



CHALMERS
UNIVERSITY OF TECHNOLOGY



Kollisionsvarningssystem för spårvagn

Varningssystem baserat på radardetektioner

Examensarbete inom Data- och Informationsteknik

BETINA ANDERSSON
EMMA RAUDBERGET

EXAMENSARBETE

Kollisionsvarningssystem för spårvagn

Varningssystem baserat på radardetektioner

BETINA ANDERSSON
EMMA RAUDBERGET



Institutionen för Data- och Informationsteknik
CHALMERS TEKNISKA HÖGSKOLA
GÖTEBORGS UNIVERSITET

Göteborg, Sverige 2018

Kollisionsvarningssystem för spårvagn
Varningssystem baserat på radardetektioner
BETINA ANDERSSON
EMMA RAUDBERGET

© BETINA ANDERSSON, EMMA RAUDBERGET, 2018.

Examinator: Peter Lundin, Institutionen för Data- och Informationsteknik

Institutionen för Data- och Informationsteknik
Chalmers Tekniska Högskola/ Göteborgs Universitet
412 96 Göteborg
Telefon: 031-772 1000

The Author grants to Chalmers University of Technology and University of Gothenburg the non-exclusive right to publish the Work electronically and in a non-commercial purpose make it accessible on the Internet. The Author warrants that he/she is the author to the Work, and warrants that the Work does not contain text, pictures or other material that violates copyright law.

The Author shall, when transferring the rights of the Work to a third party (for example a publisher or a company), acknowledge the third party about this agreement. If the Author has signed a copyright agreement with a third party regarding the Work, the Author warrants hereby that he/she has obtained any necessary permission from this third party to let Chalmers University of Technology and University of Gothenburg store the Work electronically and make it accessible on the Internet.

Omslag:
Spårvagn i Göteborg. Foto: Göteborgs Spårvägar

Institutionen för Data- och Informationsteknik
Göteborg 2018

Sammanfattning

I dagens spårvagnstrafik sker årligen olyckor som både kostar pengar och leder till att människor får uppsöka sjukhusvård. Projektets syfte är att undersöka vilka olyckor som kan förhindras med ett kollisionsvarningssystem, samt ta fram en prototyp på ett sådant system. Arbetet genomförs hos företaget Aptiv.

Olycksstatistik har granskats och en analys av olyckstyperna och spårvagnstrafiken är genomförd. Spårvagnstrafiken skiljer sig från övrig trafik då den rör sig långsammare och att det är kortare avstånd till föremål runt omkring spårvagnen. De flesta olyckorna beror på den mänskliga faktorn.

Varningssystemet som har utvecklats med hjälp av spårvagnsloggar fungerar som så att två radarsensorer uppfattar rörliga och fasta hinder. Datan från radarn behandlas sedan i ett program som analyserar tid till kollision. Systemet varnar genom att tända olika färgade lampor baserat på hur tidskritiskt läget är.

Hur väl systemet varnar är ännu inte utrett. Varningarna är testade på spårvagnens loggar varvid de antas varna vid rätt tillfällen. Det har även genomförts varningstester där en lastbil använts istället för en spårvagn, även här anses varningarna ske korrekt, men detta bör undersökas vidare.

Fördelar med systemet är att kollisioner som beror på ouppmärksamma förare kan förhindras. Förslag för vidareutveckling är adaptiv farthållare, så att spårvagnen automatiskt kan reglera hastigheten utifrån fordon eller hinder framför vagnen, samt en undersökning kring förarens interaktion med varningarna.

Nyckelord: spårvagn, kollisionsvarningssystem, olycksstatistik, SRR, TTC.

Abstract

Tram traffic today has accidents that each year cost money and causes people to seek medical attention. The purpose of this project is to conduct which accidents can be prevented with a forward collision warning system, as well as create a warning system prototype. The work will be carried out at the company Aptiv.

A review of the accident statistics and an analysis of the accident types and overall tram traffic has been completed. Tram traffic differs from other traffic since it moves at a slower pace and there is shorter distance to objects around the tram. Most accidents happen due to human error.

The warning system has been developed with the help of tram logs. There are two radar sensors who detect moving objects and stationary obstacles. Their data is analysed in a program that calculates time to collision. The system will warn the tram driver by lightning different colored lamps depending on how time critical the situation is.

How well the system works is yet to be investigated. The warning functions have been tested on the tram logs and seem to warn at the right time. The warnings have also been tested on a truck instead of a tram, and here they also appear to warn correctly, but this needs further investigations.

A perk with this system is that collisions caused by inattentive drivers can be prevented. A suggestion for further development is to implement adaptive cruise control, so that the tram automatically can regulate its speed regarding to other vehicles or obstacles in front of the tram, as well as conduct a study on how the tram driver interact with the warnings.

Keywords: tram, forward collision warning system, accident statistics, SRR, TTC.

Förord

I denna rapport beskrivs hur ett kollisionsvarningssystem för en spårvagn konstrueras. Arbetet är ett examensarbete på Chalmers för institutionen för Data- och Informationsteknik. Arbetet har utförts hos Aptiv på avdelningen för elektronik och säkerhet. Även Göteborgs Spårvägar har bidragit med stöd och kompetens gällande spårvagnar.

Vi vill ge ett stort tack till de personer som hjälpt oss med projektet. Ert engagemang har varit oerhört uppskattat.

Aptiv

Martin Larsson

Mattias Merking

Andreas Andersson

Chalmers Tekniska Högskola

Erland Holmström

Göteborgs Spårvägar

Krste Cvetkovski

Patrik Alm

Betina Andersson, Göteborg, Maj 2018

Emma Raudberget, Göteborg, Maj 2018

Innehållsförteckning

1	Inledning	1
1.1	Avgränsningar	2
2	Teori	3
2.1	Spårvagnar	3
2.2	Stoppsträcka	3
2.3	Hårdvara	4
2.4	Mjukvara	7
3	Analys	9
3.1	Olyckor	9
3.2	Implementering	10
3.3	Funktionsutveckling och insamling av data	12
4	Genomförande	13
4.1	Utveckling av analyseringsverktyg	13
4.2	Konstruktion av hårdvara	15
4.3	Utveckling av varningssystemet	18
4.4	Test på spårvagn	21
4.4.1	Genomgång av spårvagnsloggar	24
4.5	Arbetsmetod	25
5	Resultat	27
6	Diskussion	29
6.1	Hållbarhet och etik	30
6.2	Förslag på vidareutveckling	31
6.3	Slutsats	31
	Referenser	35
A	Olycksstatistik	I
B	Bilder från spårvagnstest	III

1

Inledning

Flera av spårvagnsolyckorna i Göteborg leder till att personer inblandade i olyckan får uppsöka sjukhusvård och tyvärr sker även dödsfall. En majoritet av dessa olyckor där människor blir skadade är resultatet av den mänskliga faktorn, exempelvis en trött och stressad spårvagnsförare eller ouppmärksamma medtrafikanter. Göteborgs Spårvägar har tidigare undersökt möjligheten att automatisera och skapa varningssystem för spårvagnarna likt de som finns i moderna bilar men utan framgång och har benämnt det som en omöjlighet [24]. Automationsföretaget Aptiv ser dock stor potential i att automatisera spårvagnstrafiken och vill genom detta minska antalet olyckor. Med hjälp av sensorer skall ett varningssystem för föraren utvecklas. Detta system skall sedan testas för att avgöra om spårvagnar överbevisligen kan automatiseras.

Bakgrund

Den 17 mars 2015 publicerades en artikel i Göteborgs-Posten att Göteborgs Spårvägar har genomfört tester med att implementera ett automatiskt kollisionsvarningssystem i två spårvagnar. Systemet som testades var samma som används i moderna bilar och slutsatsen var att kollisionsvarningssystemet inte fungerade alls enligt dåvarande trafikchef på Göteborgs Spårvägar [24]. Trafikchefen nämnde även att en implementation av samma automatiska nödbromssystem som finns i bilar skulle vara omöjligt att göra i en spårvagn.

Spårtrafiken bör dock ha stor möjlighet att automatiseras på sikt då spårvägarna har en precis bana att utgå ifrån. Aptiv är ett företag som verkar inom automatisering av fordon. Nu ska påståendet om att det skulle vara omöjligt att automatisera en spårvagn undersökas.

Syfte och mål

I syfte att försöka minska olycksstatistiken ska ett kollisionsvarningssystem för föraren utvecklas med avseende att varna för hinder runt spårvagnen. Syftet är även att innan utvecklingen startar söka och sammanställa information om olycksstatistik för spårtrafiken i Göteborg.

Målet är att utveckla ett varningssystem, som varnar spårvagnsföraren när det finns risk för kollision. Systemet ska även ge en hänvisning kring när en automatisk nödbroms bör aktiveras och när allmänheten runt om spårvagnen ska varnas. Det ska också vara väl anpassat till spårvagnstrafiken gällande varningstider och montering, och interaktionen mellan spårvagn och människa ska kännas väl genomtänkt. Detta

ska vara ett första steg mot självkörande spårvagnar.

Frågeställning

- Hur många olyckor och vilken typ av olyckor skulle kunna förhindras med ett system som detta?
- Hur kan ett varningssystem för spårvagnsföraren utformas och implementeras?

1.1 Avgränsningar

Sensorerna som används är SRR3T-sensorer från Aptiv [9]. Anledningen till att radar används är att de är hållbara och fungerar i alla väder, vilket är optimalt för en spårvagn. Eftersom endast ett varningssystem ska utvecklas är radarsensorer tillräckliga för projektet.

Funktionerna kommer skrivas i MATLAB enligt företagets önskningsar. MATLAB är ett användbart verktyg för att kunna analysera loggar och skriva funktioner för att beräkna när kollisioner mellan fordon ska inträffa [20]. MATLAB har även en inbyggd funktion för att omvandla koden till C++, vilket används för att kunna kommunicera med radar-sensorerna. C++ används även för att programmera Arduino Nano och Raspberry Pi modell 3.

Då tillgången till spårvagnar begränsar detta examensarbete kommer endast funktioner utvecklas för en typ av spårvagn, spårvagnsmodellen M32. Funktionerna kommer att testas i efterhand på data insamlad från ett tillfälle då sensorer och utrustning satt monterad på en spårvagn. Datan baserar sig på flertalet testscenarion. Valet av testscenarion baseras på de situationer som orsakar verkliga olyckor såsom bil som svänger framför spårvagn, krock med spårvagn på samma spår men även vid de fall då personer passerar spåren. När en spårvagn ej finns tillgänglig kommer istället en lastbil användas för att samla in data.

Spårvagnstesterna kommer inte att utföras i trafik, utan på ett depåområde. Spårvagnsdata såsom hastighet och girvinkelhastighet kommer inte att kunna erhållas från vagnen. Därför måste externa sensorer användas för att få dessa parametrar.

2

Teori

Här beskrivs den teori och tekniska kunskap som har samlats in och legat som grund för systemets konstruktion.

2.1 Spårvagnar

Spårvagnarna används idag i åtta av Sveriges kommuner, och år 2020 kommer de bli nio stycken [1]. De är dessutom energieffektiva, bland annat på grund av att de har stålhjul vilket har lägre friktion än gummihjul [2]. I Göteborg kör spårvagnen med en medelhastighet på 10-15 km/h i innerstaden. Den högsta tillåtna hastigheten för en spårvagn är 60 km/h, vilket är en hastighet den kan komma upp i när den kör på egen banvall.

Spårvagnen kan gasa och bromsa. Om föraren vill göra ett växelbyte föranmäler föraren detta några meter innan bytet. Detta görs genom en knapptryckning varpå spårvagnen kommunicerar med banans system så att växelbytet sker [3]. Spårvagnen har effektiva bromsar för att kunna stanna snabbt i stadsmiljö. Vid nödsituationer kan en magnetskenbroms användas, annars används normal driftbroms som har en bromsprestanda på 1.4m/s^2 [4]. Magnetskenbroms effektiviserar inbromsningen med 25-30 procent jämfört med normal driftbroms [5]. För några år sedan byttes tidigare bromssystem ut på vissa av Göteborgs spårvagnar, dessa bromssystem var 30 år gamla. Spårvagnarna har nu en bromslåda som reglerar luftmängden som håller bromsarnas avstånd från rälsen. När spårvagnsföraren vill bromsa kommer signaler skickas till styrdatorn som reglerar lufttrycket [6].

