



CHALMERS



Autonom kundkorg

Användning av Ultra Wideband radio och LiDAR för realtidspositionering samt objekt-detektering och kollisionsundvikning

Kandidatarbete inom System- och Reglerteknik

ROBYN BLIXTH, OSKAR EKMAN, HARALD JANHÄLL,
ALBIN JÖNSSON, RICKARD LY, HERMAN STRÅGEGÅRD

INSTITUTIONEN FÖR ELEKTROTEKNIK

CHALMERS TEKNISKA HÖGSKOLA

Göteborg 2025

www.chalmers.se

KANDIDATARBETE 2025

Autonom kundkorg

Användning av Ultra Wideband radio och LiDAR för
realtidspositionering samt objekt-detektering och kollisionsundvikning

ROBYN BLIXTH, OSKAR EKMAN, HARALD JANHÄLL,
ALBIN JÖNSSON, RICKARD LY, HERMAN STRÅGEGÅRD



CHALMERS

Institutionen för Elektroteknik
Avdelningen för System- och Reglerteknik
EENX16-25-11
CHALMERS TEKNISKA HÖGSKOLA
Göteborg 2025

Autonom kundkorg

Användning av Ultra Wideband radio och LiDAR för realtidspositionering samt objekt-detektering och kollisionsundvikning

© ROBYN BLIXTH, OSKAR EKMAN, HARALD JANHÄLL, ALBIN JÖNSSON, RICKARD LY, HERMAN STRÅGEGÅRD, 2025.

Handledare: Nikolce Murgovski, Institutionen för Elektroteknik
Examinator: Jonas Sjöberg, Institutionen för Elektroteknik

Kandidatarbete 2025: EENX-25-11
Institutionen för Elektroteknik
Avdelning för System- och reglerteknik
Chalmers Tekniska Högskola
SE-412 96 Göteborg
Telefon +46 31 772 1000

Omslagsbild: En bild på den färdiga prototypen för detta projekt.

Skriven i L^AT_EX
Göteborg 2025

Autonom kundkorg

Användning av Ultra Wideband radio och LiDAR för realtidspositionering samt objekt-detektering och kollisionsundvikning

Robyn Blixth, Oskar Ekman, Harald Janhäll, Albin Jönsson,
Rickard Ly, Herman Strågegård

Institutionen för Elektroteknik

Chalmers Tekniska Högskola

Sammanfattning

Denna rapport beskriver hur implementeringen av Light Detection and Ranging (LiDAR) och Ultra Wideband (UWB), tillsammans med enkel mekatronik, signalfiltrering och reglerteknik, tillämpats i syfte att skapa en prototyp av autonom kundkorg som kan följa efter en specifik ledare och samtidigt undvika kollision med omgivningen. Kundkorgen realtidspositionerar relativt till ledaren, personen den följer, med UWB-enheter placerade på både ledaren och kundkorgen. Kalmanfilter används för att reducera brus i mätvärdena, och PID-regulatorer används för att styra korgen till ledaren. Hinder i körvägen detekteras med LiDAR. Prototypen testades empiriskt mot en kravspecifikation, som låg till grund för utvärderingen, men även utifrån andra prestandarelaterade aspekter. Fälttester visade att prototypen uppfyllde de centrala funktionerna, men vissa utmaningar kvarstår. De främsta utmaningarna som förhindrade att samtliga krav uppfylldes var noggrannheten i UWB-positioneringen, aspekter av den mekaniska konstruktionen och brister i kollisionsundvikningen som ibland gav falska negativa och falska positiva hinder.

Nyckelord: Ultra Wideband, LiDAR, autonom kundkorg, objekt-detektering, kollisionsundvikning, realtidspositionering, Kalmanfilter, reglersystem.

Autonomous shopping basket

Use of Ultra-Wideband Radio and LiDAR for Real-Time Positioning, Object Detection, and Collision Avoidance

Robyn Blixth, Oskar Ekman, Harald Janhäll, Albin Jönsson,
Rickard Ly, Herman Strågegård

Institution for Electrical Engineering

Chalmers University of Technology

Abstract

This report describes the implementation of Light Detection and Ranging (LiDAR) and Ultra-Wideband (UWB), in combination with basic mechatronics, signal filtering, and control engineering, for the purpose of developing a prototype of an autonomous shopping cart that can follow a specific leader while simultaneously avoiding collisions with its surroundings. The cart determines its real-time position relative to the leader - the person it follows - using UWB devices on the leader and mounted on the cart. A Kalman filter is used to reduce noise in the measurements, and PID controllers are employed to steer the cart toward the leader. Obstacles in the path are detected using LiDAR. The prototype was tested empirically against a requirements specification, which formed the basis for evaluation, as well as from other performance-related aspects. Field tests showed that the prototype fulfilled the core functionalities, but some challenges remained. The main issues that prevented full compliance with the requirements were related to the accuracy of UWB positioning, aspects of the mechanical design, and deficiencies in the collision avoidance system sometimes resulting in both false positive and false negative in obstacle detection.

Keywords: Ultra-Wideband, LiDAR, autonomous shopping basket, object detection, collision avoidance, Real-Time positioning, Kalman filter, control system.

Förord

Detta kandidatarbete har genomförts med hjälp och stöd från flertalet personer och organisationer. Vi vill här rikta dem ett stort tack.

Vi vill först och främst tacka vår handledare Nikolce Murgovski för hans stöd och idéer genom hela arbetet. Vi väl även varmt tacka Gustav Malmström, som genom sitt examensarbete på SAAB i Jönköping gav oss möjligheten att få låna Ultra Widebandsensorer vilket gav oss möjlighet till att genomföra realtidspositionering. Vi vill dessutom rikta ett stort tack till föreningen ETA och dess medlemmar på Chalmers Tekniska Högskola, som varit till stor hjälp under arbetet. Till sist vill vi även tacka Maxi ICA Stormarknad Göteborg för att vi fick erhålla en kundkorg som användes till vår robot och för att vi fick möjligheten att testa roboten i deras butiksmiljö.

Robyn Blixth, Oskar Ekman, Harald Janhäll,
Albin Jönsson, Rickard Ly, Herman Strågegård

Göteborg, Maj 2025

Akronymer

Nedan följer en lista över akronymer som har använts genom hela kandidatarbetet, listade i alfabetisk ordning:

DC	Direct Current
LiDAR	Light Detection and Ranging
LOS	Line-of-sight
NLOS	Non-line-of-sight
PWM	Pulse Width Modulation
RPM	Rotationer per minut
ToF	Time of Flight
TWR	Two-Way Ranging
TDoA	Time Difference of Arrival
UART	Universal Asynchronous Receiver/Transmitter
USB	Universal Serial Bus
UWB	Ultra Wideband



Innehåll

Akronymer	vii
Figurer	xi
Tabeller	xiii
1 Inledning	1
1.1 Syfte	1
1.1.1 Kravspecifikation	2
1.1.2 Valt koncept	2
1.2 Avgränsningar	2
2 Teori	3
2.1 LiDAR	3
2.2 Ultra Wideband	3
2.3 Kalmanfilter	5
2.3.1 Predikteringsekvationer	6
2.3.2 Korrigeringsekvationer	6
2.4 Reglersystem	7
2.4.1 PID-regulator	8
3 Komponentval och konstruktion	9
3.1 Drivlina	9
3.2 Automation	10
3.2.1 Hårdvara	10
3.2.2 Mjukvara	13
3.3 Konstruktion	14
4 Implementation	17
4.1 Implementering av UWB	17
4.2 Implementering av LiDAR	18
4.3 Implementering av Kalman Filter	19
4.4 Implementering av kod	21
4.5 Styrssystem	23
4.5.1 Parametrar	23
4.5.2 Motorsignaler	23

5	Tester och Resultat	25
5.1	Precision och noggrannhet för MDEK1001	25
5.2	Positionsmätning genom UWB med Kalmanfilter	27
5.3	Positionsmätning genom UWB i cirkulär bana	29
5.4	Avståndsmätning för LiDAR-enheten	30
5.5	Mätning av avvikelse vid färd rakt fram	32
5.6	Mätning av hastighet	34
5.7	Belastningstest	35
5.8	Mätning av bromssträcka	35
5.9	Svänga runt hörn	35
5.10	Avstånd från ledaren	37
5.11	Följa ledare bland hinder	37
5.12	Test i matvarubutik	37
5.13	Resultat för resterande krav i kravspecifikationen	38
6	Diskussion	41
6.1	Testresultat	41
6.1.1	LiDAR	41
6.1.2	Ultra Wideband	41
6.1.3	Körförmåga	42
6.2	Avvikelser från planering	43
6.3	Jämförelse med tidigare kandidat- och examensarbete	44
6.4	Vidareutveckling	44
6.5	Samhälleliga och etiska aspekter	45
7	Slutsats	47
	Referenser	49
A	Kravspecifikation	III
B	Beräkning för val av motor	V
C	Kostnader för prototypens komponenter	VII

Figurer

2.1	Estimerad position för en tagg genom trilateration. Avstånden d_1, d_2 och d_3 representerar avstånden mellan ankarna och taggen. Skärningspunkten mellan cirklarna är den estimerade positionen.	4
2.2	Visualisering av det diskreta Kalman filtrets rekursiva loop som växlar mellan prediktering och korrigering.	6
2.3	Blockdiagram för ett enkelt reglersystem med återkoppling där r är referensvärdet, e är felet, w_s är störningar i systemet, w_m är störningar i sensorerna, y är processvärdet och y_m är uppmätta processvärdet.	8
3.1	CAD-modell av växellådan med drivhjul.	10
3.2	Raspberry Pi 4 Model B	11
3.3	Raspberry Pi Pico	11
3.4	HAT-MDD10 Motordrivare till Raspberry Pi	11
3.5	UWB-utvecklingskit MDEK1001.	12
3.6	Okdo Lidar Module Kit för LiDAR LD06 Raspberry Pi	12
3.7	Illustration av kommunikationsflödet mellan robotens komponenter. Pilar symboliserar kommunikation via kabel, medan streckade linjer symboliserar trådlös kommunikation.	13
3.8	Konstruktion och placering av komponenter för roboten. Undersidan ses till vänster och ovasidan ses till höger.	14
3.9	3D-printad modell av en ”självscanner” med UWB-taggen i handtag och powerbank ovanpå.	15
4.1	Fyra ankare med trilaterationscirklar, samt markerat avstånd och vinkel mellan tagg och referenspunkt.	18
4.2	LiDAR-sensors synfält i miljö med flera hinder. De blå punkterna är objekt och de gula linjerna är vägen strålen tagit. Röda linjerna markerar det närmsta punkterna i synfältet och den uppmätta distansen för dessa, samt konfidensvärdet, skrivs ut högst upp i rutan.	19
4.3	Ofiltrerade x-koordinater jämförda med Kalmanfiltrerade x-koordinater för olika värden på processbruset Q_{var}	21
4.4	Flödesschema för huvudloopen i kontrollsystemet, som visar hur roboten navigerar mot målet och undviker hinder.	22
5.1	Verklig position (färgade kryss) jämfört med uppmätt position (färgade prickar) för 10 olika positioner.	26

