Communications*, *7*. https://doi.org/10.1038/ncomms10793

DataFromSky. (2025). *DataFromSky*. https://datafromsky.com/trafficsurvey/

Gerlough, D. L. (1955). *Simulation of Freeway Traffic on a General-purpose Discrete Variable Computer*. University of California.

GoodVision. (2025a). *GoodVision*. https://goodvisionlive.com/

GoodVision. (2025b). *GoodVision Traffic Data Collection*. https://goodvisionlive.com/solutions/traffic-data-collection/

GoodVision. (2025c). *GoodVision Traffic Modelling*. https://goodvisionlive.com/solutions/traffic-modelling/

GoodVision. (2025d). *GoodVision Video Insights*. https://goodvisionlive.com/goodvision-video-insights/

Google. (2025). *Google Street View*. https://www.google.com/streetview/

Guo, R. J., Wang, X. J., & Wang, W. X. (2014). Estimation of critical gap based on raff's definition. *Computational Intelligence and Neuroscience*, *2014*. https://doi.org/10.1155/2014/236072

Ho, T. K. (1995). Random decision forests. *Proceedings of 3rd International Conference on Document Analysis and Recognition*, *1*, 278–282 vol.1. https://doi.org/10.1109/ICDAR.1995.598994

Lieberman, E. B., Associates, K., York, N., Worrall, R. D., & Bruggeman, J. (1972). *LOGICAL DESIGN AND DEMONSTRATION OF UTCS-1 NETWORK SIMULATION MODEL*.

McDonald, M., Brackstone, M., & Sultan, B. (1998). *INSTRUMENTED VEHICLE STUDIES OF TRAFFIC FLOW MODELS*. https://api.semanticscholar.org/CorpusID:109234469

Mousa, R. M. (2004). *Simulation Modeling Techniques and Application in Highway Traffic Management: A State-of-the Art Review*. https://doi.org/10.13140/RG.2.1.3226.9843

Murray, L. (2024, November 11). *Arc de Triomphe*. Encyclopedia Britannica.

OpenStreetMap Foundation. (2025). *OpenStreetMap®*. https://www.openstreetmap.org/copyright

Qiao, F., Liu, T., Sun, H., Guo, L., & Chen, Y. (2021). Modelling and simulation of urban traffic systems: present and future. *International Journal of Cybernetics and Cyber-Physical Systems*, *1*, 1. https://doi.org/10.1504/IJCCPS.2021.113100

Retting, R., Persaud, B., Gårder, P., & Lord, D. (2001). Crash and injury reduction following installation of roundabouts in the United States. *American Journal of Public Health*, *91*, 628–631. https://doi.org/10.2105/AJPH.91.4.628

Rodegerdts, L. . (2010). *Roundabouts : an informational guide, 2nd ed*. TRB.

Ross, P. Van, & Gibson, D. R. (1977). REVIEW OF ROAD TRAFFIC NETWORK SIMULATION MODELS. *Transportation Research Record*. https://api.semanticscholar.org/CorpusID:61825927

SCB. (2025, August 13). *Elbilsmarknaden har mattats av*. SCB. https://www.scb.se/hitta-statistik/redaktionellt/elbilar/

Trafikverket. (2014). *TRVMB Kapacitet och framkomlighetseffekter*.

Transportation Research Board. (2000). *Highway Capacity Manual (4th Edition)* (4th ed.). Transportation Research Board.

Transportstyrelsen. (2025, January 1). *Trängselskatt i Göteborg*. Transportstyrelsen.

Appendix

Python code:

CSV file:

```
import os
import pandas as pd
import numpy as np
import random
import string
from datetime import datetime
folder_path = 'Data'

import warnings
major_sheet_name = "VEHICLE LIST (Major)"
minor_sheet_name = "VEHICLE LIST   (Minor Wait -> Min"
workbooks_with_data = []
# categories = ["A", "B", "C", "D", "E", "F", "G", "H", "I", "J", "K", "L", "M"]
base_categories = list(string.ascii_uppercase)
extended_categories = ['1' + letter for letter in string.ascii_uppercase]
categories = base_categories + extended_categories
print(categories)
for filename in os.listdir(folder_path):
  if filename.lower().endswith(('.xls', '.xlsx')):
    file_path = os.path.join(folder_path, filename)
    try:
      sheet_names = pd.ExcelFile(file_path).sheet_names
      if major_sheet_name in sheet_names or minor_sheet_name in sheet_names:
        category = random.choice(categories)
        workbooks_with_data.append((category, filename))
    except Exception as e:
      print(f"Error reading {filename}: {e}")


print("Detected Excel workbooks with relevant sheets:")
for category, name in workbooks_with_data:
  print(f"Roundabout {category}: {name}")
import os
import pandas as pd
import numpy as np
import random
import string
from datetime import datetime

# === SETTINGS ===
data_folder = "Data"
major_sheet_expected = "vehicle list (major)"
minor_sheet_expected = "vehicle list (minor wait -> min"

categories = list(string.ascii_uppercase) + ["1" + c for c in string.ascii_uppercase]
results = []

# === HELPER FUNCTIONS ===
def clean_time_string(time_str):
  if pd.isna(time_str):
    return time_str
  return str(time_str).replace(",", ".")

def extract_date_time(datetime_str):
  dt = pd.to_datetime(datetime_str)
  return dt.date(), dt.time()

print(f"Starting analysis in folder: {data_folder}")
```

```
for filename in os.listdir(data_folder):
    if filename.lower().endswith((".xls", ".xlsx")):
        file_path = os.path.join(data_folder, filename)
        try:
            xl = pd.ExcelFile(file_path)
            print(f"\nProcessing file: {filename}")
            sheet_map = {s.strip().lower(): s for s in xl.sheet_names}

            if major_sheet_expected in sheet_map and minor_sheet_expected in sheet_map:
                roundabout = random.choice(categories)

                major = pd.read_excel(file_path, sheet_name=sheet_map[major_sheet_expected])
                minor = pd.read_excel(file_path, sheet_name=sheet_map[minor_sheet_expected])

                major["ENTRY TIME"] = major["ENTRY TIME"].apply(clean_time_string)
                minor["ENTRY TIME (Minor Wait)"] = minor["ENTRY TIME (Minor Wait)"].apply(clean_time_string)
                minor["EXIT TIME (Minor In)"] = minor["EXIT TIME (Minor In)"].apply(clean_time_string)
                major["ENTRY TIME"] = pd.to_datetime(major["ENTRY TIME"], errors='coerce')
                minor["ENTRY TIME (Minor Wait)"] = pd.to_datetime(minor["ENTRY TIME (Minor Wait)"], errors='coerce')
                minor["EXIT TIME (Minor In)"] = pd.to_datetime(minor["EXIT TIME (Minor In)"], errors='coerce')

                for _, row in minor.iterrows():
                    vehicle_id = row["ID"]
                    class_type = row["CLASS"]
                    entry_time = row["ENTRY TIME (Minor Wait)"]
                    exit_time = row["EXIT TIME (Minor In)"]

                    if pd.isna(entry_time) or pd.isna(exit_time):
                        continue

                    # Check for overlap with major entries
                    mask = (major["ENTRY TIME"] >= entry_time) & (major["ENTRY TIME"] <= exit_time)
                    rejected = major.loc[mask].copy()

                    if rejected.empty:
                        continue

                    date, start_time = extract_date_time(entry_time)
                    _, end_time = extract_date_time(exit_time)

                    num_rejected = len(rejected)
                    last_rejected_time = rejected["ENTRY TIME"].max() if num_rejected > 0 else pd.NaT

                    rejected_times = rejected["ENTRY TIME"].sort_values().reset_index(drop=True)
                    gaps = rejected_times.diff().dropna().reset_index(drop=True)

                    # Compute max rejected gap (for entire sequence)
                    max_rejected_gap = gaps.max().total_seconds() if not gaps.empty else np.nan

                    # Identify accepted gap as the gap after the last rejected vehicle in the major stream
                    accepted_gap = np.nan
                    if not rejected_times.empty:
                        major_times_sorted = major["ENTRY TIME"].sort_values().reset_index(drop=True)
                        if last_rejected_time in major_times_sorted.values:
                            idx = major_times_sorted[major_times_sorted == last_rejected_time].index[0]
                            if idx + 1 < len(major_times_sorted):
                                accepted_gap = (major_times_sorted[idx + 1] - last_rejected_time).total_seconds()

                    # Add decision = 0 for each rejected major vehicle with offered gap
                    for i, (_, mj) in enumerate(rejected.iterrows()):
                        rejected_gap = gaps[i - 1].total_seconds() if i > 0 and i - 1 < len(gaps) else np.nan
                        results.append({
                            "Vehicle ID": vehicle_id,
                            "Major vehicle ID": mj["ID"] if "ID" in mj else np.nan,
                            "Roundabout": roundabout,
                            "Class": class_type,
```

```
            "Date": date,
            "Minor stream vehicle approaching": start_time,
            "Minor stream vehicle entering major stream": end_time,
            "Number of vehicles in major rejected": num_rejected,
            "Last rejected vehicle passage timestamp": last_rejected_time.time() if pd.notna(last_rejected_time) else
np.nan,
            "Maximum rejected gap": max_rejected_gap,
            "Offered gap": rejected_gap,
            "Decision": 0
          })

        # Add decision = 1 row for final accepted vehicle
        results.append({
          "Vehicle ID": vehicle_id,
          "Major vehicle ID": np.nan,
          "Roundabout": roundabout,
          "Class": class_type,
          "Date": date,
          "Minor stream vehicle approaching": start_time,
          "Minor stream vehicle entering major stream": end_time,
          "Number of vehicles in major rejected": num_rejected,
          "Last rejected vehicle passage timestamp": last_rejected_time.time() if pd.notna(last_rejected_time) else
np.nan,
          "Maximum rejected gap": max_rejected_gap,
          "Offered gap": accepted_gap,
          "Decision": 1
        })

  except Exception as e:
    print(f"Error processing {filename}: {e}")

# Final dataframe
df = pd.DataFrame(results)

# Remove rows with missing Offered gap
df = df.dropna(subset=["Offered gap"])

# Remove rows where Offered gap > 8.5 and Decision == 1
df = df[~((df["Offered gap"] > 8.5) & (df["Decision"] == 1))]

print("\n=== FINAL OUTPUT ===")
print(f"Total rows: {len(df)}")
print(df.head())
```