2.2 Stoppsträcka

Stoppsträckan är summan av bromssträckan och reaktionssträckan. Reaktionstiden ligger normalt mellan 0.4-2.7 sekunder, den vanligaste reaktionstiden är en sekund [7]. Formeln för reaktionssträckan är $s = vt$, där s är reaktionssträckan, v är fordonets hastighet och t är förarens reaktionstid [8].

Bromssträckan beräknas med formeln $s = v^2/2a$, där s är bromssträckan, v är fordonets hastighet, och a är fordonets retardation. Spårvagnens retardation är som nämnt 1.4m/s^2 när normal driftbroms används. Retardationen kan minska vid halt underlag, exempelvis lövhalka är problematiskt för spårvagnar och kan förlänga

bromssträckan [4]. Retardationen kan även öka vid användning av magnetskenbroms.

Stoppträckan kan slutligen beräknas som reaktionssträckan + bromssträckan, eller från grunden med formeln $s = vt + v^2/2a$ [8].

2.3 Hårdvara

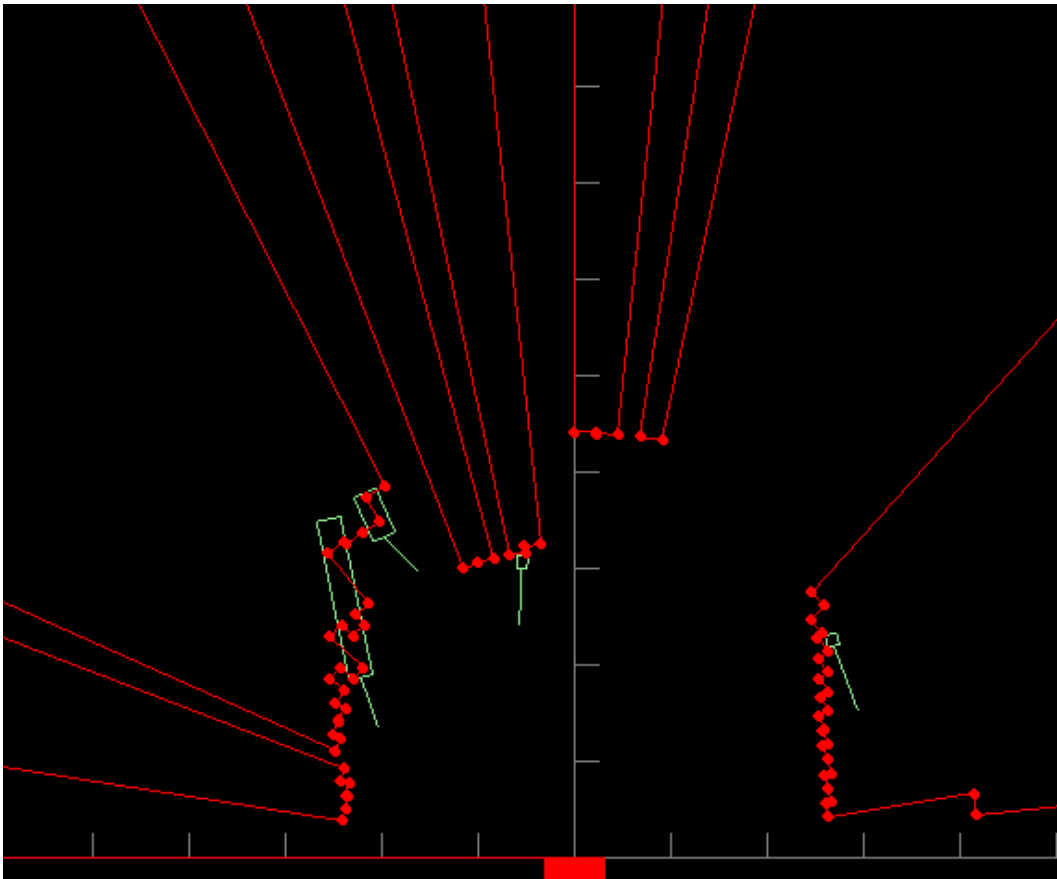
För att kunna konstruera ett kollisionsvarningssystem krävs det sensorer och detta systems huvudsensorer är radarsensorerna SRR3T. Radarsensorerna kräver i sin tur indata i form av spårvagnens hastighet och girvinkelhastighet för att uppfatta verkligheten korrekt.

Radarsensor SRR3T

SRR3T är en radarsensor där SRR står för "Short Range Radar" och i moderna bilar är det vanligt att SRR-sensorer används [9]. Radarteknik fungerar genom att radiovågor skickas ut till föremål vars reflektioner och ekon tas emot av radarn.

SRR3T-sensorn innehåller mjukvara som både kan detektera föremål som är stilla eller i rörelse. Detekteringen av föremål i rörelse kallas "Object Tracker" (objekt) och för stillastående föremål "Freospace" (freespace) [10]. De rörliga hindren är alltså objekt, och det klassas som rörligt om hindret rör sig över cirka 2m/s. Objekten har riktning, hastighet och position och radarn kan hantera flera olika objekt samtidigt. Just SRR3T kan hantera tio objekt åt gången. Freespace kan användas för att upptäcka de hinder som inte rör sig. Det görs genom att SRR3T har 60 sektorer som stängs av på ett visst avstånd där det finns ett hinder. Dessa sektorer täcker hela radarns synfält. Se figur 2.1 för exempel på objekt och freespace.

Radarsensorerna behöver veta vilken position och vinkel de är monterade med, samt mått på fordonet de är monterade på. Monteringsinformationen erhåller radarn genom att den programmeras.

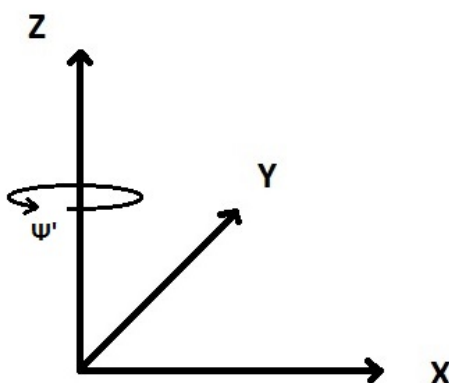


Figur 2.1: *Freespace och objekt. De röda strecken symboliserar de 60 sektorer som stängs av där ett hinder upptäcks. Rörliga objekt visas som de gröna fyrkanterna.*

Fördelar med radar är att funktionaliteten ej begränsas av diverse väderförhållanden såsom snö, dimma och regn. Dessutom är den billigare än många andra system. Något som begränsar radarns funktionalitet är att den inte kan urskilja färger. Detta skulle exempelvis kunna försvåra vid trafikljus eller vägskyltar om systemet behöver kunna tolka deras betydelser [11]. Tidigare undersökningar visar att radar är en effektiv lösning för självkörande fordon, och att den är ett viktigt komplement till kamerasystem [12].

Gyroskop

Med hjälp av en gyroskopsensor kan girvinkelhastigheten (yaw rate) utläsas. I detta projekt används sensorenheten MPU9250 som består av en accelerometer, ett gyroskop och en magnetometer varvid endast gyroskopet används. MPU9250 har ett treaxligt gyroskop och med detta kan spårvagnens girvinkelhastighet erhållas. Girvinkelhastigheten, benämnt "yaw rate" i fordonsbranschen, definieras som vinkelhastigheten kring den vertikala z-axeln [8]. Detta visas i figur 2.2.



Figur 2.2: Bild som illustrerar hur yaw rate mäts. Z-axeln är den axel som går upp från gravitationspunkten. Yaw rate mäts genom att kolla med vilken vinkelhastighet fordonet rör sig runt denna axel.

Rådatan från gyroskopets z-axel har ett förbestämt intervall på $[-1000, 1000]$ [13]. För att radarn skall kunna tolka värdena korrekt måste värdena filtreras och göras proportionerliga [14].

Hastighet med optisk läsgaffel

Enligt tidigare kräver radarn även indata i form av hastighet. Hastigheten i detta projekt mäts med hjälp av en optisk läsgaffel. Läsgaffeln har en givare som skickar en puls varje gång ljus blockeras från att nå en av gaffelns sidor [15]. Genom att kombinera läsgaffeln och en enkoderskiva, en cirkulär skiva med symmetriska hål i, kan hastighet erhållas. Detta görs genom att enkoderskivan sätts fast på en axel. På denna axel sitter även ett fixt hjul. Enkoderskivan snurrar inuti läsgaffeln och släpper in och blockerar ljus med sina hål. När hjulet och axeln roterar erhålls antal varv per tidsintervall. Hjulets hastighet fås då som hjulets omkrets multiplicerat med antal varv per tidsenhet [16].

CAN

CAN är en metod som främst används när fordon ska kommunicera med varandra. Det är en förkortning för "Controller Area Network" [17]. CAN används när det finns krav på snabb kommunikation, exempelvis vid motorstyrning och bromssystem. Kommunikationen sker genom att det skickas meddelanden på en databuss. Meddelandena består av data-bitar, och skickas asynkront. Varje meddelande består av en identifierare samt datan som ska analyseras. Själva databussen består av kabelpar, en "high" och en "low", som kopplats från en nod till en annan. Noderna är exempelvis en mikrodata.

Arduino Nano

Arduino Nano är ett mikrokontrollerkort med stöd för att koppla in annan elektronisk hårdvara. Den fungerar ungefär som en mikrodata. Arduino Nano är baserad på mikrokontrollern ATmega328 och har både analoga och digitala in- och utgångar [18]. Arduinon saknar ett operativsystem men kan fortfarande köra

kompilerad kod, detta kan vara fördelaktigt när ett operativsystem anses vara överflödigt. En Arduino är också generellt enklare att använda och passar bra för att hantera data från sensorer i realtid. En nackdel med Arduino i jämförelse med en Raspberry Pi är att den inte är lika kraftfull [19].

2.4 Mjukvara

Mjukvara är essentiellt i utvecklingen av kollisionsvarningssystemet. I detta projekt används MATLAB, Arduino IDE och Orcas. I MATLAB sker funktionsutvecklingen, med Arduino programmeras gyroskopet och läsgaffeln, och Orcas används för att logga data. Här nedan presenteras mjukvarorna något djupare.

MATLAB

MATLAB är en programplattform där det bland annat går att utveckla algoritmer, bearbeta signaler, utföra matematiska beräkningar och analysera data. Plattformens programmeringsspråk är MATLAB och dessutom kan verktyg för specifika områden läggas till. Ett exempel är stödet för att omvandla MATLAB-kod till C++, vilket är användbart för implementering av funktioner i en mikrodator [20].

Arduino IDE

I Arduino IDE används ett programspråk som påminner om C++ och det finns möjlighet att skriva kod och lägga till bibliotek [18]. Dock finns inget stöd för att redigera biblioteksfiler direkt i Arduinos utvecklingsmiljö, utan detta måste göras med ett annat program. Arduino bygger på öppen källkod och det finns mycket kod och många bibliotek på internet. Programmeringsmiljön tillhandahar även möjligheten att kompilera kod och visa seriell data från sensorer i realtid.

Orcas

Orcas är en programvara som främst används internt på Aptiv. Med Orcas loggas information från sensorer och webbkamera under tester. Insamlad data kan analyseras efter testerna med både ett visuellt användargränssnitt och genom att titta på den insamlade informationen. De loggfiler som samlats in kan sedan exporteras till MATLAB för att bistå funktionsutvecklingen eller för vidare analysering.

3

Analys

Att konstruera ett kollisionsvarningssystem till en spårvagn är en utmaning. Därför delas analysen upp i flera delområden som analyseras separat och presenteras här nedanför. Sammanfattningsvis kan det nämnas att det är en komplex huvuduppgift med många faktorer att ta hänsyn till.