5.2	Exempel på uppställning för avståndsmätning med hjälp av UWB med LOS (vänster) och NLOS (höger).	28
5.3	Verklig position (färgade kryss) jämfört med filtrerad, uppmätt position (färgade prickar) för 10 olika positioner runt roboten.	28
5.4	Uppställning för test av UWB-mätning i cirkulär bana.	30
5.5	Verklig bana (heldragen svart linje) jämfört med filtrerad, uppmätt position (blå prickar) för gående person med UWB-taggar.	30
5.6	Exempeluppställningar för test av LiDAR-enheten.	31
5.7	Grafisk presentation av resultatet från LiDAR-testet där även avvikelsen mellan verklighet och uppmätt distans är presenterad.	31
5.8	Uppställning inför test vid mätning av avvikelse vid färd rakt fram.	32
5.9	Resultat från testning av avvikelse i rak färdriktning, digitalt återskapat, där varje enskild färg markerar vägen roboten tog under en iteration av testet och centrumlinjen är markerad i svart.	34
5.10	Testuppställning vid svängning av hörn.	36
5.11	Körväg av robot för mätning 2 vid test med tagg-bärare bakom hörn vid start.	36
5.12	Ett fotografi tagen från testet i matvarubutik.	38
5.13	Sammanställning av vilka krav i kravspecifikationen som uppfyllts där grönt innebär uppfyllt, rött innebär ej uppfyllt, gult innebär delvis uppfyllt och blå innebär ej testat.	40
A.1	Kravspecifikation	IV
B.1	Friläggning av kundkorg med hjul placerade för differentialstyrning, samt de krafter som uppstår vid rörelse.	V

Tabeller

2.1	Notation av vektorer och matriser i Kalmanfilter.	6
5.1	Vänster tabell visar medelvärdet av medelfel och standardavvikelse i x- och y-riktning, medan höger visar medianen av dessa.	27
5.2	Feluppskattning för LOS. Vänster tabell visar medelvärdet av medelfel och standardavvikelse i x- och y-riktning, medan höger visar medianen av dessa.	29
5.3	Feluppskattning för NLOS. Vänster tabell visar medelvärdet av medelfel och standardavvikelse i x- och y-riktning, medan höger visar medianen av dessa.	29
5.4	Mätresultat från mätning av avvikelser vid färd rakt fram, där största avvikelsen [cm] över en färdad sträcka på 282 centimeter noterats. Positivt värde är förskjutning åt höger.	33
5.5	Resultat från tester av robotens förmåga att svänga runt hörn.	36
C.1	Komponentlista med inköpspriset för varje komponent samt det sammanlagda priset för alla komponenter. Samtliga priser är uttryckt i SEK.	VII

1

Inledning

I en alltmer automatiserad värld utvecklas ständigt produkter och föremål för att underlätta och effektivisera vardagen. Redan idag har matvarubutiker introducerat smarta lösningar för sina kunder. Exempelvis finns det kundvagnar som kan låsas upp med en app på mobilen istället för med mynt, samt kundvagnar som med en skärm på sig visar olika erbjudanden från butiken [1][2]. Ännu mer utvecklade kundvagnar använder artificiell intelligens och maskininlärning för att scanna det kunderna handlar direkt när produkten läggs ner i kundvagnen, och där betalning sker direkt på kundvagnen [3].

Alla ovannämnda exempel ger möjlighet till förbättrad kundupplevelse men ingen lösning inkluderar hur kundvagnar eller kundkorgar förflyttar sig, kunden måste fortfarande dra eller bära korgen. Det finns enstaka exempel på kundvagnar som körs autonomt, dock är dessa exempel endast illustrerade i korta och väldigt redigerade videor som speglar möjliga framtida matvarubutiker [4]. En självkörande kundkorg skulle kunna följa efter kunden genom butiken, kunna anpassa sin hastighet och svänga utan att kollidera med hinder som uppkommer. När korgen körs helt autonomt, utan att kunden behöver greppa den kan inkluderingen av fler kunder öka. Det kan gälla äldre, skadade personer eller andra som på något sätt har svårigheter att bära eller dra en kundkorg efter sig.

Den autonoma kundvagnen har utforskats av kandidatarbeten tidigare. År 2022 vidareutvecklades en prototyp från ett tidigare års arbete [5]. Stora framsteg i mjukvara gjordes genom nytt regler- och kollisionssundvikningssystem med hjälp av en kamera och artificiell intelligens. Roboten lyckades följa en person, men vissa förbättringsområden kvarstod. Prototypen kunde inte urskilja mellan olika individer och följde efter godtycklig människa. Problem i följningsförmågan uppstod då ledaren försvann ut från synfältet, exempelvis runt hörn. Vidare led en asymmetrisk konstruktion och sneda hjul till svårigheter att köra rakt fram. Slutligen saknade den förmåga att navigera runt hinder.

1.1 Syfte

Syftet med projektet är att utveckla en prototyp av en autonom kundkorg som ska kunna följa efter en utvald person utan att tappa bort sig eller börja följa ett oönskat mål. Den skall även på egen hand detektera, undvika och navigera runt hinder.

1.1.1 Kravspecifikation

En kravspecifikation har tagits fram för att konkretisera vilka mål som ska uppfyllas utifrån syftet (se Figur A.1 i Appendix A). Större fokus har lagts på prototypens förmåga att navigera i en butiksmiljö. De tester som gjorts och vilket resultat de gav beskrivs mer i Kapitel 5.

1.1.2 Valt koncept

Med kravspecifikationen i åtanke valdes det mest önskvärda konceptet som tillämpas i den slutliga prototypen. En kort beskrivning ges nedan:

Prototypen ska vara i samma storleksordning som en handdragen kundkorg. Ett batteri driver de två likströmsmotorerna som i sin tur driver respektive hjul. Svängning möjliggörs med differentialstyrning och bromsning sker genom motorbroms. Drivlinan styrs av en dator på roboten som får insignaler från Ultra Wideband (UWB) och LiDAR sensorer på fordonet. UWB sensorer ger position till en specifik individ som ska efterföljas och en LiDAR-sensor bestämmer avstånd till hinder runtomkring fordonet. Detaljerad teoretisk beskrivning av sensorerna ges i avsnitt 2.1, 2.2.

1.2 Avgränsningar

För att undvika att projektet blev allt för omfattande togs nedanstående avgränsningar fram:

- Ej genomföra analys för produktionskostnader kopplade till form och funktion.
- Endast anta att produkten används i en inomhusmiljö utan stora nivåskillnader eller ojämnt underlag.
- Ta fram konceptförklaringar men inte ritningar.
- Endast undersöka möjligheter för rörelse av vagnen, ej andra funktioner som exempelvis varuigenkänning eller betalssystem.
- Ej undersöka hur eventuella batterier eller andra energikällor laddas för den färdiga produkten.

Utöver avgränsningarna fanns det även en budget på 5000 SEK som sattes av Institutionen för Elektroteknik.

2

Teori

I detta kapitel presenteras teorin bakom UWB, LiDAR, Kalmanfilter och reglersystem. Kapitlet syftar till att redogöra för hur roboten bestämmer sin position med UWB, hur LiDAR detekter objekt, hur Kalmanfilter hjälper till att dämpa mätbrus och hur reglersystem reglerar robotens rörelse.

2.1 LiDAR

LiDAR är en förkortning av "Light Detection And Ranging" och innefattar all teknologi som mäter avstånd med hjälp av ljus [6]. Det finns många olika former av LiDAR men i detta projekt användes en "2D planar scanning LiDAR". Denna LiDAR-enhet kan mäta avstånd i alla riktningar runt sig i det horisontella planet, med andra ord en 360 graders vy. Tekniken bygger på att en laserstråle sänds ut från enheten och träffar en yta som reflekterar tillbaka delar av signalen. LiDAR-enheten beräknar det avstånd till den träffade ytan genom att mäta ToF, som använder följande formel förenklad [7][8]

$$distan\ s = \frac{c \cdot T}{2}, \quad (2.1)$$

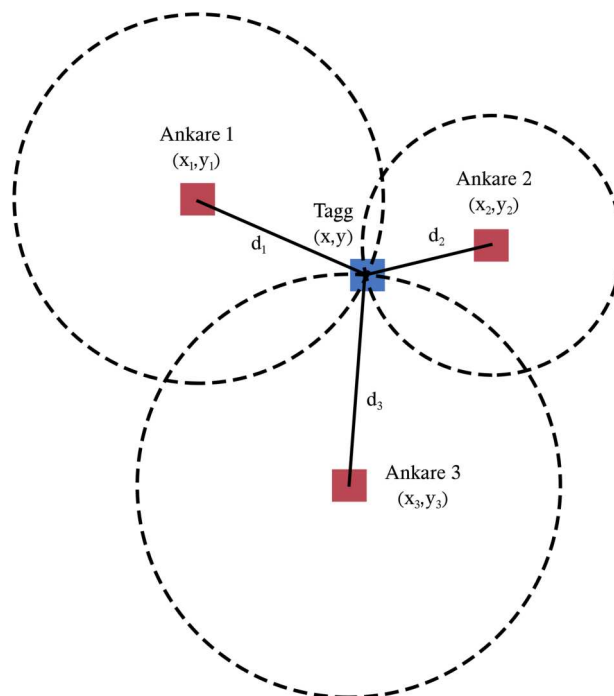
där c [m/s] är ljusets hastighet och T är tiden, i sekunder, det tar mellan att lasern skickats och att den mottagits. Enheten för kvoten blir då meter. LiDAR-enheten roterar kontinuerligt och gör flera mätningar per varv och med insamlad mätdata är det möjligt att skapa en tvådimensionell bild av omgivningen omkring LiDAR-enheten.