Random forest:

```
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split, cross_val_score
from sklearn.feature_extraction import DictVectorizer
from sklearn.pipeline import make_pipeline
from sklearn.feature_extraction import DictVectorizer
from sklearn.ensemble import RandomForestClassifier

pipeline = make_pipeline(
  DictVectorizer(),  # One-hot encoding
  RandomForestClassifier(n_estimators=100, random_state=0)
)

from sklearn.metrics import accuracy_score
import warnings
warnings.filterwarnings("ignore")

# Load the cleaned CSV file
df = pd.read_csv("C:\\Users\\VICHAN\\Downloads\\Python\\data.csv")
```

```
# Drop non-informative or problematic columns
drop_cols = ["Vehicle ID", "Major vehicle ID", "Date",
        "Minor stream vehicle approaching", "Minor stream vehicle entering major stream",
        "Last rejected vehicle passage timestamp"]
df = df.drop(columns=drop_cols)

# === Prepare features and target ===
X = df.drop(columns=["Decision"])     # Feature matrix
y = df["Decision"]             # Target variable (classification)

# === Train-test split ===
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=0)

# === One-hot encoding of categorical features ===
dv = DictVectorizer()
X_train_encoded = dv.fit_transform(X_train.to_dict(orient="records"))
X_test_encoded = dv.transform(X_test.to_dict(orient="records"))

# === Define depths to test ===
max_depths = [2, 4, 6, 8, 10, None]

# === Train and evaluate Random Forests ===
n_estimators_values = [1, 5, 10, 50, 100, 200]

for n_trees in n_estimators_values:
  train_accuracies = []
  test_accuracies = []
  print(f"\nTraining RF with {n_trees} trees...\n")

  for depth in max_depths:
    rf_model = RandomForestClassifier(n_estimators=n_trees, max_depth=depth, random_state=0, n_jobs=-1)
    rf_model.fit(X_train_encoded, y_train)

    y_train_pred = rf_model.predict(X_train_encoded)
    y_test_pred = rf_model.predict(X_test_encoded)

    train_acc = accuracy_score(y_train, y_train_pred)
    test_acc = accuracy_score(y_test, y_test_pred)

    train_accuracies.append(train_acc)
    test_accuracies.append(test_acc)

    print(f"Depth: {depth} | Train acc: {train_acc:.4f} | Test acc: {test_acc:.4f}")

  # Plotting
  plt.figure(figsize=(8, 5))
  plt.plot([str(d) for d in max_depths], train_accuracies, label="Training accuracy", marker="o")
  plt.plot([str(d) for d in max_depths], test_accuracies, label="Test accuracy", marker="s")
  plt.xlabel("Max depth of RF trees")
  plt.ylabel("Accuracy")
  plt.title(f"RF with {n_trees} trees: underfitting vs. overfitting")
  plt.legend()
  plt.grid(True)
  plt.show()

dv = pipeline.steps[0][1]  # DictVectorizer
rf_model = pipeline.steps[1][1]  # RandomForestClassifier

# Access fitted steps by lowercase names
dv = pipeline.named_steps['dictvectorizer']
rf_model = pipeline.named_steps['randomforestclassifier']

# Get feature names and importances
feature_names = dv.get_feature_names_out()
importances = rf_model.feature_importances_
```

```
# Sort by importance and plot
import numpy as np
import matplotlib.pyplot as plt

indices = np.argsort(importances)[::-1]

plt.figure(figsize=(10, 6))
plt.title("Feature Importances from Random Forest")
plt.bar(range(len(importances)), importances[indices], align="center")
plt.xticks(range(len(importances)), feature_names[indices], rotation=90)
plt.tight_layout()
plt.show()
```