3.1 Olyckor

Det sker årligen hundratals olyckor med spårvagnar i Göteborg och årligen är det ett hundratal passagerare som får uppsöka sjukvård till följd av en spårvagnsolycka. Den vanligaste olyckstypen är att spårvagnarna krockar med andra fordon. Dessa olyckor beror till största del på att bilister inte alltid följer trafikregler, exempelvis när de sneddat över vagnspåren [21]. På senare år har runt 10-20 personer varje år fått allvarligare skador till följd av en spårvagnsolycka, och därmed behövt vårdas på sjukhus i över 24 timmar [22]. Redan under de första månaderna av 2018 har det inkommit flera rapporteringar i media kring just spårvagnsolyckor. Bland annat inträffade en olycka vid Frölunda i februari då en spårvagn körde in i en framförvarande vagn efter att föraren missuppfattat att spårvagnen framför stått stilla. Som följd av olyckan fick nio passagerare uppsöka sjukvård. I de allra flesta olyckor där människor kommer till skada sker skadorna inne i vagnen. Ofta sker dessa olyckor i samband med att vagnen bromsar hastigt [3]. I Göteborg sker de allra flesta olyckorna vid Centralstationen. För ytterligare statistik, se bilaga A.

Anledningen till olyckan i Frölunda var att föraren var stressad. Föraren beskriver hur denne nyligen tagit över vagnen, var sen till hållplatsen, och på grund av stress kollade i tidtabellen istället för på vägen. På grund av detta hann föraren inte reagera över att spårvagnen framför stod stilla, utan trodde istället bara att denna körde långsamt [23]. Krocken berodde således på den mänskliga faktorn, vilket också är anledningen till majoriteten av alla spårvagnsolyckor i Göteborg. Bland annat till följd av att spårvagnsförarna är trötta eller stressade, eller att förare i kringtrafiken felar och exempelvis svänger ut framför spårvagnarna [21],[24]. En olycka likt den i Frölunda kostar miljoner att åtgärda, och olyckor där spårvagnen måste tas in och undersökas sker näst intill dagligen. Det kostar även samhället generellt mycket pengar eftersom vägar ibland måste stängas av och liknande. Göteborgs Spårvägar har till följd av olyckan i Frölunda börjat fundera över huruvida självkörande spårvagnar skulle kunna förhindra dessa olyckor [3].

Som förslag på åtgärder för att reducera antalet olyckor behöver något göras för att föraren skall återfå sin uppmärksamhet, inte är stressad eller ett system som nödbromsar vagnen utifall föraren inte gör det i rätt tid. Ett annat förslag är att enbart låta fotgängare få passera spåren på utvalda ställen.

Alternativet att endast låta fotgängare passera spåren på utvalda områden vid de mest olycksdrabbade sträckorna är ett alternativ som kräver ombyggnationer, analyser av de nya trafikflödena och en utredning ifall det hade kunnat minska olycksstatistiken. Det berör även andra faktorer inom bland annat stadsplanering som inte vi som utvecklare kan rå över. Stressade förare är inte heller något som kan åtgärdas inom detta examensarbete. Men det går att konstruera ett system som gör att föraren återfår sin uppmärksamhet vilket borde kunna reducera antalet olyckor.

3.2 Implementering

2013 testades ett framåtriktat kollisionsvarningssystem på spårvagnar och då användes en Mobile-Eye-kamera från ett Volvo Cars-projekt. Några av svårigheterna då var att tekniken inte hade kommit lika långt som i dagsläget, och att systemet inte var anpassat för spårvagnarna. De problem som upptäcktes under detta test var att systemet varnade för kontaktledningsstolparna och mötande spårvagnar eftersom vagnarna kör närmare varandra än vad bilar kör. Det uppstod även många varningar för passagerare som stod nära vagnarna. Detta då människor vanligtvis uppehåller sig väldigt nära vagnarna. För att påvisa hur nära människor rör sig vagnarna förklarade Göteborgs Spårvägar att de haft flera fall då personer på hållplatserna fått spårvagnens sidospelar i huvudet. Ett ytterligare problem var att varningssystemet var kamerabaserat. Då det inte är tillåtet att spara bilder på civila personers ansikten blev det svårt att analysera datan efteråt [3][4].

Utvecklarna av det system som testades 2015 insåg under projektets gång att det tar lång tid att få tillstånd för att göra modifikationer på spårvagnarna, både för att fästa saker på vagnen samt att få ta ut den med utrustning i trafik. När tillstånd äntligen kom var det nästan dags att avsluta projektet [25]. På grund av tidspressen ville utvecklarna maximera känsligheten på sitt system för att kunna samla in så mycket data som möjligt. Detta gav naturligtvis många falska varningar, vilket var något som dåvarande trafiksäkerhetschefen från Göteborgs Spårvägar belyste i en diskussion i Göteborgs-Posten [24].

För det kollisionsvarningssystem som skall utvecklas under detta examensarbete återstår några problem. Systemet behöver indata i form av hastighet och girvinkelhastighet för att erhålla tillförlitliga radardetektioner. Tyvärr klassas det som en tillståndskrävande modifikation att koppla in radarn till spårvagnens egna hastighetsgivare. Inom tidsramen för detta projekt finns ingen möjlighet att erhålla tillstånd från Göteborgs Spårvägar för att få använda spårvagnens hastighet. Spårvagnen har inte heller någon information om girvinkelhastigheten. Det krävs även tillstånd för att fästa material eller för att ändra spårvagnarnas utseende. Då inget får sticka ut

från spårvagnen finns ingen möjlighet att köra spårvagnen i trafik med sensorerna monterade på vagnen. Dessutom förutsätts det att systemets hårdvara är enkel att montera på och av och endast behöver ett vanligt eluttag för att inte göra någon permanent åverkan på spårvagnen som kräver tillstånd.

Däremot kan tillståndsproblematiken kringås genom att endast genomföra testerna på ett spårvagnsdepåområde. Fördelen med ett depåområde är att det är en kontrollerad miljö där inga personer från allmänheten vistas. För att undvika att modifiera spårvagnen kan vi använda externa hjälpsystem för att läsa av spårvagnens hastighet och yaw rate. Detta under förutsättning att hjälpssystemen skickar datan till kollisionsvarningssystemet med rätt intervall. Hastigheten kan fås ut genom att nyttja GPS-teknik, cykeldator och med sensorer som kan detektera rotationshastighet för spårvagnshjulet såsom enkoderskivor med avläsare, hallsensorer och tachometer.

Då testerna på spårvagn sker både inomhus och utomhus förkastas idén om att använda GPS-teknik då tillförlitligheten inte är tillräckligt hög inomhus med en billig GPS-sensor utan en markbaserad GPS-basstation. Dessutom avfärdas förslaget att använda en färdig cykeldator, då det är för tidskrävande att programmera en mikrokontroller att skicka datan från cykeldatorn till systemet med rätt tidsintervall, signaler och kalibreringsmöjligheter. Dock är det inte enkelt att montera av och på sensorer vid spårvagnens hjul då det råder platsbrist. Istället kan en roterande axel vid spårvagnens boggi användas för erhålla hastighetsdata genom att nyttja det faktum att axeln roterar. Även kring den roterande axeln finns ett begränsat utrymme.

Att använda hallsensorer förkastas då det blir svårt att placera ut magneterna på exakt samma avstånd från varandra på den roterande axeln. För att få hög upplösning krävs flera magneter vilket försvårar montering och placering.

Ett annat alternativ för att få ut spårvagnens hastighet är att använda ett mindre hjul som kan ligga emot spårvagnens roterande axel. Hjulet behöver utformas på ett sådant sätt att enkoderskivan kan kopplas till hjulet och rotera med samma hastighet som hjulet inuti den optiska läsgaffeln. Därmed kan spårvagnens hastighet, efter viss kalibrering, erhållas. Anledningen till att det krävs ett separat hjul som ligger emot spårvagnens axel är för att enkoderskivan är väldigt smal och liten och går sönder om den skulle läggas direkt mot axeln. Lösningen med ett separat hjul som har möjlighet att rotera enkoderskivan inuti läsgaffeln är alltså en stabilare lösning.

Som beskrivet i inledningen är ett av målen med kollisionsvarningssystemet att indikera för automatisk nödbroms. Dock kommer inte nödbromsen att implementeras inom ramen för examensarbetet då varken tid, tillstånd eller resurser finns.

3.3 Funktionsutveckling och insamling av data

Det färdiga kollisionvarningssystemet förutsätts att ha tre huvudfunktioner baserat på tiden kvar till kollision. Dessa tre är följande: en lampa som motsvarar en signal som varnar personer och fordon runt spårvagnen, en lampa som motsvarar lampor och ljud inne i förarhytten för att varna föraren och slutligen en lampa som indikerar att den automatiska nödbromsen har aktiverats. Lampan som varnar personer och fordon runt spårvagnen är den funktion som skall aktiveras med längst tid kvar till kollision och lampfunktionen för automatisk nödbroms är den som skall aktiveras då det är kortast tid kvar. Dessutom används färdiga funktioner för att detektera föremål och hinder med hjälp av radarmjukvarorna "Object Tracker" och "Freespace".

I Göteborg finns flera typer av spårvagnar vars egenskaper skiljer sig åt. Detta försvårar funktionsutvecklingen då spårvagnstyperna har olika överhäng, rotationcentrum, datakommunikation samt åtskilda fästmöjligheter för sensorer. Därför görs bedömningen att endast utveckla funktioner för en typ av spårvagn för att kvarstanna inom projektets tidsram.

Ett steg i funktionsutvecklingen är att verifiera funktionerna genom att testa dessa. Tester kan utformas på flera sätt, exempelvis genom att implementera funktionerna och i realtid se hur funktionerna svarar på inkommande sensordata, eller genom att samla sensordata och i efterhand mata funktionerna med datan, eller att skicka konstruerad data till funktionerna. Oavsett om funktionerna testas i realtid, i efterhand eller med hjälp av konstruerad data bör testscenarion konstrueras. Testscenariernas syfte är att återspegla verkligheten i den mån det går. Lämpliga testscenarion kan vara sådana som anses vara viktiga fall för systemet att hantera på önskat sätt eller sådana fall som ofta sker i verkligheten. Att välja vilka och mängden testfall kan vara svårt och det finns många parametrar att ta hänsyn till. För spårvagnstrafik föreslås följande testscenarion: Spårvagn som kör in på ett hållplatsläge där det står några personer, varav några går över precis innan spårvagnen anländer till plattformen, bil som svänger precis framför spårvagn, passering av mötande spårvagn, körning bakom en annan spårvagn och personer som rör sig kring en spårvagn i rörelse.

4

Genomförande

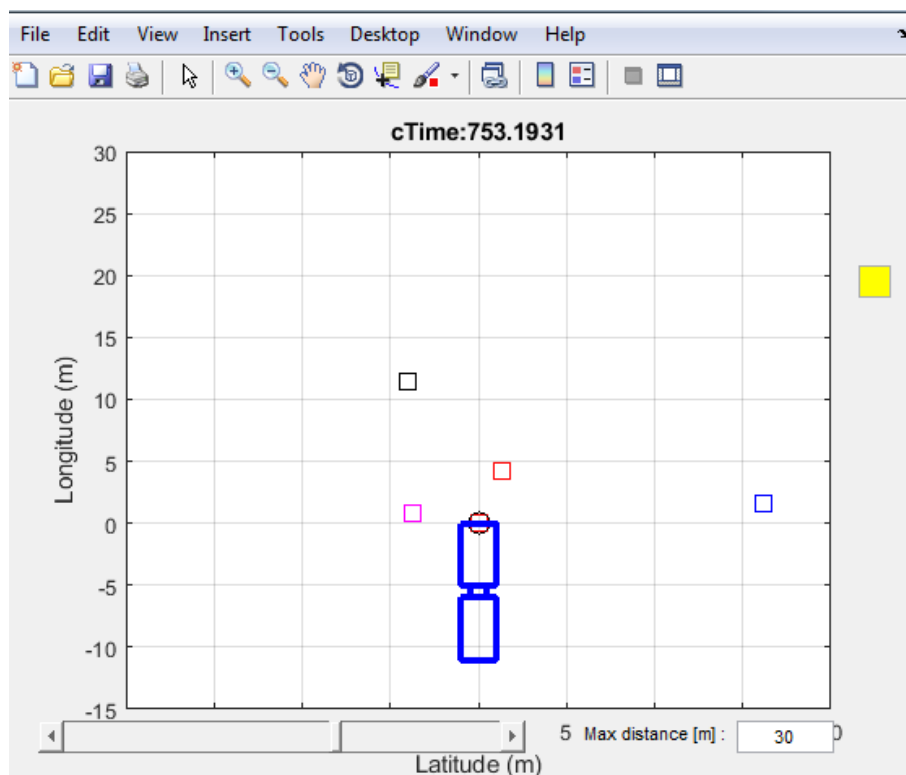
I detta kapitel beskrivs projektets arbetsprocess. Det som redogörs för är beslut som fattats, metoder för att verifiera det som utvecklats, problem som uppstått samt lösningar på dessa.