2.2 Ultra Wideband

UWB är en radioteknik som enligt FCC (Federal Communications Commission) sträcker sig över ett brett frekvensområde, från 3,1 GHz till 10,6 GHz och med en bandbredd nära eller över 500 MHz [9]. Tekniken sänder korta pulser vilket möjliggör snabb dataöverföring med låg strömförbrukning [10]. På grund av sin breda bandbredd har UWB en mycket låg överföringseffekt, vilket gör att den inte stör andra smalbandiga teknologier som UWB överlappar med [11]. För exempelvis Wi-Fi uppfattas UWB mest som bakgrundsbrus, även om detta bland annat använder frekvensband på 5 eller 6 GHz [12].

För att använda UWB i ett system för realtidspositionering (förkortat RTLS på engelska), används ankare och taggar för att estimeras en position [9]. Ett ankare placeras på en känd koordinat och en tagg kopplas på den person eller det objekt vars position som ska spåras. För att få en position i 2D-planet krävs tre ankare, medan fler ankare ger en högre mätnoggrannhet [13]. Det finns mer än en beräkningsmetod för att estimeras en position för taggen, till exempel Two-Way Ranging (TWR), Time Difference of Arrival (TDoA) och Reverse TDoA [9]. För de UWB-moduler som använts i detta projekt beräknas TWR med förprogrammerad inbäddad kod [14]. Likväl kommer denna beräkningsmetod förklaras för att ge en matematisk förståelse för trianguleringen som tillämpas i projektet.

Figur 2.1 visar hur tre ankare med sina kända positioner används för att estimeras en position för taggen. Taggen sänder ut en kort puls som sedan fångas upp av alla ankarna [9]. Därefter svarar ankarna genom att skicka en puls tillbaka till taggen, och genom att taggen registrerar rundresetiden, samt ankarnas bearbetningstid, kan taggen beräkna avståndet till vardera ankare likt ToF, enligt ekvation (2.1), där T är rundresetiden subtraherat med ankarens bearbetningstid. Vid mätning mellan ett ankare och en tagg erhålls ett avstånd på l längdenheter, men inte en riktning. Varje ankare kan då ses som mittpunkten på varsin cirkel med radien l längdenheter. Med trilateration bestäms positionen för taggen som den punkt där alla cirklar skär varandra [15]. Notera att Figur 2.1 illustrerar ett idealt fall, där taggen beräknats till en viss punkt. I realistiska fall möts inte cirklarna alltid vid en punkt, utan de skapar ett litet område, vilket ger en osäkerhet i den estimerade positionen.



Figur 2.1: Estimerad position för en tagg genom trilateration. Avstånden d_1, d_2 och d_3 representerar avstånden mellan ankarna och taggen. Skärningspunkten mellan cirklarna är den estimerade positionen.

Eftersom taggens position ligger i skärningspunkten mellan cirklarna, kan cirkelns ekvation användas för att beräkna de okända koordinaterna (x, y) . Ekvationen formuleras på följande sätt

$$d_i = \sqrt{(x - x_i)^2 + (y - y_i)^2}, \quad (2.2)$$

där index i representerar respektive ankare. Därefter kvadreras båda sidor och högerledet utvecklas för alla $i = 1, 2, \dots, n$, vilket ger

$$d_i^2 = x^2 - 2x_i x + x_i^2 + y^2 - 2y_i y + y_i^2. \quad (2.3)$$

Det finns olika metoder för att lösa ut x och y där ett sätt är att subtrahera ekvationerna med varandra. Detta görs för att eliminera termerna x^2 och y^2 . För varje ekvation i som subtraheras med varje ekvation j fås

$$d_i^2 - d_j^2 = (x^2 - 2x_i x + x_i^2 + y^2 - 2y_i y + y_i^2) - (x^2 - 2x_j x + x_j^2 + y^2 - 2y_j y + y_j^2), \quad (2.4)$$

vilket sedan kan förenklas till

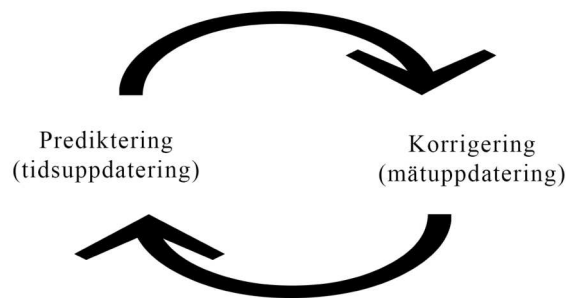
$$d_i^2 - d_j^2 = -2x(x_i - x_j) + x_i^2 - x_j^2 - 2y(y_i - y_j) + y_i^2 - y_j^2. \quad (2.5)$$

När $n = 3$ fås tre ekvationer från Ekvation (2.3), där den första subtraheras med den andra och den tredje, samt den andra med den tredje. Subtraktionen genererar linjära ekvationer som den i Ekvation (2.5). För n punkter fås $\frac{n(n-1)}{2}$ linjära ekvationer som används för att vidare lösa ut x och y . När $n = 3$ fås alltså 3 linjära ekvationer. Det finns ytterligare olika tillvägagångssätt för att vidare lösa ut x och y , men dessa metoder tas inte upp i denna rapport.

2.3 Kalmanfilter

Vid användning av sensorer uppstår brus som en naturlig följd. Brushantering är särskilt viktigt för detta projekts ändamål för att säkerställa en korrekt och precis efterföljning av fordonets mål. Adaptiva filter är en variant av filter som är användbar i en dynamisk miljö, vilket är relevant för den självkörande kundkorgen. På grund av filtrets förmåga att justera dess parametrar med avseende på en förlustfunktion, kan den användas i miljöer med okända variabler [16].

Kalmanfilter är ett adaptivt filter som är särskilt lämpat för efterföljande fordon vilket drogs som slutsats i tidigare examensarbete som använde UWB i liknande ändamål [17]. Därför kommer endast detta filter utvärderas för UWB i denna rapport. Filtret uppdaterar parametrar rekursivt från tidigare mätdata och framtida tillstånd predikteras (se Figur 2.2) [18]. I andra termer så estimerar prediktionsekvationer nästa tillstånd och korrigeringskvationer ger feedback om hur rätt estimeringen är för att bilda en bättre prediktering. Detta kan göras även om delar av det modellerade systemet är okänt. Nedan följer en mer överskådlig förklaring av ekvationerna, men härledningen av dessa kommer inte förklaras. Notationer av variabler sammanställs i Tabell 2.1.



Figur 2.2: Visualisering av det diskreta Kalman filtrets rekursiva loop som växlar mellan prediktering och korrigering.

Tabell 2.1: Notation av vektorer och matriser i Kalmanfilter.

Term	Namn	Dimension
x	Tillståndsvektor	$n_x \times 1$
z	Mätningvektor	$n_z \times 1$
F	Tillståndsovergångsmatris	$n_x \times n_x$
P	Skattad kovarians	$n_x \times n_x$
Q	Processbrusets kovarians	$n_x \times n_x$
R	Mätbrusets kovarians	$n_z \times n_z$
H	Observationsmatris	$n_z \times n_x$
K	Kalmanvikt	$n_x \times n_z$
k	Diskret tidsindex	

2.3.1 Predikteringsekvationer

I Ekvation (2.6) predikteras nästa tillstånd av processen som mäts. Detta görs genom att multiplicera tidigare estimerat tillstånd \hat{x}_{k-1} med tillståndsovergångsmatrisen F . Matrisen beskriver hur tidigare tillstånd relaterar till nästa tillstånd. Till exempel: om tillståndsvektorn innehåller position, hastighet och acceleration av ett fordon, innehåller tillståndsovergångsmatrisen rörelsekvationer som beskriver framtida rörelse [19].

$$\hat{x}_k^- = F\hat{x}_{k-1} \quad (2.6)$$

Därefter i Ekvation (2.7) predikteras skattad kovarians P , det vill säga osäkerheten i skattningen, baserat på tidigare estimerad osäkerhet P_{k-1} och processbrusets kovarians Q [18]. Q kan betraktas som osäkerheten i modellen som beskriver processen. Baserat på tidigare exempel kan det uppstå från luftfriktion eller vind som ger ett fel mellan verklig hastighet och estimerad hastighet [20].

$$P_k^- = FP_{k-1}F^T + Q \quad (2.7)$$

2.3.2 Korrigeringsekvationer

Först beräknas Kalmanvikt K_k i Ekvation (2.8), vilket bestämmer påverkan som den nya mätningen har på skattningen [18]. Denna vikt beror på osäkerheten i skatt-

ningen P_k^- och osäkerheten i mätningen R . Notera att ett högre brus i mätningarna ger större R och därmed dämpas effekten av de nya uppmätta värdena. Likartat leder stora osäkerheter i skattningen till ett stort P och nya mätvärden viktas högre än gamla prediktioner.

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1} \quad (2.8)$$

I nästa steg som visas i Ekvation (2.9), uppdateras tillståndsestimeringen \hat{x}_k^- med de uppmätta värdena i mätningsektorn z_k . Differensen mellan uppmätt värde och predikerat värde viktas med Kalmanvikt. Observationsmatrisen H beskriver hur tillståndet relaterar till mätningarna som observeras.

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-) \quad (2.9)$$

Även skattningens osäkerhet uppdateras i Ekvation (2.10) som reflekterar hur osäkerheten minskar efter att nya mätningar inkorporeras.

$$P_k = (I - K_k H) P_k^- \quad (2.10)$$

2.4 Reglersystem

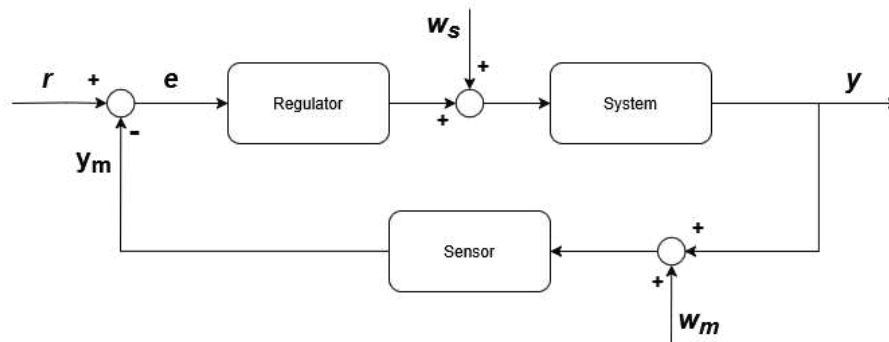
Ett reglersystem är ett system som automatiskt justerar en process för att hålla ett visst önskat tillstånd, så kallad börvärde. Det fungerar genom att mäta ett faktiskt värde (processvärde), jämföra det med börvärdet, och utifrån skillnaden (reglerfelet) påverka processen via en reglerstrategi, till exempel en PID-regulator [21]. Målet är att minimera felet och stabilt uppnå det önskade resultatet, trots yttre störningar eller förändringar i systemet.