K-Nearest neighbor:

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier

# --- Load and prepare data ---
df = pd.read_csv("C:\\Users\\VICHAN\\Downloads\\Python\\data.csv")
# Pick two numeric features
features = ["Offered gap", "Number of vehicles in major rejected"]
target = "Decision"

df = df[features + [target]].dropna()

X = df[features].values
y = df[target].values

# Split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# --- Fit KNN Classifier ---
k = 5
knn = KNeighborsClassifier(n_neighbors=k)
knn.fit(X_train, y_train)

# Predict test point
test_point = X_test[0].reshape(1, -1)
predicted_class = knn.predict(test_point)[0]
from sklearn.metrics import accuracy_score

# Predict on test set
y_pred_knn = knn.predict(X_test)

from sklearn.metrics import accuracy_score

# Predict on training and test sets
y_pred_train_knn = knn.predict(X_train)
y_pred_test_knn = knn.predict(X_test)

# Compute accuracies
knn_train_accuracy = accuracy_score(y_train, y_pred_train_knn)
knn_test_accuracy = accuracy_score(y_test, y_pred_test_knn)

# Print results
print(f"KNN Training Accuracy: {knn_train_accuracy:.4f}")
print(f"KNN Test Accuracy: {knn_test_accuracy:.4f}")

# --- Plot ---
fig, ax = plt.subplots(figsize=(8, 8))

# Plot all training points
```

```
ax.scatter(X_train[y_train == 0][:, 0], X_train[y_train == 0][:, 1], c='blue', label='Class 0')
ax.scatter(X_train[y_train == 1][:, 0], X_train[y_train == 1][:, 1], c='red', label='Class 1')

# Plot the test point
ax.scatter(test_point[0, 0], test_point[0, 1], c='green', s=120, edgecolor='black', label='Test Point')

# Draw lines and circles to neighbors
neighbor_color = 'red' if predicted_class == 1 else 'blue'
for idx in indices[0]:
    neighbor = X_train[idx]
    ax.plot([test_point[0, 0], neighbor[0]], [test_point[0, 1], neighbor[1]], color=neighbor_color, linestyle='--')
    circle = plt.Circle(neighbor, 1, color=neighbor_color, fill=False, lw=2)
    ax.add_artist(circle)