4.1 Utveckling av analyseringsverktyg

För att vi som utvecklare av varningssystemet skulle kunna tolka hinder framför spårvagnen (objekt) vidareutvecklade vi en funktion från ett projekt hos Aptiv för att kunna rita in objekt i MATLAB. Objektens latitudinella samt longitudinella position markeras med en fyrkant och spårvagnen befinner sig i origo. Det upptäcktes då att objektens latitudinella position sparas spegelvänt från verkligheten. Detta var tacksamt att få reda på innan funktionsutvecklingen startade, och det kunde enkelt identifieras genom detta analyseringsverktyget.

Med hjälp av en "slider" skissar vi objektens rörelse över tid i ett fönster med koordinatsystem. All data för objekten hämtas från tidigare loggar som samlats in med antingen lastbil eller spårvagn. Detta verktyg används som ett visuellt hjälpmedel för att kunna göra beräkningar på riskzoner och "time to collision" (TTC). Vi har även implementerat en ruta föreställande en lampa som visar varningsfärger när slidern rör sig. Färgerna är baserade på TTC. Vid $TTC = 1$ sekund bedöms läget som kritiskt och därmed visas en röd lampa upp. Detta för att kunna analysera och tolka om varningarna sker i rätt tid. Se figur 4.1. För att beräkna TTC används dock olika funktioner då de kräver olika ageranden från föraren, dessa situationer och funktioner beskrivs utförligt i ett senare avsnitt.

Viktigt att komma ihåg vid analys av objekten i detta verktyg är att rutorna ej är skalenliga. Ibland kan objektet egentligen vara mindre eller större än den fyrkant som ritats upp, vilket också upptäcktes när funktionerna testades. Objektet har dock en bredd och längd som skulle kunna användas för att få dem skalenliga, men detta är ännu inte implementerat.



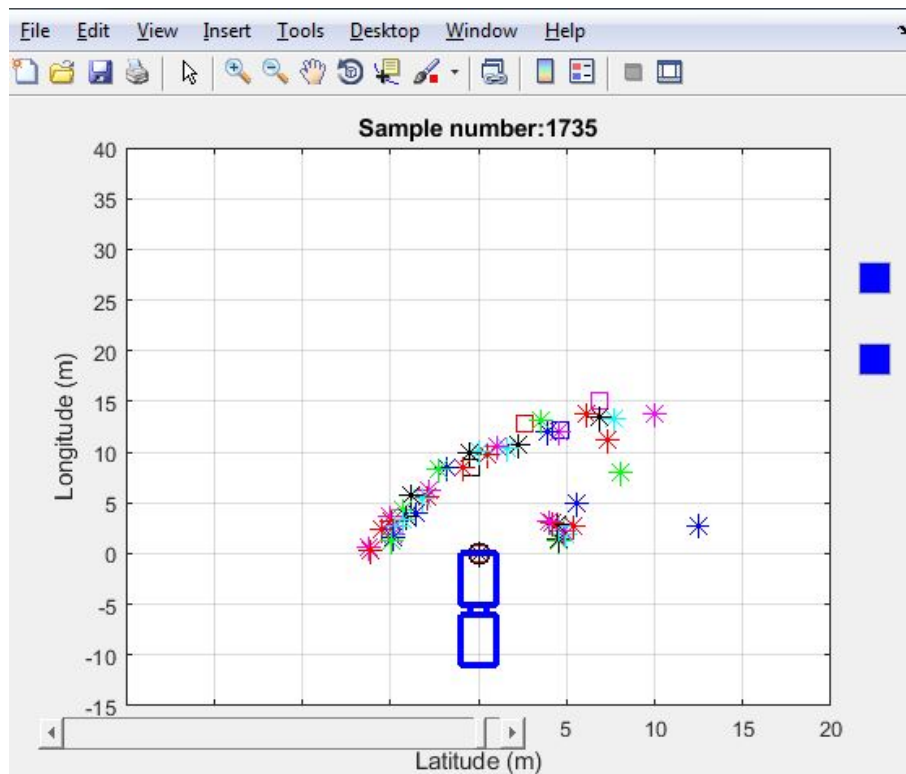
Figur 4.1: Visuellt hjälpmedel för att analysera objektets rörelse över tid. Med slidern ändras tidsampel och objekten (de mindre fyrkanterna) kommer förflyttas enligt datan för deras positioner.

Något annat som kan vara bra att komma ihåg är att hastigheten samplas var 20e millisekund medan objekten uppdateras var 50e millisekund. Detta gör att objektens tidsstämplat inte stämmer överens med hastigheten om man endast jämför sampsnummer. För att åtgärda detta ska varje samps index i objekten som undersöks multipliceras med ett värde. Detta värde måste sedan rundas upp eftersom alla index är heltal. Vid ännu närmre kontroller upptäcktes det att hastigheten samplats ungefär 2.15 gånger oftare än objekten hunnit uppdateras. Detta är inte kvoten mellan 20 och 50, men det var den datan som fanns tillhanda. Anledningen är ännu okänd, men vid användning av analyseringsverktyget multiplicerades indexet därför med antalet hastighetssampel genom antalet objektuppdateringar.

Efter en viss tids utveckling kunde även freespace skissas in i analyseringsverktyget. Freespace ritades in som stjärnor där sektorn blockerats. När radarn inte hittar en blockering sätts blockeringen istället på 300 meters avstånd, dessa stjärnor syntes då inte i analyseringsverktyget. För att kontrollera att systemet kunde varna med freespace skapades en funktion baserad på longitudinell och lateral position för sektorblockeringen. Vid varje tidssampel kollar funktionen alla 60 sektorernas koordinater. Dessa koordinater kunde sedan skickas in i samma analyseringsverktyg som för objekt, men där blockeringar av freespace gavs hastigheten 0 eftersom freespace inte har någon hastighet. Detta illustreras i figur 4.2. Vi fattade även ett beslut att inte undersöka ifall freespace-hindret eventuellt skulle förflytta sig åt

någon riktning. Det som upptäcktes då var att det blev fler falska varningar och att analyseringsverktyget blev långsamt att använda som följd av detta.

Förmodligen berodde dessa varningar på för att TTC var för högt satt för freespace. Därför beslutades det att skapa en egen funktion som endast hanterar freespace-ärenden.



Figur 4.2: Visuellt hjälpmedel för analys av objekt och freespace över tid.

4.2 Konstruktion av hårdvara

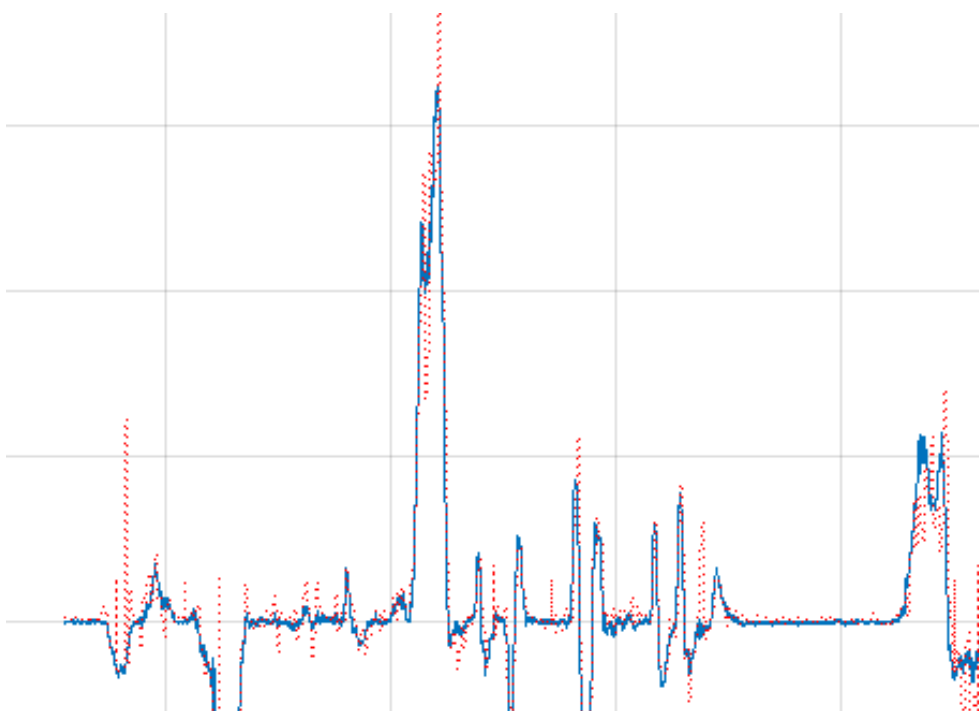
Radarsensorerna monteras på förbestämd placering. För att kunna uppskatta avstånden och göra tillräckligt långa kablar mättes detta upp på förväg hos Göteborgs Spårvägar. När kablarna sätts fast på radarn går dessa inte att montera av och därför är det viktigt att kablarna är tillräckligt långa. Därefter kopplades det tio meter långa kablar enligt ett givet kopplingschema. För radarns "master" kopplades fem olika kabelpar (databussar). För radarns "slave" var designen liknande förutom att det inte användes någon sladd för ett kabelpar som heter VCAN.

Spårvagnens girvinkelhastighet, "yaw rate", måste samplas och skickas till radarn med ett kontinuerligt intervall. Eftersom det inte finns någon sådan data att hämta från spårvagnen köptes en gyrosensor (MPU9250) in av Aptiv. I detta projekt

4. Genomförande

används gyrots z-axel för att beräkna yaw rate. Gyrot placeras längst fram vid förarens fönster.

För att tillse att girvinkelhastigheten från gyrot stämmer överens med den faktiska girvinkelhastigheten jämfördes den med värdena från en personbil. Ett CAN-case användes för att logga datan från arduinon parallellt med datan från bilen. Sedan plottades datan upp i MATLAB där de kunde jämföras. Anledningar till att de ibland skiljer sig skulle kunna tänkas vara att gyrot inte satt fast under testet utan hölls på plats av en medpassagerare, vilket kan leda till avvikelser. Datat från gyrot var lik datat från Volvon och därmed kunde gyrot användas som givare för yaw rate. Se figur 4.3.

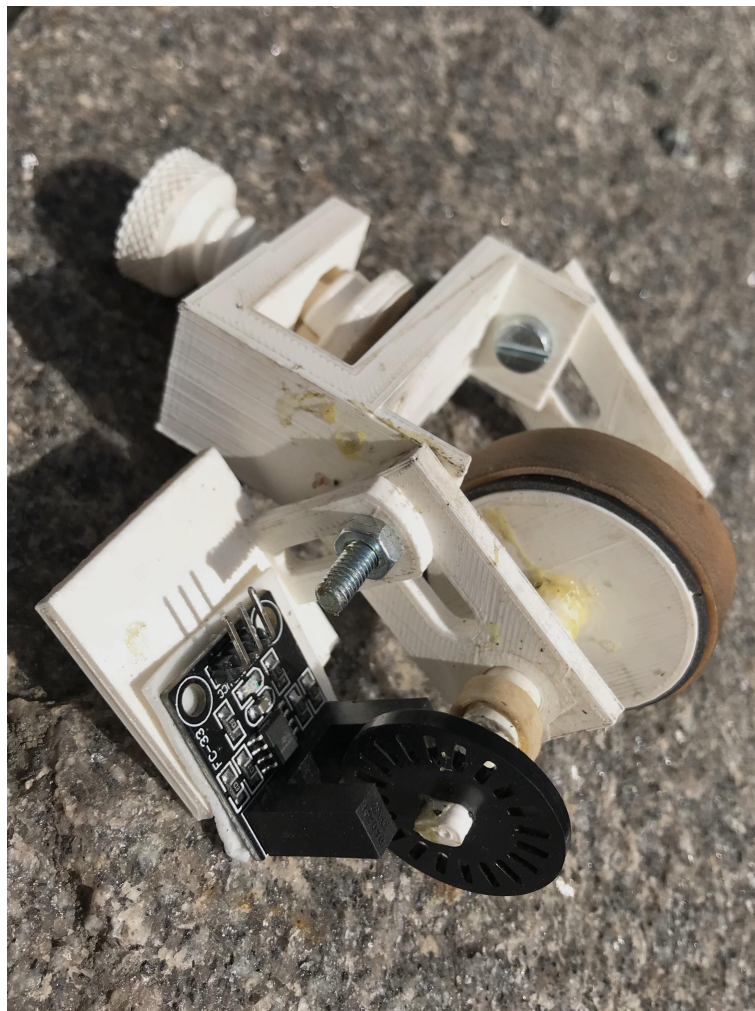


Figur 4.3: Diagram över yaw rate. Blå (heldragen) linje symboliserar "yaw rate" från Volvo V40, rött (prickad linje) symboliserar yaw rate från Arduino med gyrosensor.