Då en relativ position från UWB-ankarna till taggen erhålls i kartesiska koordinater (x, y) kan dessa omvandlas till polära koordinater (r, θ) där r är avstånd och θ är vinkel och beräknas med följande formler,

$$r = \sqrt{x^2 + y^2}, \theta = \begin{cases} \arctan\left(\frac{y}{x}\right) & x > 0 \\ \arctan\left(\frac{y}{x}\right) + \pi & x < 0, y \geq 0 \\ \arctan\left(\frac{y}{x}\right) - \pi & x < 0, y < 0 \\ \frac{\pi}{2} & x = 0, y > 0 \\ -\frac{\pi}{2} & x = 0, y < 0 \end{cases} \quad (2.11)$$

De polära koordinaterna kan enskilt användas som parametrar att reglera för att en robot ska följa efter ett mål. Avståndsparametern r regleras genom att felet i börvärde och processvärde matas in i en regulator som behandlar felet utifrån olika parametrar som kan justeras beroende på önskat resultat. De nya värdet kallas för styrsignal och matas sedan in i systemet som genom en ny funktion ändrar avståndet, så att ett nytt processvärde erhålls. Med ett nytt processvärde kan nu ett nytt fel beräknas och cykeln upprepas tills dess att det börvärdet är det samma som processvärdet. Med samma princip regleras vinkelparametern θ .

Störningar som reglersystemet utsätts för betecknas w . Störningarna ger upphov till brus i reglersystemets signaler. I Figur 2.3 presenteras en enkel modell av hur ett reglersystem kan modelleras med ett block diagram.



Figur 2.3: Blockdiagram för ett enkelt reglersystem med återkoppling där r är referensvärdet, e är felet, w_s är störningar i systemet, w_m är störningar i sensorerna, y är processvärdet och y_m är uppmätta processvärdet.

2.4.1 PID-regulator

En väldigt vanlig regulator som möjliggör stor flexibilitet i omvandling av felet till styrsignal är en PID-regulatorn. Den bygger på tre komponenter som vars påverkan på styrsignalen justeras genom att var sin konstant som bestäms utefter önskade kvalitéer hos styrsignalen. Förhållande mellan konstanterna påverkar kraftigt hur styrsignalen utformas. Den matematiska formeln för regulatorn är,

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt}, \quad (2.12)$$

där $u(t)$ är styrsignalen, $e(t)$ är felet och K_p , K_i , K_d är konstanter [21].

3

Komponentval och konstruktion

I detta kapitel redogörs först valet av de komponenter som användes till roboten. Komponentvalet är uppdelat i två kategorier, Drivlina och Automation. Vidare avslutas kapitlet med en beskrivning av robotens konstruktion och placering av komponenterna. Totalpriset för de använda komponenterna samt de individuella priserna presenteras i Appendix C.

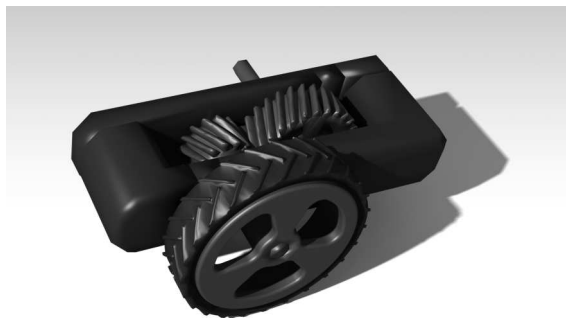
3.1 Drivlina

För att roboten skulle kunna förflytta sig utan externa krafter krävdes en drivlina. I detta delkapitel beskrivs hur motorerna dimensionerades utefter kravspecifikationen, samt vilket batteri som valdes för att driva motorerna.

I Appendix B redogörs de fullständiga beräkningarna för valet av motor. Notera att två motorer har använts i projektet, en för varje drivhjul. Beräkningarna visade att under kontinuerlig drift krävs ungefär 7,15 W per motor, ett moment på 0,23 Nm och en vinkelhastighet på 294,9 RPM. Enligt tumregler för motordimensionering bör en motor ha 1,5–2 gånger den nominella effekten som krävs i drift [22], vilket ger

$$P_{\text{nominell}} \geq 1,5 \cdot 7,15 \approx 10,73 \text{ W.}$$

En motor med moment och vinkelhastighet precis över de beräknade fanns vid tillfället inte att få tag på. För robotens framdrivning valdes därför en DC-motor med nominell effekt på 19,8 W, nominellt varvtal på 143 RPM och nominellt moment på 0,59 Nm [23]. Den valda motorn har tillräcklig nominell effekt (19,8 W > 10,73 W). För att anpassa vald motor till de krav som beräknats fram för moment och vinkelhastighet, används en 3D-printad växellåda med en uppväxling på 1:2, vilket fördubblade varvtalet till 286 RPM, vilket var nära det önskade på 294,9 RPM och ansågs vara acceptabelt. Det resulterande momentet halverades till 0,286 Nm vilket var lite högre än det önskade på 0,23 Nm, men gav en marginal för eventuella friktionsförluster. Även drivhjulen är 3D-printade och däcket skrevs ut i mjuk TPU-plast för bättre grepp. En CAD-modell av ett drivhjul monterat på växellådan visas i Figur 3.1.



Figur 3.1: CAD-modell av växellådan med drivhjul.

För att driva framdrivningen valdes ett blybatteri på 12 V och 7,0 Ah som energikälla [24]. Spänningsnivån på 12 V ligger inom intervallet som motorn hanterar.

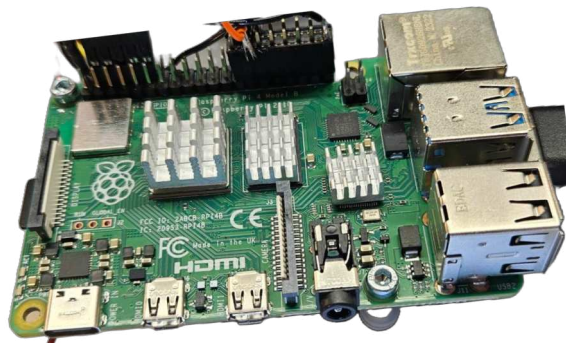
3.2 Automation

Automatiseringen var det mest omfattande implementationsområdet i projektet eftersom det inkluderade ingripande ämnen som reglering, följningsförmåga och kollisionssundvikning. För att implementera dessa funktioner krävdes olika komponenter, som beskrivs i nästkommande avsnitt, följt av ett avsnitt som kort beskriver den mjukvara som användes.

3.2.1 Hårdvara

I detta avsnitt beskrivs valet av de komponenter som utgjorde fordonets styrsystem och sensorer. Därefter beskrivs respektive komponents ändamål i projektet, samt hur kommunikationen mellan komponenterna såg ut.

Raspberry Pi 4 Model B (se Figur 3.2) är en kompakt enkortsdator och är ett populärt val i hobbyprojekt och prototypframställning [25]. Datorn är utrustad med General Purpose Input Output (GPIO) pins som kan kopplas till annan elektronisk hårdvara som sensorer vilket möjliggör inbäddad programmering. Dess beräkningskraft gör den lämpad till rollen som centralenhet som behandlar information från sensorer för att sedan skicka ut styrsignaler.



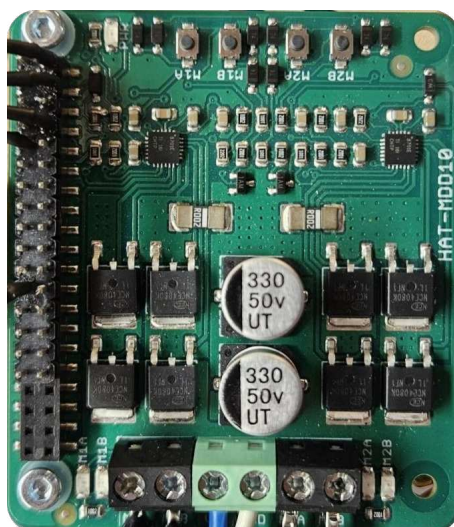
Figur 3.2: Raspberry Pi 4 Model B

Raspberry Pi Pico är en mikrokontroller med programmerbara GPIO pins som används för enklare beräkningar och styrning (se Figur 3.4) [26]. I detta projekt styr den ström till motorerna beroende på styrsignal från centralenheten.



Figur 3.3: Raspberry Pi Pico

Hat-MDD10 är en två-kanals motorkontroller som tar en PWM-signal och en hög/låg-signal per kanal för att kontrollera hastigheten samt riktningen på motorerna. Denna motorkontroller hanterar strömmar upp till 10 A och spänningar upp till 24 V. I detta projekt antog den rollen som förstärkare för den Raspberry Pi Pico som används.



Figur 3.4: HAT-MDD10 Motordrivare till Raspberry Pi

MDEK1001 (se Figur 3.5) är ett utvecklingskit av Qorvo som erbjuder en nybörjarvänlig ingång till framtagning av UWB-applikationer [27]. Utvecklingsbrädorna kan enkelt konfigureras till att vara antingen ankare eller tagg med hjälp av inbäddad fast programvara. Efter nätverket av UWB-moduler har konfigurerats används triangulering för att automatiskt beräkna taggens position som kan överföras med Universal Serial Bus (USB) till dator. Utvecklingsbrädan baseras på DWM1000 modulen som har en radio-antenn designad för lokalisering av objekt i realtid till en precision av 10 cm [28].