# Labels and legend
ax.set_xlabel("Offered gap")
ax.set_ylabel("Number of vehicles in major rejected")
ax.set_title(f"{k}-Nearest Neighbors (Predicted class: {predicted_class})")
ax.legend()
ax.grid(True)

plt.show()
```

Gaussian naive bayes:

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split

# ---- Load the CSV ----
df = pd.read_csv("C:\\Users\\VICHAN\\Downloads\\Python\\data.csv")

# ---- Select numeric features and target ----
features = ["Offered gap", "Number of vehicles in major rejected"]
target = "Decision"

df = df[features + [target]].dropna()

X = df[features].values
y = df[target].values.astype(int)  # Ensure y is numeric

# ---- Train/test split ----
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# ---- Custom Gaussian Naive Bayes classifier ----
class GaussianNaiveBayes():
    def __init__(self):
        self.mean = {}
        self.std = {}
        self.prior = {}
        self.unique_classes = 0

    def get_class_parameters(self, X_class):
        return np.mean(X_class, axis=0), np.std(X_class, axis=0)

    def fit(self, X_train, y_train):
        self.unique_classes = np.unique(y_train)
        self.means_ = np.zeros((len(self.unique_classes), X_train.shape[1]), dtype=np.float64)
        self.vars_ = np.zeros((len(self.unique_classes), X_train.shape[1]), dtype=np.float64)
        self.class_priors_ = np.zeros(len(self.unique_classes), dtype=np.float64)

        for uc, c in enumerate(self.unique_classes):
            X_c = X_train[y_train == c]
            self.means_[uc, :] = X_c.mean(axis=0)
            self.vars_[uc, :] = X_c.var(axis=0)
            self.class_priors_[uc] = X_c.shape[0] / float(X_train.shape[0])
        return self
```

```
    def gaussian_density(self, x, mu, std):
        return 1 / (std * np.sqrt(2 * np.pi)) * np.exp(-(1/2) * ((x - mu) / std) ** 2)

    def predict(self, X_test):
        n_samples, _ = X_test.shape
        y_pred_prob = np.zeros((n_samples, len(self.unique_classes)), dtype=np.float32)

        for i, (mean, var) in enumerate(zip(self.means_, self.vars_)):
            likelihood = self.gaussian_density(X_test, mean, np.sqrt(var))
            total_likelihood = np.prod(likelihood, axis=1)
            prior = self.class_priors_[i]
            y_pred_prob[:, i] = total_likelihood * prior

        y_pred_prob = y_pred_prob / np.sum(y_pred_prob, axis=1, keepdims=True)
        y_pred = np.argmax(y_pred_prob, axis=1)
        return y_pred, y_pred_prob

# ---- Fit and evaluate ----
model = GaussianNaiveBayes()
model.fit(X_train, y_train)

y_pred_train, _ = model.predict(X_train)
y_pred_test, y_pred_prob_test = model.predict(X_test)

print("Train accuracy: %.2f%%" % (np.mean(y_pred_train == y_train) * 100))
print("Test accuracy: %.2f%%" % (np.mean(y_pred_test == y_test) * 100))
from sklearn.metrics import accuracy_score

# Predict on test set
y_pred_knn = knn.predict(X_test)

# Compute accuracy
knn_accuracy = accuracy_score(y_test, y_pred_knn)
print(f"KNN Test Accuracy: {knn_accuracy:.4f}")
# Print first test prediction as example
print(f"First test prediction: class {y_pred_test[0]}, probabilities: {y_pred_prob_test[0]}")

import matplotlib.pyplot as plt

# --- Create meshgrid over feature space ---
h = 0.1  # step size
x_min, x_max = X[:, 0].min() - 1, X[:, 0].max() + 1
y_min, y_max = X[:, 1].min() - 1, X[:, 1].max() + 1
xx, yy = np.meshgrid(np.arange(x_min, x_max, h),
                     np.arange(y_min, y_max, h))

grid = np.c_[xx.ravel(), yy.ravel()]

# --- Predict class for each point in mesh ---
Z, _ = model.predict(grid)
Z = Z.reshape(xx.shape)

# --- Plot ---
plt.figure(figsize=(10, 6))
plt.contourf(xx, yy, Z, alpha=0.4, cmap=plt.cm.coolwarm)

# Plot training data
plt.scatter(X_train[y_train == 0][:, 0], X_train[y_train == 0][:, 1],
            color='blue', label='Train Class 0', edgecolor='k')
plt.scatter(X_train[y_train == 1][:, 0], X_train[y_train == 1][:, 1],
            color='red', label='Train Class 1', edgecolor='k')

# Plot test data
plt.scatter(X_test[:, 0], X_test[:, 1], marker='x', s=100,
            c='black', label='Test points')
```

```
plt.xlabel("Offered gap")
plt.ylabel("Number of vehicles in major rejected")
plt.title("Gaussian Naive Bayes Decision Regions")
plt.legend()
plt.grid(True)
plt.show()
```