Det upptäcktes under testet för "yaw rate" att sensorn inte alltid skickade ut CAN-meddelanden var 20e millisekund, utan ibland blev det istället var 30e millisekund. I jämförelse med CAN-meddelanden från Volvon som kunde skickas med ett intervall på 0.006 ms i differens, ansågs detta vara en för stor skillnad. Koden ändrades då för att kunna skicka ut data med kortare och jämnare mellanrum genom att endast läsa av sensorn var 19e millisekund istället för oavbruten avläsning. Detta för att inte belasta Arduinon på samma sätt, och genom detta kom CAN-meddelanden ut på bussen med en variation på 0.016 ms vilket kändes tillräckligt inför det första testet på spårvagnarna.

Då även spårvagnens hastighet saknades behövdes en lösning för detta. En modul

med optisk läs-gaffel samt enkoderskivor köptes in av Aptiv. För att kunna läsa av hastigheten konstruerades ett hjul samt monteringsverktyg i CAD. Dessa kunde sedan 3D-printas och fästas under vagnen för att läsa av vagnens hastighet. Att rita upp hjulet och konstruera designen tog mer tid än väntat, mycket på grund av det väldigt begränsade utrymmet samt att inga mått under spårvagnen är konstanta. Designen behövde därför vara både liten och justerbar. För att undkomma de varierande måtten under vagnen användes en klämma för att skapa en fästordning. Från denna klämma stack sedan några justerbara skenor ut med en hjulaxel. Både klämman, skenor, sensorplatta och hjulet printades ut i flera olika storlekar för att kunna ha möjlighet att fästa hjulet på olika sätt. Tanken är att hjulet ska snurra mot spårvagnens axel, och på hjulets axel sitter enkoderskivan fast och snurrar inuti läs-gaffeln, vilken räknar antalet varv per minut. Eftersom det 3D-printade hjulet hade låg friktion fästes ett gummiband på hjulet för att öka friktionstalet. Se figur 4.4 och 4.6 för att förstå hjulets konstruktion.



Figur 4.4: *Det 3D-printade hjulet med enkoderskivan, reglerbart fäste och läs-gaffeln.*

För att få fram hjulets hastighet krävdes en funktion som med hjulets omkrets och antal varv kunde räkna ut avståndet som rullats under en viss tid. Tidsinter-

vallet sattes till sekunder och avståndet mättes i meter vilket ger det 3D-printade hjulets hastighet i meter per sekund. Därefter behövdes detta jämföras med den verkliga sträckan, då hjulet snurrar mot en axel med en annan omkrets. Axeln är inte hjulaxeln till spårvagnen, utan en fristående axel som rör sig samma avstånd som spårvagnen under en viss tid. Detta gör att axelns omkrets inte är relevant i uträkningarna kring varv per minut för det lilla 3D-printade hjulet. Istället mäts hur många pulser som skulle ges för spårvagnens hjul.

Formeln för hastighet grundar sig på att man räknar hur många varv det lilla hjulet snurrar när spårvagnens hjul snurrat ett varv, sedan multipliceras detta med hastigheten, som är varv per minut för det 3D-printade hjulets omkrets. Formeln testades även genom att låta hjulet snurra mot ett cykelhjul. Både omkrets från 3D-hjulet och cykelhjulet mättes upp, sedan togs en snitthastighet från cykelhjulet fram och denna jämfördes med den hastighet som det 3D-printade hjulet uppgett. Det visades vid testet att värdet från det 3D-printade hjulet behövdes divideras med tio. För implementering på spårvagn ändras värdena från cykelhjulet till spårvagnens hjul. Därefter jämförs detta med spårvagnens egna hastighetsmätare.

Spårvagnens hastighet fås ut genom att läsgaffeln, enkoderskivan och hjul placeras dikt an mot spårvagnens axel. Då blir hastighets sambandet:

$$\begin{aligned}x &= \text{hjulets omkrets} / \text{spårvagnshjulets omkrets} \\ \text{spårvagnens kadens} &= x * \text{hjulets kadens} \\ \text{spårvagnens kadens} &= \text{hjulets kadens} * \text{spårvagnshjulets omkrets}\end{aligned}$$

All kod för hastighet och girvinkelhastighet samt CAN-kommunikation behandlas av en Arduino som placerats längst fram i förarhytten. Radarsensorerna monteras på vagnens front med hjälp av 3D-printade fästen och silvertejp. För att undvika att spårvagnens front blockerar vyn har radarsensorernas fästen modifierats så att de är justerbara. Sensorerna tar emot data från gyroskopet och hastighetsläsaren var 20:e millisekund. Om en sensor skulle gå sönder eller skadas är de enkla att byta ut eftersom kablarna endast löts fast i stift. Då kan dessa sensorer enkelt proppas in eller ut utan att kablarna ska behöva lödas om. Fler bilder från testdagen återfinns i bilaga B.

4.3 Utveckling av varningssystemet

Det krävdes mycket data för att kunna avgöra när systemet skulle varna. En del loggar samlades från lastbilen, men det saknades ytterligare information kring typiska spårvagnssituationer. I början användes det tidigare beskrivna analyseringsverktyget för att göra bedömningar kring när lamporna ska tändas. I loggarna från lastbilen ansågs objekten vara av störst vikt, men i spårvagnstrafiken är det annorlunda på grund av olika trafiksituationer och en genrellt lägre hastighet. Eftersom objekt inte uppfattas om de inte har en viss hastighet ansågs freespace vara ett

värdefullt verktyg vid hållplatserna där människor ofta går från att vara helt stilla till att röra sig väldigt plötsligt, något som objekttrackern inte hinner registrera.

När TTC för objekt skulle beräknas valdes en ganska enkel funktion. Till en början undersöker vi om objektet befinner sig inom en viss bredd framför vagnen som anses vara värd att granska. Bredden som valdes var två meter ut från spårvagnens mitt åt vardera håll. Spårvagnens bredd uppskattas vara ungefär 1.30 meter åt varje håll, därmed undersöks lite extra utrymme men inte alltför mycket för att det ska uppstå många falska varningar. Detta måste självklart studeras vidare för att kontrollera att alla farliga objekt fortfarande analyseras och att ingen säkerhet går förlorad. Sedan kontrolleras spårvagnens hastighet, om spårvagnen är stillastående anses det för tillfället vara onödigt att varna. Från början var tanken att spårvagnen inte skulle kunna köra iväg ifall någon befann sig framför den, men på grund av tidsbrist kommer inte situationen då någon befinner sig framför en stillastående vagn analyseras vidare.

Objekt

"Time to collision" (TTC) avgörs sedan genom att undersöka objektets longitudinella position. Tiden är hur lång tid det tar för spårvagnen att hinna ifatt objektet och baseras på spårvagnens hastighet samt objektets position och hastighet. Det är även känt sedan innan att spårvagnen har en retardation på 1.4m/s^2 vid normal driftbroms. Förarens reaktionstid uppskattas till en sekund, baserat på tidigare angivelser i teoriavsnittet stoppsträcka. Tiden till krock är alltså när avståndet mellan vagn och objekt är lika stort eller mindre än stoppsträckan.

Reaktionssträckan räknas endast med för den gula lampan. Då den röda lampan ska indikera automatisk nödbroms antas ingen reaktionssträcka, utan endast bromssträcka, och detta är bromssträckan för magnetskenbromsarna. Magnetskenbromsarna minskar, som beskrivet tidigare, den totala bromssträckan med 25-30 procent. För denna funktionsutveckling antogs retardationen öka med 30 procent. Röd lampa kräver alltså inga reaktionssträckor eftersom spårvagnen ska sköta ärendet utan förarens infall. Den blåa lampan kräver inte heller några reaktionssträckor. Detta eftersom blå lampa ska indikera varningar utåt, exempelvis att spårvagnens ringklocka går igång, helt enkelt att människor runt omkring ska uppmärksamma spårvagnen, och det krävs ingen interaktion från spårvagnsföraren.

Funktionerna utgår från är den allmänt kända formeln $t = s/v$. Låt "ov" beteckna objektets hastighet, "op" beteckna objektets longitudinella position, och "sv" beteckna spårvagnens hastighet. Sträckan är stoppsträckan vars beräkningar beskrivs i avsnittet stoppsträcka. Det finns även en startsträcka för objektets position. Formlerna som används för detta visas här nedan.

Röd lampa: $(op + ov * TTCr - ((sv/3.6)^2/2 * 1.4 * 1.3))/(sv/3.6)$

Gul lampa: $(op + ov * TTCy - (sv/3.6 + ((sv/3.6)^2/2 * 1.4))/(sv/3.6)$

Blå lampa: $(op + ov * TTCb - ((sv/3.6)^2/2 * 1.4)/(sv/3.6)$

Objektets position är dess position plus dess hastighet multiplicerat med dess gräns

för dess TTC. För röd lampa multipliceras bromsarnas effektivitet med 1.3 då magnetbromsen är 30 procent mer effektiv. I beräkningen för gul lampa är även reaktionstiden inkluderad.

Eventuellt överhäng eller analys kring hur spårvagnen svänger är ännu inte inkluderat i beräkningarna. I slutändan sattes TTC för röd lampa till en sekund, gul lampa till två sekunder, och blå lampa till tre sekunder. TTC-värdena uppfattas dock individuellt och bör verifieras av spårvagnsförare. Anledningen till att värdena ett, två och tre valdes var på rekommendation från personer som arbetat med TTC på Aptiv sedan innan. Vi hade ingen tid att göra egna undersökningar och utvärderingar på vad som skulle uppfattas som lämpliga tider och valde därför dessa rekommenderade tider.

Freespace

Freespace utgår som tidigare nämnt från 60 sektorer. Initialt är sektorernas längd 300 meter och om ett hinder blockerar en sektor fås avståndet till hindret. För varje tidssampel kollar funktionen avståndet till samtliga sektorer i x-led och y-led. Då spårvagnen kör nära människor och fordon är x-koordinaten extra viktig. Därför bestäms ett riskområde, i x-led och y-led, för vadera lampa. Riskområdets koordinater beror på vilken sektor och visas här nedan.

För sektor 1-30:

$$x = \text{sektorns längd} * \cos((\text{sektor} * 3) * (\pi/180))$$
$$y = \text{sektorns längd} * \sin((\text{sektor} * 3) * (\pi/180))$$

För sektor 31-60:

$$x = \text{sektorns längd} * (-\sin((\text{sektor} * 3 - 90) * (\pi/180)))$$
$$y = \text{sektorns längd} * \cos((\text{sektor} * 3 - 90) * (\pi/180))$$

Om ett hinder befinner sig i riskområdet och spårvagnen kör, beräknas hindrets tid till kollision (TTC). TTC för freespace sätts till samma värden som för objekten för respektive lampa och dessa är röd= 0.5 sekunder, gul= 1.5 sekunder och blå= 4 sekunder. Nedan återfinns formlerna för respektive lampa, där "hpy" är hindrets longitudinella position, "sv" är spårvagnens hastighet, "1.4" är spårvagnens retardation och "1.3" baseras på att magnetskenbromsarna är 30 procent effektivare än driftbromsen. Konstanten "reak" står för reaktionstiden.