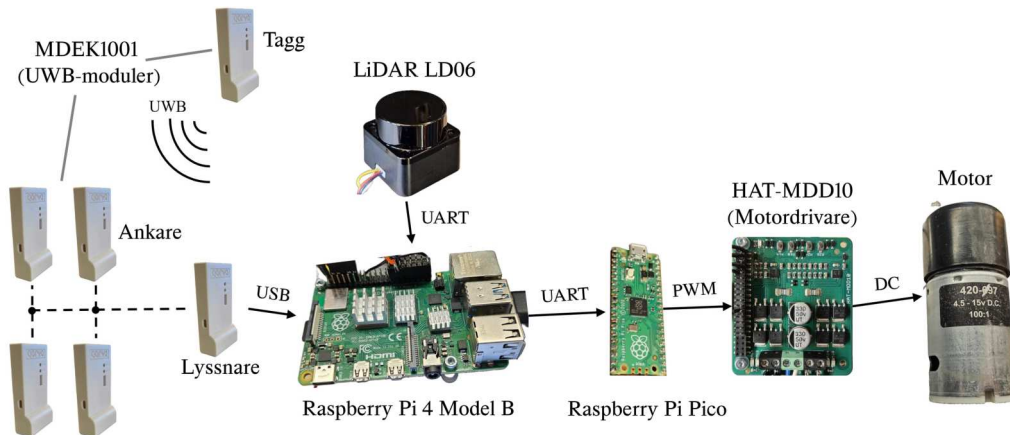
Figur 3.5: UWB-utvecklingskit MDEK1001.

LiDAR LD06 (se Figur 3.6) är en ”planar scanning LiDAR” från OKDO [7]. Modulen möjliggör för enkel anslutning till Raspberry Pi och kommunicerar via Universal Asynchronous Receiver/Transmitter (UART), som är en metod för att omvandla parallell data till seriell data [29]. LiDAR-enheten användes i detta projekt för att identifiera avståndet till väggar och andra objekt i omgivningen, för att sedan låta Raspberry Pi 4 Model B skicka styrsignal till motorer att stanna eller svänga om kollision ska undvikas.



Figur 3.6: Okdo Lidar Module Kit för LiDAR LD06 Raspberry Pi

Figur 3.7 illustrerar kommunikationsflödet mellan de ovannämnda komponenterna. En av UWB-modulerna konfigurerades till att vara tagg som med UWB-signaler kommunicerar med fyra andra moduler som konfigurerades till ankare. Efter att en position estimerats av taggen får den sista UWB-modulen, som är konfigurerad som lyssnare, denna information. En lyssnare möjliggör informationsöverföring från tagg till Raspberry Pi 4 utan att en kabel är kopplad mellan dem. Istället överför USB positionsmätning från lyssnaren till Raspberry Pi 4. LiDAR-modulen skickar även sin data via UART till Raspberry Pi 4. Vidare överförs data från Raspberry Pi 4 till Raspberry Pi Pico via UART, som i sin tur sänder en Pulse Width Modulation signal (PWM-signal) till motordrivaren. PWM avgör hur stor andel av tiden då DC-ström mellan motordrivare och motor är hög, alltså regleras effektiv spänning. Motorerna får därmed varierande effekt beroende på information från sensorerna. Samtliga komponenter strömförses av tidigare nämnda batteri, med undantag för UWB-taggen. För mer förklaring kring hur mjukvaran på Raspberry Pi 4 bearbetar data från båda sensorerna, samt hur mjukvaran på Raspberry Pi Pico skickar signaler till motordrivaren, hänvisas till Kapitel 4.



Figur 3.7: Illustration av kommunikationsflödet mellan robotens komponenter. Pilar symboliserar kommunikation via kabel, medan streckade linjer symboliserar trådlös kommunikation.

3.2.2 Mjukvara

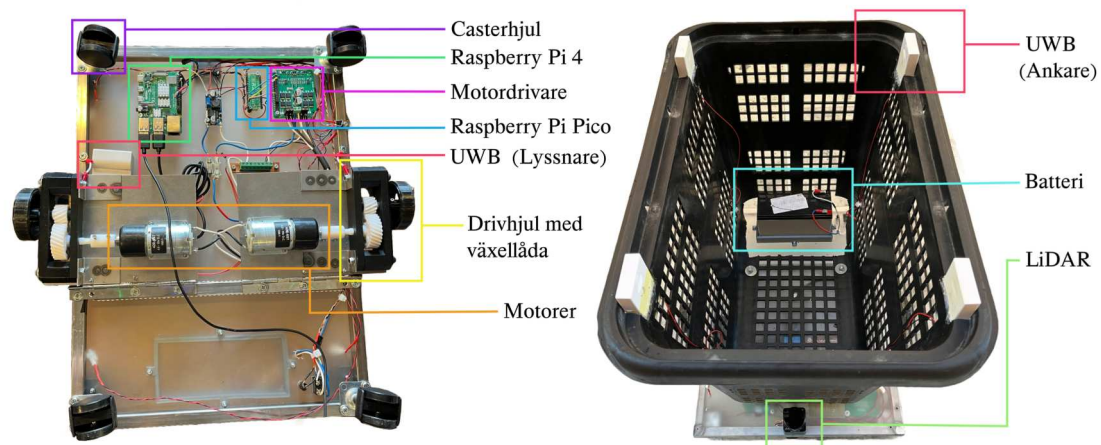
Mjukvara som använts i projektet har haft en betydande inverkan på slutresultatet. För att säkerställa ett bra resultat inom projektets tidsram valdes i första hand programvara som gruppmedlemmarna tidigare använt, där detta var applicerbart, även om denna programvara inte är lika optimerad som annan programvara.

Den beräknande och datahanterande enheten för roboten är Raspberry Pi 4 Model B, vilket är en dator, och behöver därför ett operativsystem. Raspberry Pi OS är ett linux-baserat operativsystem som är anpassat för Raspberry Pi av olika modeller. Operativsystemet erbjuder en användarvänlig GUI (Graphical User Interface), som några av gruppmedlemmarna använt tidigare, och är samtidigt resurssnålt i förhållande till andra alternativ [30].

Python användes eftersom det är ett programmeringsspråk som de flesta i gruppen är bekanta med. Fördelen med Python är att det är ett programmeringsspråk som har stor tillgång till offentligt publicerat material som kan vara användbart för projektet.

3.3 Konstruktion

Konstruktionen av roboten utformades med hjälp av de önskemål och krav som ställdes i kravspecifikationen. Dessa önskemål och krav handlade om robotens dimensioner, vikt och att komponenterna inte skulle synas. I Figur 3.8 visas robotens undersida (till vänster) och ovansida (till höger), där placering för respektive komponent markerats. Notera att robotens framsida är riktad uppåt på den vänstra bilden och nedåt på den högra bilden.



Figur 3.8: Konstruktion och placering av komponenter för roboten. Undersidan ses till vänster och ovansidan ses till höger.

Ramen för roboten skapades av fyrkantströr i stål för att ge en robust och tålig grund. En plastskiva monterades på ramen för att skapa en monteringsyta för komponenterna. Drivhjulen med dess växellådor, placerades i mitten av båda långsidorna, och kopplades till varsin motor. Placeringen av drivhjulen, tillsammans med differentialstyrningen, gör att roboten kan svänga runt sin egen axel, vilket möjliggör svängning även vid trånga ytor. Ramen skapades för att passa till motorer som inte krävde extra växling, vilket gjorde att växellådorna med drivhjul sticker utanför ramen. I hörnen, undertill på roboten, monterades caster-stödhjulen för ökad stabilitet. Raspberry Pi 4, Raspberry Pi Pico, motordrivaren och den UWB-modul som agerade som lyssnare, monterades ovanför de skivor som motorerna var monterade på. Skivorna var fästa till en bjälke, av samma slag som ramen, med gångjärn. Dessa belastades sedan med var sin tryckfjäder och tillsammans uppgav de fjädring i drivaxlarna. Korgen monterades på ovansidan. Framför korgen placerades LiDAR-enheten eftersom vyn framtill var av störst intresse. De fyra UWB-modulerna som konfigurerades till ankare monterades på korgens topp, nära dess hörn. Ett hål gjordes i korgen för

att skapa plats för batteriet. För att uppnå rätt spänningar för komponenterna användes en DC-DC omvandlare och en linjär spänningsregulator. De två sistnämnda är dock inte markerade i Figur 3.8. En av- och på-knapp implementerades också mellan batteriet och resten av systemet, för att enkelt kunna stänga av och sätta på systemet utan att koppla bort batteriet manuellt. Av- och på-knappen syns inte i Figur 3.8. Prototypens slutgiltiga dimensioner blev 460 mm i bredd, 540 mm i djup och 490 mm i höjd.

En modell av en scanner 3D-printades för att efterlikna känslan av att hålla i en självscanner med UWB-modulen inuti. Taggen placerades i handtaget och strömförsörjdes av en powerbank som placerades ovanpå (se Figur 3.9).



Figur 3.9: 3D-printad modell av en "självscanner" med UWB-tag i handtag och powerbank ovanpå.

4

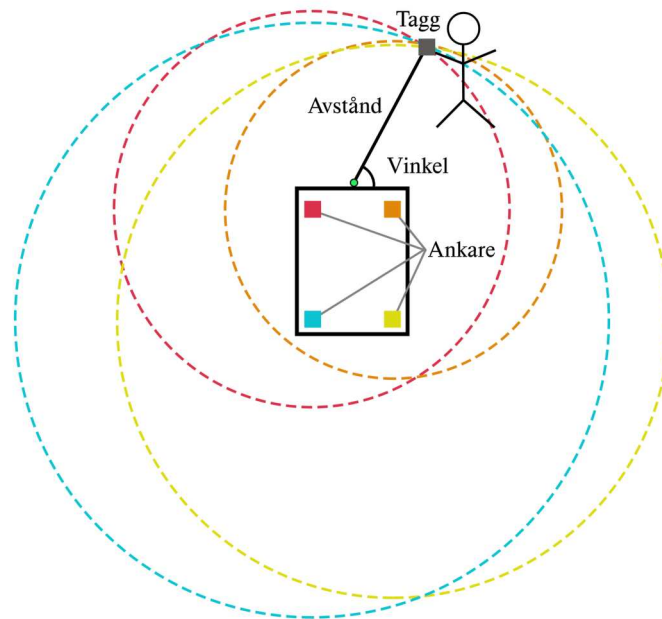
Implementation

I detta kapitel förklaras hur UWB-modulerna och LiDAR-enheten implementerades, samt hur Kalmanfiltret skapades. Vidare ges en överskådlig beskrivning om hur koden som är skriven fungerar, samt en förklaring till hur kollisionsundvikningen fungerar. Till sist ges förklaringar till hur olika parametrar fastställdes.