Support vector machine:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.svm import SVC
from sklearn.model_selection import train_test_split

# Load your data
df = pd.read_csv("data.csv")

# Pick two numeric features and the target
features = ["Offered gap", "Number of vehicles in major rejected"]
target = "Decision"

df = df[features + [target]].dropna()

X = df[features].values
y = df[target].values

# Split
X_train, X_test, y_train, y_test = train_test_split(X, y, random_state=42, test_size=0.2)

# Train SVM without scaling
svm_model = SVC(kernel='rbf', gamma='scale')
svm_model.fit(X_train, y_train)
# Predictions and accuracy
y_train_pred = svm_model.predict(X_train)
y_test_pred = svm_model.predict(X_test)

train_acc = accuracy_score(y_train, y_train_pred)
test_acc = accuracy_score(y_test, y_test_pred)

print(f"Train Accuracy: {train_acc:.3f}")
print(f"Test Accuracy: {test_acc:.3f}")

# Plot decision boundary
x_min, x_max = X[:, 0].min() - 1, X[:, 0].max() + 1
y_min, y_max = X[:, 1].min() - 1, X[:, 1].max() + 1
xx, yy = np.meshgrid(np.linspace(x_min, x_max, 500),
            np.linspace(y_min, y_max, 500))

Z = svm_model.predict(np.c_[xx.ravel(), yy.ravel()])
Z = Z.reshape(xx.shape)

plt.figure(figsize=(8, 6))
plt.contourf(xx, yy, Z, alpha=0.3, cmap=plt.cm.coolwarm)

# Plot training data
plt.scatter(X_train[y_train == 0][:, 0], X_train[y_train == 0][:, 1], c='blue', label='Class 0 (Train)')
plt.scatter(X_train[y_train == 1][:, 0], X_train[y_train == 1][:, 1], c='red', label='Class 1 (Train)')

# Test data points
plt.scatter(X_test[:, 0], X_test[:, 1], color='black', marker='x', label='Test points')

plt.xlabel("Offered gap")
plt.ylabel("Number of vehicles in major rejected")
plt.title(f"SVM Without Scaling\nTrain Acc: {train_acc:.3f} | Test Acc: {test_acc:.3f}")
```

```
plt.legend()
plt.grid(True)
plt.show()
```

Accuracy comparison:

```
import matplotlib.pyplot as plt

# Example accuracies (replace with your actual results)
models = ["Random Forest", "KNN", "SVM", "Naive Bayes"]
train_accuracies = [1.0, 0.9378, 0.935, 0.9043]  # Replace with actual train acc
test_accuracies = [0.9351,  0.9008, 0.893, .8740]  # Replace with actual test acc

x = np.arange(len(models))
width = 0.35

fig, ax = plt.subplots(figsize=(8, 6))
bars1 = ax.bar(x - width/2, train_accuracies, width, label='Train Accuracy')
bars2 = ax.bar(x + width/2, test_accuracies, width, label='Test Accuracy')

ax.set_ylabel('Accuracy')
ax.set_title('Model Accuracy Comparison')
ax.set_xticks(x)
ax.set_xticklabels(models)
ax.set_ylim(0, 1.1)
ax.legend()
ax.bar_label(bars1, fmt='%.2f', padding=3)
ax.bar_label(bars2, fmt='%.2f', padding=3)

plt.grid(True, linestyle='--', alpha=0.6)
plt.tight_layout()
plt.show()
```