Röd lampa: $(hpy - ((sv/3.6)^2/2 * 1.4 * 1.3)/(sv/3.6)$

Gul lampa: $(hpy - (reak * (sv/3.6) + ((sv/3.6)^2/2 * 1.4))/(sv/3.6)$

Blå lampa: $(hpy - ((sv/3.6)^2/2 * 1.4)/(sv/3.6)$

När freespace-funktionen implementerades tillsammans med objektfunktionen i det visuella hjälpmedlet blev hjälpmedlet väldigt långsamt, eftersom alla 60 sektorer och objekt visades samtidigt. Detta berodde på att lamporna skapades på nytt för varje tidssampel och dessutom för varenda sektor. Därefter ändrades detta till en lampa som skapas i början av funktionen, vars färg endast uppdateras när tid till kollision-kriterierna är uppfyllda. Om lampan redan har den färg som skall visas,

uppdateras inte lampans färg. Dessutom prioriteras lampornas färg, om lampan är röd kan den inte ändras till gul eller blå, även om kriterierna för blått och gult är uppfyllda. Är lampan gul kan den inte ändras till att bli blå, men om kriterierna för rött uppfylls ändras den från gul till röd.

Utöver problemen med freespace i det visuella hjälpmedlet finns en riskfaktor, nämligen att funktionen inte beräknar var hindrerna kommer att befinna sig vid nästa tidpunkt. Därför gjordes en bedömning ifall varningssystemet skulle varna en gång för mycket, vilket var en negativ synpunkt från testet som genomfördes 2013 sedan, eller om varningssystemet ska vara lite mer slapt vilket kan dra ner på säkerheten. Efter att ha analyserat och jämfört spårvagnens trafiksituationer valdes att hellre undvika för många falska varningar. Detta då de flesta spårvagnsolyckor, ca 200 mer än övriga typer, sker med andra fordon, och fordon har större möjligheter att ses som objekt då de ofta rör sig med en större hastighet.

4.4 Test på spårvagn

Det första testet skedde i början av maj, och ämnade att samla in loggar för att undersöka hur spårvagnstrafiken skiljer sig från biltrafiken. Testet genomfördes i depåområdet i Gårda hos Göteborgs Spårvägar. De situationer som skulle testas var följande:

- Möte med en annan spårvagn
- En spårvagnshållplats med väntande passagerare
- Elstolpar eller staket som spårvagnen vanligtvis kör väldigt nära
- Gångtrafikanter som går bredvid spårvagnen
- Gångtrafikanter som hastigt ger sig ut framför spårvagnen
- Ett fordon som kör framför spårvagnen
- Påkörning av en person alternativt en pappdocka

Inför det första testet hade det på förväg bestämts att radarn skulle sitta längst fram på vagnen, 1.6 meter ifrån varandra, 0.9 meter upp och med en vinkel på nio grader. Detta var tvunget att förbestämmas då mjukvaran konfigurerades av anställda på Aptiv. Ingen av dessa utvecklare kunde vara med under testdagen, därför krävdes det att radarsensorerna monterades på väldigt exakt eftersom vi inte hade möjlighet att ändra mjukvaran under testdagen. Det visade sig vara svårt att fästa dem med dessa exakta mått. Om positionen behövdes ändras förlorades vinkeln och vice versa. Det slutade med att positionerna och vinkeln fick anses godtyckliga. I slutändan uppmättes en differens på någon centimeter i höjd, vilket också kan ses i figur 4.5. Detta kan förstås leda till att datan inte uppfattas helt korrekt.



Figur 4.5: Radarsensorer monterade på spårvagn av typen M32.

Under testdagen gjordes upptäckten att plattan med nätverkslådor, strömförsörjning och CAN-utrustning var för stor för att få plats i utrymmet på golvet i förarhytten. Detta berodde på att vi inte hade tagit hänsyn till att få plats med testutrustningen och en person som sköter loggningsverktyget Orcas vid uppmätningen av spårvagnens förarhytt. Vi hade endast mätt upp plats för testutrustningen. Dessutom hade kablaget knutit ihop sig på vissa stället vilket gjorde att de var för korta för att hela Arduino-lösningen skulle kunna få en bra position. Plattan fick därmed placeras lutandes innanför fönstret vilket var en ganska osäker position som också orsakade skada på vissa kablar då plattan kunde trycka till kablarna när spårvagnen svängde. Längre kablar är därmed att rekommendera. Kablarna var (initialt) sex meter långa, och plattan ungefär 50x30 centimeter. Det uppstod även ett problem under testdagen med att loggningsverktyget Orcas hängde sig och tappade kontakt med alla sensorer. Enda lösningen var att stänga av och sätta på strömmen till alla sensorer, vilket också bör lösas inför kommande tester.

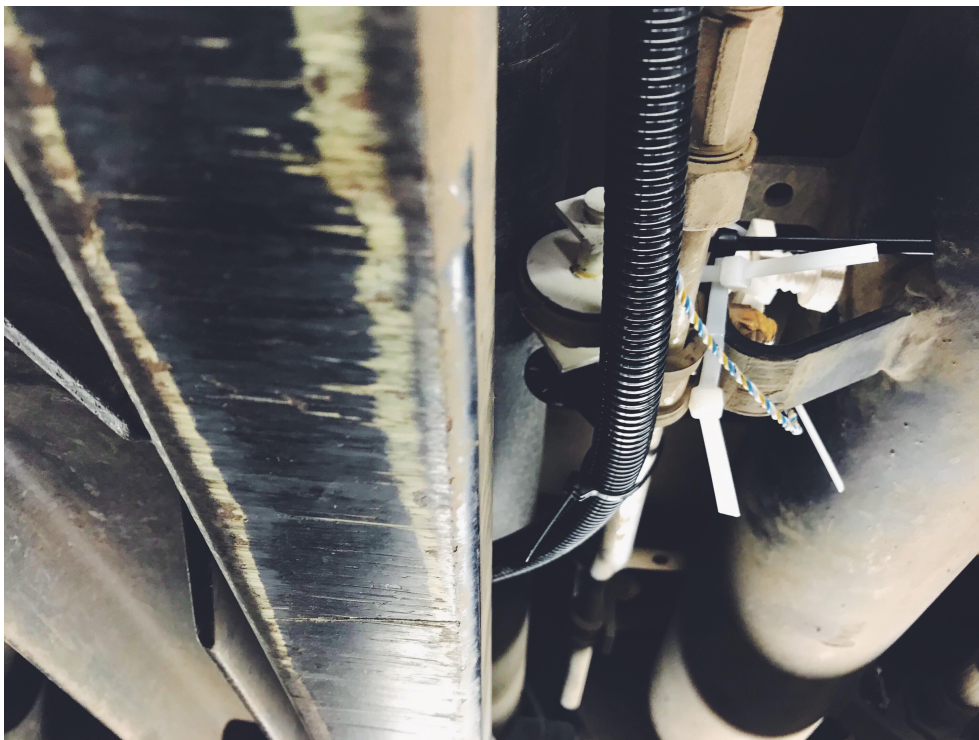
En annan svårighet var det 3D-printade hjulet som skulle mäta hastigheten. Vid första tillfället då spårvagnen körde framåt med hjulet monterat, höll hjulet på att knäckas. Detta berodde på att axeln som hjulet låg emot rörde sig mer än förväntat och att hjulet var riktat rakt mot axeln. Därmed kunde axeln trycka till och dra med hjulet till dess att hjulet höll på att förstöras. Lyckligtvis kunde detta korrigeras, men det drog ut på tiden. Figur 4.6 illustrerar hur hjulet monterades från början.



Figur 4.6: I figuren visas den initiala monteringen av hjulet på spårvagnen. Hjulet ligger rakt mot axeln.

Detta åtgärdades genom att hjulet monterades på nytt med en nedåtriktad vinkel istället för den initialt raka placeringen av hjulet. Se figur 4.7. Hjulet kunde då inte följa med axelns rörelser hela vägen upp till den mest kritiska punkten (där det nästan gick av) vilket gav ett betydligt bättre och mer hållbart resultat. Däremot kunde hastigheten variera mycket då hjulet stundtals slirade mot axeln istället för att snurra.

Det lilla hjulet hade en omkrets på ungefär 13.5 centimeter och spårvagnens egna hjul hade en ungefärlig omkrets på 205 centimeter. Hjulet snurrar därmed cirka 15 varv när spårvagnens hjul snurrat ett varv. Då alla dessa mått var ungefärliga blev den faktiska hastigheten inte helt precis med spårvagnens egna. Eftersom axeln snurrade i samma takt som spårvagnshjulen var dess påverkan försumbar i beräkningarna. Detta kunde självklart ha kalibrerats ytterligare under testdagen, men på grund av bland annat tidsbrist och ett väldigt begränsat utrymme vid Arduinon ansågs det vara svårt att komma åt att ladda upp ny kod till den. Hastigheten var tillräckligt lik för att kunna samla in loggarna. Det fanns dock ingen beräkning för huruvida spårvagnen åkte framåt eller bakåt. Girvinkelhastigheten, "yaw rate", ansågs att vara tillförlitlig fastän att då spårvagnen var stillastående varierade girvinkelhastigheten konstant mellan 0 och -0.01 radianer per sekund.



Figur 4.7: I bilden är hjulet vinklat snett nedåt till skillnad från den raka initiala monteringen. Hjulet är vinklat för att undvika knäckning.

Efter montering kunde alla loggar samlas och sparas verktyget Orcas. Alla testfallen kunde simuleras med undantag för en påkörning då ingen pappfigur fanns tillgänglig. Hållplatsen skapades genom att några personer ställde sig en bit framför spårvagnen väldigt nära spåret, så som det ser ut vid typiska spårvagnshållplatser. Göteborgs Spårvägar berättade även att en vanlig olycka är när andra fordon gör u-svängar framför spårvagnen. Även detta återskapades med en bil, dock medan spårvagnen stod stilla för att förhindra en olycka. Även staket, svängar med blockerad vy och mötande spårvagnar kunde ordnas.

4.4.1 Genomgång av spårvagnsloggar

Lyckligtvis hade de flesta testfallen filmats med mobilkamera, men på de fall där video saknades kunde det vara svårt att veta vad som faktiskt pågick och huruvida alla objekt- och freespace-detektioner stämde överens med verkligheten. I vissa fall märktes att saker som inte alls rörde sig kunde uppfattas som objekt, exempelvis kunde en lyktstolpe bli ett objekt. Detta var svårt att avgöra när video saknades.

Det blev även många fel när spårvagnen backade eftersom läsgaffeln inte kunde hantera detta, då den inte kan bestämma rotationsriktning. Även när spårvagnen backade snurrade hjulet mot axeln och gav en hastighet. Samtidigt uppfattade radarsensorerna det som att föremål rörde sig närmre spårvagnen vilket gjorde att de uppfattades som rörliga objekt. Dessa loggar var inte särskilt användbara då det blev upp emot tio objekt som rörde sig åt alla möjliga håll. Möjligheten att logga även när spårvagnen backar bör självklart implementeras inför kommande tester.

Loggarna lades in i tidigare beskrivet analyseringsverktyg. I näst intill alla testfall uppfattades hinder som freespace och inte objekt. Detta antas bero på spårvagnens och hindrens låga hastigheter, men skulle även kunna tänkas bero på att radar-sensorerna inte varit helt korrekt monterade. Alternativt att det saknades fler testfall som skapade objekt, exempelvis passagerare som går längs med vagnen en längre sträcka samtidigt som den kör.

I de fall då radarn upptäckte objekt kunde det även dyka upp nya objekt men längre bort. När testfallet med bilen som gör en u-sväng framför spårvagnen skulle loggas, detekterades flera falska objekt en bit bakom bilen. När filmerna från testet undersöktes visades det att det inte fanns fler objekt vid det tillfället. Detta hände även vid andra tillfällen som objekt detekterades, alltså att "spökobjekt" upptäcktes en längre bit bort ifrån det faktiska objektet. Detta ansågs bero på att radarn ej varit helt korrekt monterade, som beskrivet sedan innan. Det uppfattades dock inte som ett större problem då dessa spökobjekt alltid befann sig längre bort och alltså inte påverkade testerna av funktionerna.

Något annat som upptäcktes när loggarna analyserades var att hastigheten hade fått ett annat värde i filerna. Exempelvis när hastigheten var noll skrevs det ändå ut 28 i MATLAB. Detta misstänks bero på att loggarna tolkades fel i MATLAB-scripten.