4.1 Implementering av UWB

Implementeringen av UWB-modulen MDEK1001 var problemfri. En app skapad av företaget själva laddades ner till en Android-enhet, där både ankarna, taggen och lyssnaren kunde anslutas. Figur 4.1 visar placeringen av ankarna, med avstånd och vinkel från referenspunkten till taggen. Referenspunkten bestämdes till att vara punkten precis framför LiDAR-enheten. Ankarna i detta projekt placerades i en rektangel, lite innanför ramens kant. Notera att ankarna inte hade samma höjd som ramen, utan placerades högre upp. Figur 3.8 visar tydligt placeringen i höjddled för ankarna.

Genom appen ställdes position in för ankare manuellt. För att bestämma var i koordinatsystemet som respektive ankare befann sig, mättes avståndet mellan referenspunkten och varje ankare i x -led och y -led. För att få ut data om taggens position användes egenskriven kod i Python som kommunicerar med UWB-modulerna via UART. Koden byggde på modulernas inbyggda mjukvara som använde trilateration, vilket beskrivs i avsnitt 2.2. Från positionen beräknades ett avstånd från referenspunkten till den person som höll i taggen, samt vinkel mellan tagg och referenspunkt.

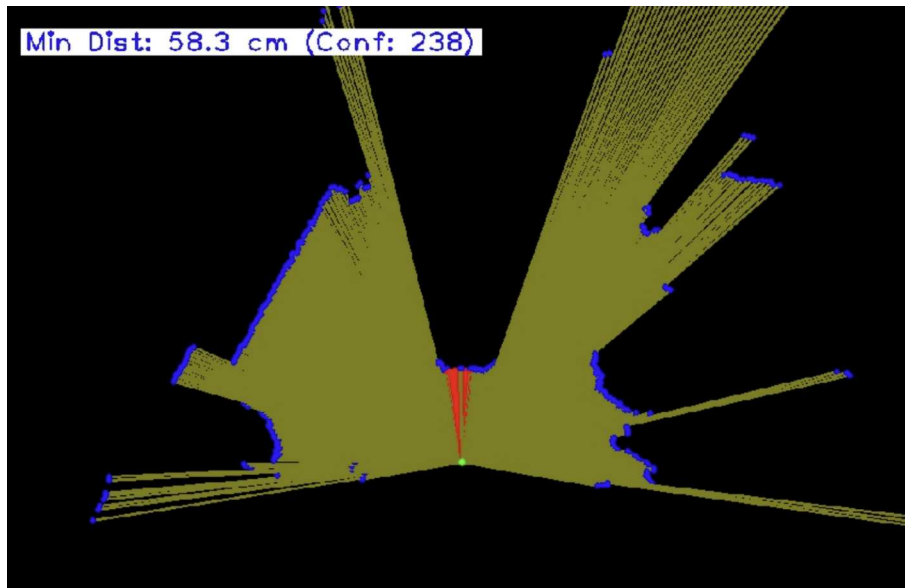


Figur 4.1: Fyra ankare med trilaterationscirklar, samt markerat avstånd och vinkel mellan tagg och referenspunkt.

4.2 Implementering av LiDAR

För att möjliggöra avståndsmätning och hinderdetektering integrerades en LiDAR-sensor på roboten. Den kommunicerar med Raspberry Pi 4 via en seriell anslutning (UART). Sensorn monterades längst fram på roboten för att maximera synfältet framåt, men detta medförde också vissa begränsningar då sensorn endast kunde se framåt och därmed inte detekterade objekt bakom sig.

Ett GitHub-repository [31] skrivet i Python användes för att hantera signalerna från LiDAR-enheten. En LiDAR gör det möjligt att se omgivningen i 360 grader, men eftersom att syftet var att upptäcka hinder framför roboten, samt att korgen som monterades bakom LiDAR-enheten hade setts som ett hinder, minskades vyn till 210 grader framåt. Ett konfidensvärde följde med varje mätning. Med ett konfidensfilter var det sedan möjligt att ignorera mätningar som gav ett konfidensvärde under en justerbar gräns. Denna implementation utvärderades ej. I Figur 4.2 visas LiDAR-enhetens synfält, och markerat i rött är det objekt som ligger närmast enheten, vars avstånd till LiDAR-enheten också går att avläsa.



Figur 4.2: LiDAR-sensorns synfält i miljö med flera hinder. De blå punkterna är objekt och de gula linjerna är vägen strålen tagit. Röda linjerna markerar det närmsta punkterna i synfältet och den uppmätta distansen för dessa, samt konfidensvärdet, skrivs ut högst upp i rutan.

4.3 Implementering av Kalman Filter

För att tillämpa Kalman filtrets ekvationer som förklarades i teoriavsnitt 2.3 behövdes matriser och vektorer som beskriver systemet att bestämmas. Processen att bestämma dessa redogörs för i detta avsnitt och grundade sig på tidigare examensarbete.

Tillståndsvektorn x , som från och med nu noteras som s för att inte förväxlas med x -koordinaten, kunde beskrivas som följande för ett diskret tidssteg k [17]:

$$\hat{s}_k = \begin{bmatrix} x_k \\ \dot{x}_k \\ y_k \\ \dot{y}_k \end{bmatrix},$$

där x , y var koordinater i x - och y -led och \dot{x} , \dot{y} var hastighet i respektive riktning. Med finit differens approximerades hastigheten som derivatan av positionen i diskret tid $t = k\Delta t$:

$$\dot{x}_k \approx \frac{x_{k+1} - x_k}{\Delta t}.$$

Nästa tillstånd beskrevs som:

$$\hat{s}_{k+1} = \begin{bmatrix} x_k + \Delta t \cdot \dot{x}_k \\ \dot{x}_k \\ y_k + \Delta t \cdot \dot{y}_k \\ \dot{y}_k \end{bmatrix} = F \hat{s}_k.$$

Tillståndsövergångsmatrisen blev därmed:

$$F = \begin{bmatrix} 1 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Tiden mellan UWB-mätningar mättes till $\Delta t = 0,14$ s.

UWB-modulerna som användes kunde mäta position i tre dimensioner, men för vårt syfte behandlades endast koordinater i planet och därmed blev mätningsvektorn

$z_k = \begin{bmatrix} x_k \\ y_k \end{bmatrix}$. Förhållandet mellan mätningsvektorn och tillståndsvektorn gavs av

$z_k = H s_k + v$ där v var brusvektor. H matrisen extraherade x- och y-koordinaten från tillståndsvektorn. $H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$.

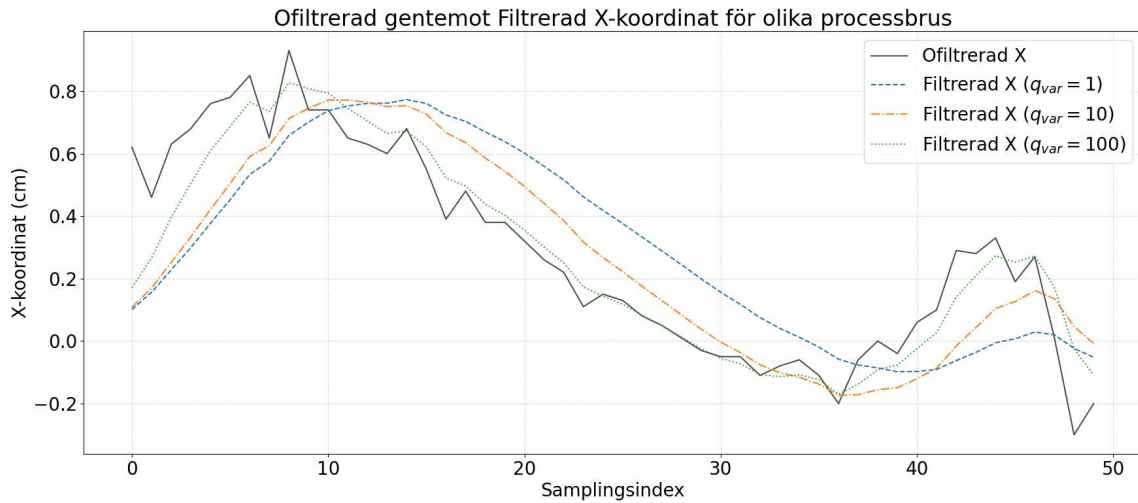
R matrisen speglade de mätosäkerheter i UWB-antennerna vilket reflekterades i de uppmätta koordinaterna. I vårt fall var det en 2x2 kovariansmatris $R = \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix}$.

En simplificering gjordes i antagandet att mätbruset i x och y var oberoende av varandra. Bestämmandet av standardavvikelsen σ för varje koordinat redogörs i avsnitt 5.1. Resultatet blev $\sigma_x = 23$ och $\sigma_y = 18$ i cm för avstånd inom 5 m.

Därefter behandlades processbrusets kovarians Q . På grund av svårigheter att modellera detta förenklades det till att vara ett brus som försköt ett predikterat tillstånd till att motsvara det verkliga tillståndet [17]. Matrisen förenklades till

$$Q = \begin{bmatrix} q_{var} & 0 & 0 & 0 \\ 0 & q_{var} & 0 & 0 \\ 0 & 0 & q_{var} & 0 \\ 0 & 0 & 0 & q_{var} \end{bmatrix}.$$

Ett högre Q gjorde att förtroendet i modellen blir lägre eftersom prediktionen behöver större förskjutning. I praktiken blev filtret mer responsivt men brusigt för högt Q . Å andra sidan jämnade filtret ut prediktionerna mer för lägre Q men reaktioner till plötsliga ändringar blev långsammare. Detta samband visas i Figur 4.3 där Kalmanfiltret av olika magnitud på q_{var} applicerades på positionen av en gående människa inom 2 meter av roboten. Kalmanfiltret innehöll de matriser och vektorer som tidigare bestämts i detta avsnitt. $q_{var} = 10$ valdes som en medelväg mellan snabb responsivitet och förmåga att jämma ut brus.