4.5 Arbetsmetod

I början av projektet skapades en planering. Planeringen hade ett mål inför varje vecka då arbetssättet har varit SCRUM-inspirerat. Målet var att se till att varje vecka medför något av värde. För att uppnå veckomålet skapades mindre deluppgifter. Dessa deluppgifter var som "elefantbitar", vilket innebär att de alla var värdefulla - det fanns ingen mening med att vänta på att en uppgift skulle bli klar innan en annan kunde påbörjas. Dessa uppgifter sparades sedan på en online anslagstavla där de gick från "Att göra" till "Pågående" till "Slutförd". Vissa uppgifter kunde även flyttas till veckan efter om de inte blev avklarade. Att bryta ner de större arbetsuppgifterna till mindre bitar har också varit väldigt hjälpsamt och gjort att arbetet flutit på utan några stopp där det plötsligt inte finns något att göra. Det gick till en början bra att följa planeringen, men målen ändrades efter ungefär halva tiden då hårdvaran ansågs vara en viktig och tidskrävande bit som inte tagits i beräkning när första planeringen skrevs.

För att samla in information till projektet samt för att skapa förståelse för utvecklingsmiljöerna har vi bland annat sökt efter artiklar och videor via nätet. Statistiken har samlats in genom att hämta information från diverse artiklar där tidigare trafiksäkerhetschefen och presstalespersoner från Göteborgs Spårvägar uttalat sig om olyckor och olyckstyper. Sedan har statistik även fått via mail från en kontakt på Göteborgs Spårvägar. Denna kontakt har även bidragit med information kring hur tidigare tester har fungerat.

För att få förståelse för hårdvaran har kunniga personer från Aptiv varit

4. Genomförande

främsta informationskällan. Sedan har även en del datablad lästs igenom. För att kunna ändra och programmera sensorerna har vi utgått från öppen källkod som ligger på Github.com, varav försäljningssidorna hänvisar köparna till vissa länkar på Github. För att säkerställa att den information som hittats och tagits fram är korrekt har resultatet testats mot verkligheten, exempelvis genom att kontrollera att de hastighetsvärden som sensorerna skickar ut stämmer med den faktiska hastigheten.

5

Resultat

Det som har tagits fram är en prototyp på ett varningssystem samt ett portabelt loggningssystem. Systemet är portabelt då det inte funnits tillstånd för att montera eller hämta ut data från spårvagnarna. Detta medför dock att systemet inte får användas i trafik på grund av att spårvagnarnas trafikregler inte tillåter att något sticker ut från fronten på spårvagnen.

Systemet består av två radarsensorer, en hastighetsmätare med tillhörande hjul, mätare för girvinkelhastighet ("yaw rate") och en Arduino som hanterar all kod. Varningssystemet i sig är ej testat på en spårvagn, men på de loggar som samlats från spårvagnen.

Varningen sker genom att olika "lampor" tänds när ett objekt befinner sig i riskzonen eller när vagnens freespace blockeras. Den röda lampan motsvarar en automatisk nödbroms, och i denna algoritm finns därmed ingen reaktionstid inräknad. Gul lampa ska tändas för att uppmärksamma föraren och ge denne tid för att bromsa innan hindret. Lampan tänds därför när tid för kollision uppskattas till 2 sekunder. Reaktionssträckan är inräknad här. Den blå lampan tänds för att varna kringtrafik, exempelvis genom att ringa i spårvagnens klocka för att uppmärksamma personer runt vagnen. Blå lampa tänds då tid till kollision är 3 sekunder.

Varningstiderna är baserade på allmänna formler som $s = v * t$. De har utvecklats med loggar från både spårvagn och lastbil som underlag. I analyseringsverktyget som utvecklats för att kunna kontrollera varningarna sker både gula och blå varningar, dock inga röda varningar vilket är logiskt eftersom ingen krock inträffade.

Varningssystemet behöver dock testas och verifieras för att vara helt säker på att alla varningssituationer registreras av funktionerna och att tiderna uppskattas korrekt.

Under utvecklingsprocessen har projektets målbild ibland behövt ändras. Detta beror till stor del på att vid projektets start var det en självklarhet att få ut hastigheten från spårvagnen. När det visade sig att detta inte var möjligt gick mycket tid åt för att skapa en hastighetsmätare. Detta medförde att allt som skulle utvecklas inte hanns med. Bland annat blev det ingen närmre undersökning kring hur förarna skulle interagera med varningssystemet, eller hur det skulle utnyttjas på bästa sätt. Det blev heller ingen undersökning kring hur väl systemet fungerade, och varningssystemet testades aldrig i spårvagnstrafiken. Ett mål var också att se om olycksstatistiken kunde minska, detta har heller inte kunnat verifieras ännu. Det som däremot utvecklats är ett portabelt kollisionsvarningssystem som kan hantera

både stillastående och rörliga hinder.

Ett mål var att minska olycksstatistiken. Vid en genomgång av olycksstatistiken som inhämtats visar den att majoriteten av olyckorna beror på den mänskliga faktorn. Det är även så att de allra flesta olyckorna som sker är spårvagn mot ett annat fordon, dessa är runt 200 mer än den näst vanligaste förekommande olyckstypen som är fall i vagn. Larmsystemet skulle alltså kunna förhindra flera olyckor om de beror på ouppmärksamma förare, antingen inuti spårvagnen eller i kringtrafiken. Varningssystemet verkar dock för att skapa hastiga inbromsningar, detta skulle kunna leda till att olyckor med fall i vagn ökar, eller att denna typ av olycka helt enkelt inte kan förhindras med detta varningssystem. Systemet ska dock kunna motverka flera av de andra olyckstyperna, som sammanlagt enligt statistiken motsvarar runt 300 varje år. Hur många av dessa som kan förhindras är inte fastställt.

För att sammanfatta projektet i jämförelse med de syften och mål som beskrevs i det inledande kapitlet finns det potential för systemet att minska olycksstatistiken. Det har inhämtats olycksstatistik, information kring spårvagnar, och spårvagnsloggar som ligger till grund för hur systemet varnar i olika situationer. Ett varningssystem med tre olika nivåer har utvecklats. Systemet behöver modifieras för att fungera i spårvagnstrafiken då det inte följer spårvagnarnas monteringsregler. Det har ännu inte genomförts någon undersökning kring hur väl interaktionen mellan människa och varningssystemet fungerar.

6

Diskussion

Då de absolut vanligaste olyckorna med spårvagn är just krock med andra fordon skulle många av de 300 skadliga olyckor som årligen sker kunna förhindras med hjälp av detta system. Systemet bör även ha möjlighet att spara in på stora kostnader, då olyckor kostar mycket pengar att åtgärda oavsett om någon blir skadad eller ej. Utöver skador som Göteborgs Spårvägar måste betala skulle även samhällspengar kunna sparas in genom att inga vägar behöver stängas av, ingen räddningspersonal behöver åka till platsen och övriga omkostnader. Något som systemet dock inte är anpassat för är de fall som sker inne i spårvagnen till följd av hastiga inbromsningar. Fallolyckorna är som tidigare nämnt en vanlig typ av olycka där personer tar skada och vi har inte undersökt till vilken bredd denna olyckstyp skulle kunna förhindras.

Vidare är det viktigt att tänka på att de allra flesta olyckor händer kring Centralstationen, som är en väldigt befolkad plats. Inga loggar har samlats vid en sådan plats, och risken finns att systemet skulle kunna ge många falska varningar eller missa att varna vid vissa situationer. Det saknas mycket data för att kunna avgöra hur bra systemet fungerar i dagsläget.

Systemet är även konstruerat för att inte ge falska varningar, detta ansågs viktigt då tidigare trafiksäkerhetschef hos Göteborgs Spårvägar ansett att det tidigare systemet varnat alldeles för ofta mot sådant som inte var ett hinder. En risk med detta är som sagt att vissa hinder kanske inte tas i beräkning så som de bör. Detta kan vara särskilt aktuellt vid freespace då det var svårt att sortera vad som var fasta hinder och vad som var en människa som plötsligt kunde gå ut mitt i vägen. Varningssystemet måste ha en ganska liten riskzon för att undvika falska varningar, särskilt som spårvagnen kör väldigt nära hållplatser, elstolpar och andra spårvagnar. Vid en analys av olycksstatistiken ansågs det vara okej att ha små marginaler för när freespace ska varna. Detta eftersom de absolut flesta olyckorna sker mot andra fordon eller inuti vagnen. Olyckor med andra fordon beror ofta på, som Göteborgs Spårvägar berättade, att andra fordon försöker svänga in framför spårvagnen eller göra u-svängar framför denna. Om detta sker i trafik kan det tänkas att dessa fordon bör uppfattas som objekt, och det är som sagt enklare att beräkna en mer exakt kollisionstid för objekt eftersom de har hastighet. Samtidigt är det något som spårvagnsföraren bör vara medveten om så att denne inte förlitar sig helt på varningssystemet.

Även hårdvaran behöver modifieras för att kunna fungera i spårvagnstrafiken. Eftersom ingen får sticka ut, vilket radarsensorerna gör, finns en tanke att sensorerna skulle kunna fästas innanför spårvagnens front. Där finns gott om utrymme. Detta kunde inte genomföras under testerna eftersom det hade krävt att det skulle göras hål i spårvagnen. Sedan är det inte särskilt hållbart att ha ett hjul och en lång kabel under spårvagnen som ger en väldigt varierande hastighet. Istället bör det sökas tillstånd långt i förväg om systemet ska implementeras på en spårvagn. Detta togs inte i beaktning inför detta projekt, men som rekommenderas om projektet ska göras om. Med dessa optimeringar blir också datan från radarn mer pålitlig eftersom allt kan monteras på med korrekt höjd och vinkel, och hastigheten blir mer stabil. Då hade de spökdetektioner som uppstod under de loggar vi samlat in förmodligen inte registrerats av radarn.

Varningssystemet skulle kunna ses som ett steg mot självkörande spårvagnar. Det upptäcker hinder och ger hänvisningar kring när det är fritt fram att köra spårvagnen samt när den bör bromsa och nödbromsa. Ett nästa steg skulle kunna vara att implementera en adaptiv farthållare. En svårighet kan dock vara att mycket i spårvagnen är manuellt, exempelvis gamla bromssystem. Självkörande spårvagnar lär därför bli på framtida spårvagnar och inte på de spårvagnar som finns i dagsläget.

6.1 Hållbarhet och etik

Med hjälp av det utvecklade varningssystemet finns det möjligheter att bespara ekonomiskt kapital genom att förhindra olyckor. Detta genom att förhindra krockar och urspårningar, vilket som tidigare nämnt kostar mycket pengar. Pengarna skulle istället kunna läggas på att utveckla spårvagnstrafiken som är ett mer miljövänligt alternativ än annan kollektivtrafik.

När det kommer till det etiska perspektivet skulle systemet också kunna bespara förarna eventuell ångest och stress om de lyckas undvika kollisioner. Även liv kan räddas och flertalet skador och sjukhusvistelser skulle kunna förhindras. Detta är positivt och hjälper mot målet om noll dödsfall i Sveriges trafik. Varningssystemet behöver också självklart testas innan det används i trafiken, och det bör göras en undersökning kring hur spårvagnsföraren interagerar med systemet. Föraren bör inte bli helt beroende av systemet eftersom teknik alltid kan fel. Detta är särskilt viktigt eftersom varningssystemet har utvecklats med en tanke att det inte ska ge falska varningar, det litar alltså på att föraren fortfarande kan göra egna bedömningar, exempelvis vid start av spårvagnen. Vid vidareutveckling av systemet skulle det dock kunna leda till att förarna inte anses behövas längre om spårvagnen blir självkörande, men det är inte aktuellt för det systemet som har utvecklats i detta projekt.

6.2 Förslag på vidareutveckling

För att kunna förhindra olyckor med fall i vagn hade det varit optimalt om systemet kunnat kombineras med en adaptiv farthållare för att skapa mjuka inbromsningar i rätt tid, vilket föreslås för vidareutveckling av projektet.

Det bör även genomföras en studie kring människans interaktion med systemet (HMI) för att varningssystemet ska vara fullt funktionellt. Systemet kan sedan optimeras utifrån denna information. För att få full förståelse för när systemet ska varna och när föraren ska reagera bör man även samla in fler loggar av spårvagnstrafiken. Dessutom kan det vara viktigt att analysera situationer som föraren är med om där det inte sker olyckor.

6.3 Slutsats

Det utvecklade varningssystemet har potential att förhindra olyckor som sker till följd av den mänskliga faktorn. Systemet kommer inte kunna hjälpa de olyckor som sker inuti vagnen, exempelvis vid hastiga inbromsningar, men förhindra olyckor med fotgängare och kollisioner med andra fordon. Dessa uppskattas vara ungefär 300 varje år. Systemet behöver testas i trafik för att kunna säkerställa hur många olyckor som kan förhindras. Med hjälp av varningssystemet skulle även pengar av större summor kunna sparas in.

För att få systemet användbart i trafiken rekommenderas ett tillstånd för att koppla upp utrustningen på spårvagnens dator och därmed få ut hastigheten, samt att radarsensorerna kan monteras innanför spårvagnens front så att de inte sticker ut. Det rekommenderas även att fler loggar från andra miljöer samlas in för att systemet ska bli pålitligt och ha mycket data att utgå från.

Referenser

- [1] A. Sjögren, "Spårvägen börjar byggas i Februari", i *Sydsvenskan* (2017) [online] Tillgänglig:
<https://www.sydsvenskan.se/2017-01-18/sparvagen-borjar-byggas-i-februari>
- [2] "Spårvagnar är miljövänliga", i *Sparvagnsstaderna.se*. [online] Tillgänglig:
<http://www.sparvagnsstaderna.se/miljovanligt>, Hämtad 2018-05-13.
- [3] K. Cvetkovski och P. Alm, privat kommunikation, mars 2018.
- [4] K. Cvetkovski, (2018). Info om spårvagnar till examensarbete. [email].
- [5] "Magnetskenbroms", i *Sv.wikipedia.org* [online] Tillgänglig:
<https://sv.wikipedia.org/wiki/Magnetskenbroms>, Hämtad 2018-05-17.
- [6] "Säkra bromssystem i Göteborgs spårvagnar", i *Svenskverkstad.se* (2016). [online] Tillgänglig:
<http://www.svenskverkstad.se/sakrare-bromssystem-i-goteborgs-sparvagnar>, Hämtad 2018-05-22.
- [7] X. Ma, I. Andréasson, "*Estimation of Driver Reaction Time from Car-Following Data*", i DiVA - Digitala Vetenskapliga Arkivet. [Online]. Tillgänglig:
<https://www.diva-portal.org/smash/get/diva2:11472/FULLTEXT01.pdf>, Hämtad 2018-05-18.
- [8] A. Tussy, R. Gustafsson, D. Koenig, *Intermediate Algebra*, uppl. 4, Belmont, USA: Brooks/Cole, 2010.
- [9] "Short Range Radar (SRR)", i *Telecomabc.com* [online] Tillgänglig:
<http://www.telecomabc.com/s/srr.html>, Hämtad 2018-05-13
- [10] Clasén, H. (2018). Kollisionsvarning Spårvagn. [presentation]
- [11] "Advantages and Disadvantages of RADAR systems", i *LIDAR and RADAR Information* (2017) [online] Tillgänglig:
<http://lidarradar.com/info/advantages-and-disadvantages-of-radar-systems>, Hämtad 2018-05-13.

- [12] D. Santo, "Autonomous Cars' Pick: Camera, Radar, Lidar?", i *EE Times* (2016) [online] Tillgänglig:
https://www.eetimes.com/author.asp?section_id=36&doc_id=1330069, Hämtad 2018-05-13.
- [13] MPU-9250 Product Specification, San Jose, USA: InvenSense Inc., 2014. [Online]. Tillgänglig:
<https://store.invensense.com/datasheets/invensense/MPU9250REV1.0.pdf>, Hämtad 2018-05-22.
- [14] "MPU-9250 Hookup Guide", i *sparkfun.com* [online] Tillgänglig:
<https://learn.sparkfun.com/tutorials/mpu-9250-hookup-guide>, Hämtad 2018-05-13.
- [15] "Modul med optisk lasgaffel", i *Electrokit.com* [online] Tillgänglig:
https://www.electrokit.com/produkt/modul-med-optisk-lasgaffel/?gclid=EAIaIQobChMIzNbZ3YKD2wIV2prVCh2OVQ5vEAQYASABEgLq6PD_BwE, Hämtad 2018-05-13.
- [16] "FC-03 and attachInterrupt() problems", i *forum.arduino.cc* (2015) [online] Tillgänglig:
<https://forum.arduino.cc/index.php?topic=342650.0>, Hämtad 2018-05-13.
- [17] S. Corrigan, Introduction to the Controller Area Network (CAN), Dallas, USA: Texas Instruments Incorporated, 2016. [Online]. Tillgänglig:
<http://www.ti.com/lit/an/sloa101b/sloa101b.pdf>, Hämtad: 2018-06-07.
- [18] "Arduino Nano", i *Store.arduino.cc*. [online] Tillgänglig:
<https://store.arduino.cc/arduino-nano>, Hämtad 2018-05-22.
- [19] C. McFadden, "Raspberry Pi and Arduino: What's the Difference and Which Is Best for Your Project?", i *Interestingengineering.com* (2018). [online] Tillgänglig:
<https://interestingengineering.com/raspberry-pi-and-arduino-whats-the-difference-and-which-is-best-for-your-project>, Hämtad 2018-05-22.
- [20] "What is MATLAB?", i *Se.mathworks.com* [online] Tillgänglig:
<https://se.mathworks.com/discovery/what-is-matlab.html>, Hämtad 2018-05-22.
- [21] "Olyckorna minskar på Göteborgs Spårvägar", i *Göteborgs Spårvägar* (2014). [online] Tillgänglig:
<http://goteborgssparvagar.se/olyckorna-minskar-pa-goteborgs-sparvagar/>, Hämtad 2018-05-13.
- [22] P. Alm, (2018). Olycksstatistik. [email].

- [23] J. Marjomaa, "Förarens version av spårvagnsolyckan i Frölunda", i *SVT Nyheter (2018)* [online] Tillgänglig:
<https://www.svt.se/nyheter/lokalt/vast/forarens-version-av-sparvagns-olyckan-i-frolunda>, Hämtad 2018-05-13.
- [24] U. Nyström, "Omöjligt använda varningssystem på vagnarna", i *Göteborgs-Posten (2015)* [online] Tillgänglig:
<http://www.gp.se/nyheter/goteborg/omojligt-anvanda-varningssystem-pa-vagnarna-1.68733>, Hämtad 2018-05-13.
- [25] Representant från Consat, privat kommunikation, mars 2018.

A

Olycksstatistik

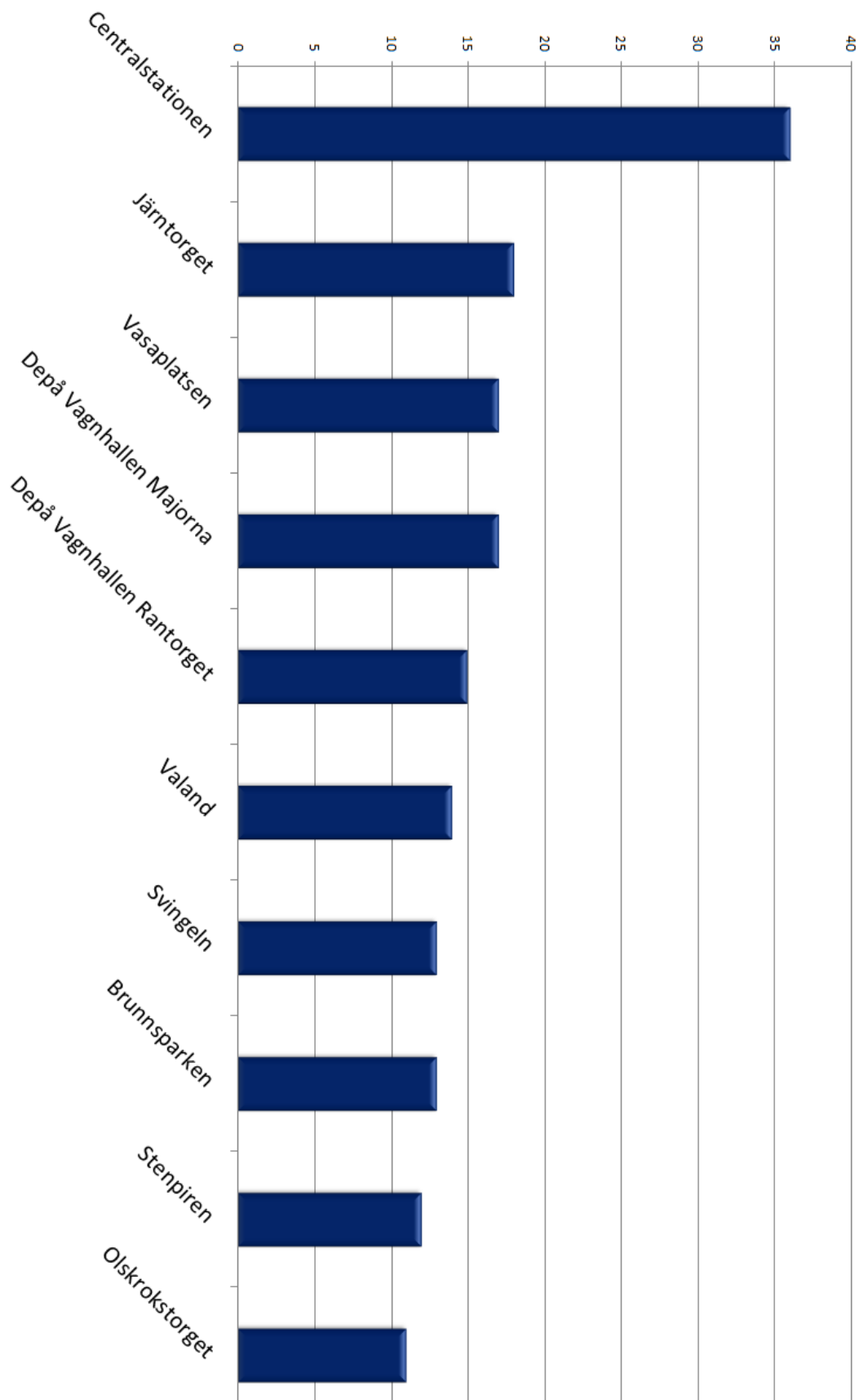
I detta avsnitt presenteras tre tabeller för olycksstatistik. Först presenteras antal olyckor och tillbud under perioden 2012-2017, därefter statistik över frekvenser för olika typer av olyckor och slutligen en tabell med antal olyckor som sker vid senast lämnad hållplats.

Spårvagnstrafik	2017	2016	2015	2014	2013	2012
Tillbud (1-3)	3541	2868	2849	2289	2235	2235
Olycka (4-7)	534	495	622	562	502	504
Lindrigt skadade	86	94	126	126	126	107
Allvarligt skadade	11	9	18	18	8	6
Omkomna	1	1	0	1	0	3

Olycksstatistik antal olyckor och tillbud 2012-2017

Händelsetyp	2017	2016	2015	2014	2013	2012
Vägtrafikolycka (spårvagn mot annat fordon)	243	254	243	223	237	257
Fall i vagn	67	56	125	110	69	53
Kollision (spårvagn mot spårvagn)	14	17	13	15	14	11
Fotgängareolyckor (spårvagn mot person)	17	12	21	17	25	27
Påkörning (spårvagn mot föremål/anordning)	34	37	25	38	28	30

Olycksstatistik typer av olyckor



Olycksstatistik olyckor kopplat till senast lämnad hållplats

B

Bilder från spårvagnstest



Test när passagerare går framför spårvagnen

B. Bilder från spårvagnstest



Radarsensorernas montering



Radarsensorernas montering



Test när spårvagn möter annan spårvagn



Test när spårvagn möter annan spårvagn

B. Bilder från spårvagnstest



Test när bilen gör en u-sväng framför spårvagnen



Bild från inuti spårvagnen



Test för att simulera en hållplats

B. Bilder från spårvagnstest



Test för att simulera en hållplats



Test för att simulera personer som går längs spåret