Figur 4.3: Ofiltrerade x-kordinater jämförda med Kalmanfiltrerade x-kordinater för olika värden på processbruset Q_{var} .

Slutligen bestämdes den initiala kovariansen för skattningen P_0 och initiala tillståndsvektorn s_0 . Dessa innehöll startvärden som filtret sedan rekursivt uppdaterade och var relevant för hur snabbt filtret konvergerade mot det initiala tillståndet. Eftersom målet ofta placerades stillastående en meter rakt framför roboten vid start så sattes dessa till att vara:

$$P_0 = \begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix}, s_0 = \begin{bmatrix} 0 \\ 0 \\ 100 \\ 0 \end{bmatrix}.$$

4.4 Implementering av kod

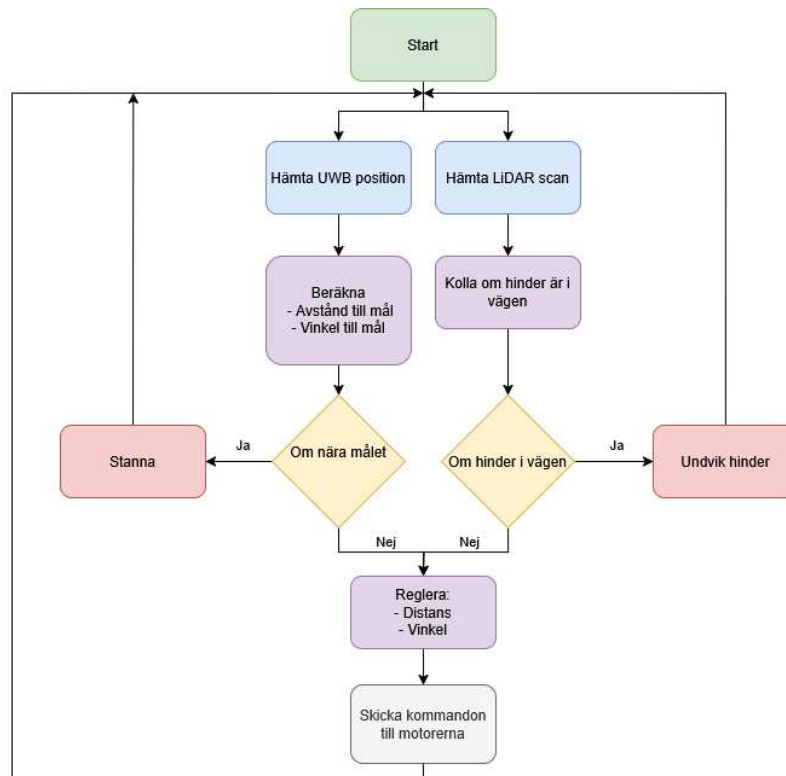
Vid uppstart initierade systemet UWB-modulen för att bestämma robotens position, LiDAR-sensorn för att upptäcka hinder, samt UART-kommunikationen för att skicka styrkommandon till en Raspberry Pi Pico. Den sistnämnda användes för att styra motorerna.

I varje iteration av huvudloopen hämtades robotens aktuella position i förhållande till målet via UWB. Utifrån detta beräknades både avståndet till målet och riktningen dit.

Om roboten befann sig tillräckligt nära målet, stannade den. Om ett hinder upptäcktes inom en definierad säkerhetszon framför roboten, aktiverades ett hinderundvikande beteende. Beroende på hindrets position och målets riktning, kunde roboten exempelvis backa eller svänga för att undvika kollision.

Om inget hinder fanns och roboten inte var framme, användes PID-reglering för att styra rörelsen. En PID-regulator kontrollerade hastigheten baserat på avståndet till målet, medan en annan styrde riktningen utifrån vinkelavvikelsen.

De beräknade styrkommandona skickades via UART till Raspberry Pi Pico, som styrde motorerna. Loopen upprepades kontinuerligt, vilket gjorde att roboten hela tiden anpassade sin rörelse i realtid baserat på målets position och hinder i omgivningen. Figur 4.4 visar ett flödesschema över denna huvudloop i koden.



Figur 4.4: Flödesschema för huvudloopen i kontrollsystemet, som visar hur roboten navigerar mot målet och undviker hinder.

Systemet implementerades med en strategi för att undvika hinder som kombinerade direkt kollisiondetektion med ett avståndsbaserat styrfält. En LiDAR-sensor användes för att kontinuerligt övervaka området framför roboten och identifiera potentiella hinder. När ett föremål befann sig för nära roboten, har denna stannat och utfört en undvikande manöver genom att rotera eller backa i en kontrollerad riktning, beroende på hinderpositionen i förhållande till rörelseriktningen.

Utöver direkta hinder tog systemet även hänsyn till närliggande objekt genom att införa ett potentiellt styrfält. Styrningen justerades redan på avstånd från hinder genom påverkan från det potentiella styrfältet. Fältet beräknades baserat på hinderpositioner i ett begränsat synfält framför roboten. En hinderposition definierades till att vara en LiDAR-mätning som upptäckte en punkt inom en meter till sensorn. Hinder nära roboten ackumulerade fler punkter från LiDAR-sensorn. Algoritmen räknade antalet hinderpositioner i synfältet på vänster och höger sida om roboten

och gav en starkare styrsignal för att svänga i motsatt riktning för varje sådan position.

4.5 Styrsystem

Styrsystemet i detta sammanhang avser den programvara som datorn kör och dess utformning. Följande två avsnitt beskriver hur olika relevanta parametrar fastställdes och hur motorsignalerna är uppbyggda.

4.5.1 Parametrar

För att roboten skulle uppnå önskad funktion implementerades diverse parametrar i programmet. Bland de viktigaste av dessa var regulatorparametrar, styrsignalbegränsningar, önskat följningsavstånd och säkerhetsdistans till hinder.

De två signalerna som regleras, det vill säga vinkel och distans till målet (θ och d), behandlas i separata reglersystem och tilldelas därför individuella reglerparametrar. Valet av parametervärden gjordes genom att iterativt testa, med hjälp av grundförståelse för påverkan av de olika komponenterna, olika kombinationer. Det framkom att den integrerande faktorn för de båda regulatorerna inte bidrog till någon förbättring av körförmågan. Vidare för att minimera oscillationer begränsades vinkelregulatorn till att endast reglera fel över absolutvärde 10 grader.

4.5.2 Motorsignaler

Regulatorernas styrsignaler skickades till mikrokontrollern som därefter sände motorsignalen till motordrivaren enligt Ekvation (4.1) - (4.2).

$$p_{vänster} = v_{fram} + v_{sväng} \quad (4.1)$$

$$p_{höger} = v_{fram} - v_{sväng} \quad (4.2)$$

Styrsignalen v_{fram} från distansregulatorn kopplades till hastigheten framåt och antog ett värde mellan 0 och 100. Distansregulatorn var inställd på endast positiva värden då backning när ledaren närmade sig korgen skulle försvåra processen att placera varor i korgen. Styrsignalen $v_{sväng}$ från vinkelregulatorn avgjorde hur mycket roboten svängde och antog värden mellan -100 och 100. Dessa enhetslösa värden kombinerades enligt Ekvation (4.1)-(4.2) för att bilda en procentsats p för respektive hjul. Från dessa ekvationer tydliggörs även hur svängningen uppstod som resultat av att $v_{sväng}$ bildade en differens mellan rotationshastigheten av motorer. Positiv $v_{sväng}$ gjorde att vänsterhjul snurrade snabbare än högerhjul alltså svängde roboten åt höger. Vidare avgjorde tecknet framför p polariteten för motorn. Absolutvärdet av p begränsades till att vara maximalt 100 då procentsatsen motsvarade arbetscykeln för PWM-signalen till motordrivaren. På så sätt bestämde p effektmatningen för respektive motor.

5

Tester och Resultat

För att undersöka och analysera hur komponenterna fungerar i praktiken har ett antal tester genomförts. Syftet med dessa tester var att se hur väl de krav och önskemål som ställdes i kravspecifikationen uppfylldes (se Figur A.1 i Appendix A), samt för att testa sensorernas noggrannhet. Nedan presenteras en lista på de tester som gjorts. I följande avsnitt ges en beskrivning av dessa tester tillsammans med testernas resultat:

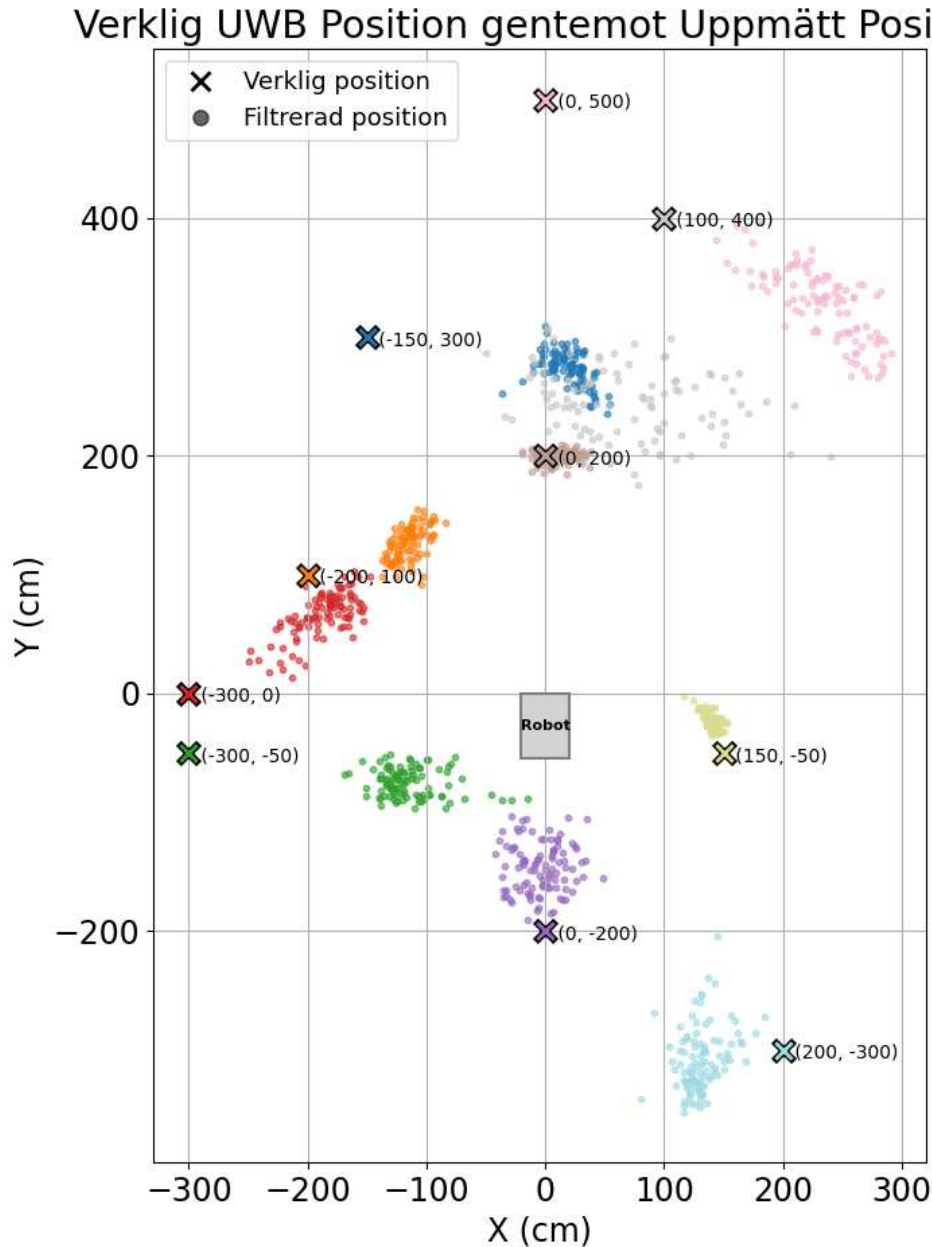
- Precision och noggrannhet för MDEK1001
- Positionsmätning genom UWB med Kalmanfilter
- Positionsmätning genom UWB i cirkulär bana med Kalmanfilter
- Avståndsmätning för LiDAR-enheten
- Mätning av avvikelse vid färd rakt fram
- Mätning av hastighet
- Belastningstest
- Mätning av bromssträcka
- Svänga runt hörn
- Avstånd från ledaren
- Följa ledare bland hinder
- Test i matvarubutik.

5.1 Precision och noggrannhet för MDEK1001

För att undersöka vilken positionsprecision och noggrannhet UWB-modulerna hade, genomfördes ett test där uppmätt position jämfördes med verklig position. Enligt datablad rekommenderas ett avstånd mellan ankarna på 20 till 25 meter för bättre prestanda [32]. Dessutom förväntades taggen befinna sig inom det området som ankarna spänner upp. Dock var ankarna i detta projekt monterade på roboten och kunde därför inte uppnå avstånd inom det rekommenderade intervallet, därav intresset att undersöka hur det påverkade mätningarna.

Taggen, som var monterad i scannern, sattes fast på ett stativ för få den i ungefärlig höjd som om att en person höll i den (70 cm över marken). Stativet ställdes på tio godtyckligt valda positioner inom fem meter. 100 mätvärden samlades in för varje position. Gränsen på fem meter valdes på basis från tidigare observationer där både noggrannhet och precision försämrades avsevärt efter det avståndet.

Figur 5.1 visar ovanifrån vilka punkter som valdes (de färgade kryssen) och jämför det med de uppmätta punkterna med korresponderande färg.



Figur 5.1: Verklig position (färgade kryss) jämfört med uppmätt position (färgade prickar) för 10 olika positioner.

Sedan analyserades normalfördelningen av felet mellan verklig och uppmätt x- och y-koordinat för varje position. Eftersom normalfördelningen skiljde sig avsevärt från position till position gjordes en förenkling. Standardavvikelsen σ i x- och y-riktning för alla positioner beräknades som medelvärdet av varje standardavvikelse. Detsamma gjordes för medelfelet μ . Måtten sammanställs i Tabell 5.1. Ytterligare beräk-

nades medianen av dessa mått för att minimera påverkan från avvikande positioner med mycket högre medelfel och standardavvikelse än resterande positioner. Ett exempel på en sådan avvikare är det rosa krysset i Figur 5.1. Medianvärdena sammanställs i Tabell 5.1. Den genomsnittliga avvikelser användes i Kalmanfiltret från avsnitt 4.3.

Tabell 5.1: Vänster tabell visar medelvärdet av medelfel och standardavvikelse i x- och y-riktning, medan höger visar medianen av dessa.

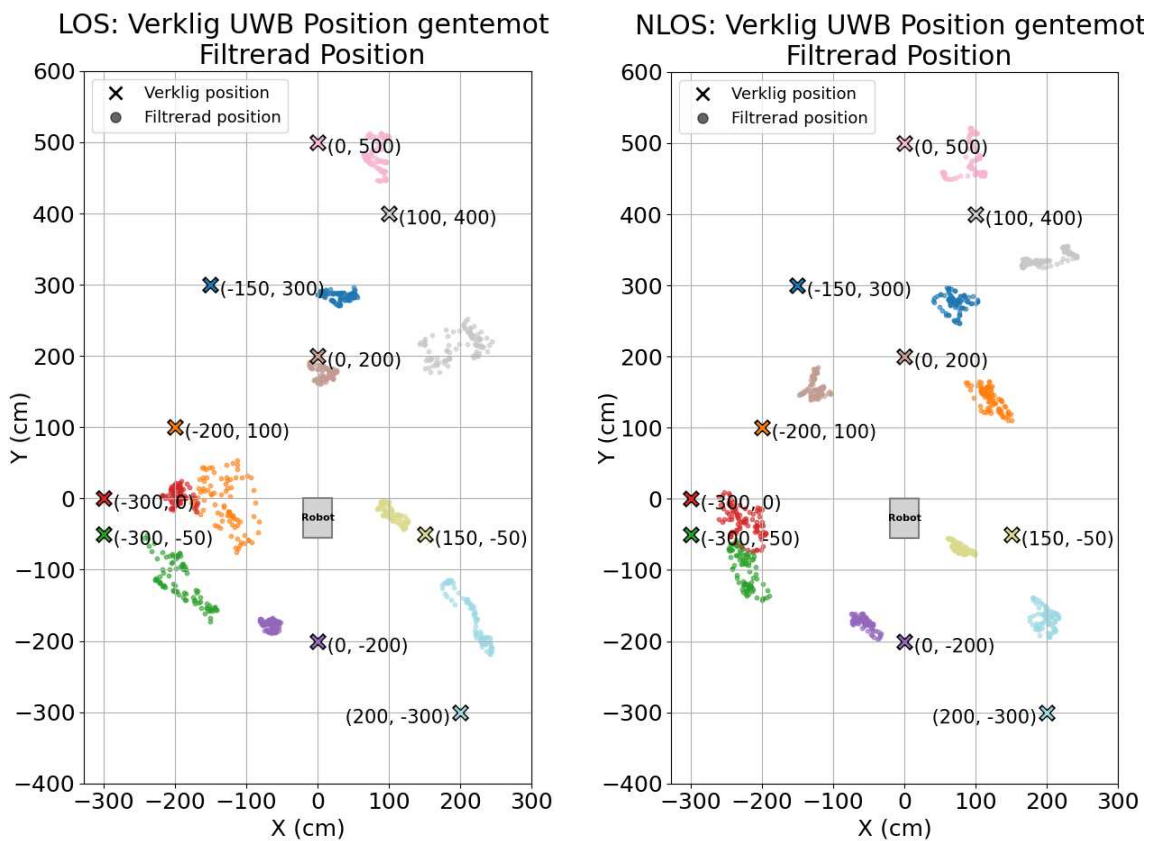
Mått	Uppmätt fel (cm)	Mått	Uppmätt fel (cm)
Genomsnittlig μ_x	67,9	Median μ_x	47,6
Genomsnittlig μ_y	-21,5	Median μ_y	-4,2
Genomsnittlig σ_x	22,5	Median σ_x	17,9
Genomsnittlig σ_y	17,7	Median σ_y	16,6

5.2 Positionsmätning genom UWB med Kalmanfilter

Testet som mätte precision och noggrannhet för MDEK1001 visade tydliga fel på korrekt uppmätning av avstånd, samt tydliga tecken på brus. Ett Kalmanfilter skapades därefter, och mer exakt hur det togs fram förklaras i avsnitt 4.3. Utförandet av detta test gick ut på att placera den färdiga roboten på en öppen yta, och syftade till att verifiera krav 1.8 i kravspecifikationen, som anger ett maximalt tillåtet fel i avståndsuppmätning på 0,3 meter och önskemål 1.12 - kan identifiera ledaren bakom solida objekt. Testet gjordes i två etapper, där den första var att mäta med line-of-sight (LOS) och därefter med non-line-of-sight (NLOS). Stativet placerades ut på olika positioner runt om roboten (se Figur 5.2), där dessa positioner var samma positioner som användes för att skapa Kalmanfiltret (se avsnitt 5.1). Uppmätningarna genomfördes och jämfördes med de verkliga positionerna, först utan objekt mellan tagg och robot och därefter med svarta trälådor mellan för att uppnå NLOS. Figur 5.3 visar resultatet för testet och Tabell 5.2 - 5.3 sammanställer medelfel och standardavvikelse för uppskattningarna. Något som inte framgår av figurerna eller tabellerna är att för NLOS-fallet tog det i vissa fall mer än dubbelt så lång tid att samla de 100 punkterna.



Figur 5.2: Exempel på uppställning för avståndsmätning med hjälp av UWB med LOS (vänster) och NLOS (höger).



(a) Utan blockerad vy mellan robot och mål (b) Med blockerad vy mellan robot och mål

Figur 5.3: Verklig position (färgade kryss) jämfört med filtrerad, uppmätt position (färgade prickar) för 10 olika positioner runtom roboten.

Tabell 5.2: Feluppskattning för LOS. Vänster tabell visar medelvärdet av medelfel och standardavvikelse i x- och y-riktning, medan höger visar medianen av dessa.

Mått	Uppskattat fel (cm)	Mått	Uppskattat fel (cm)
Genomsnittlig μ_x	55,4	Median μ_x	76,1
Genomsnittlig μ_y	-23,2	Median μ_y	-17,1
Genomsnittlig σ_x	16,8	Median σ_x	13,8
Genomsnittlig σ_y	17,7	Median σ_y	14,5

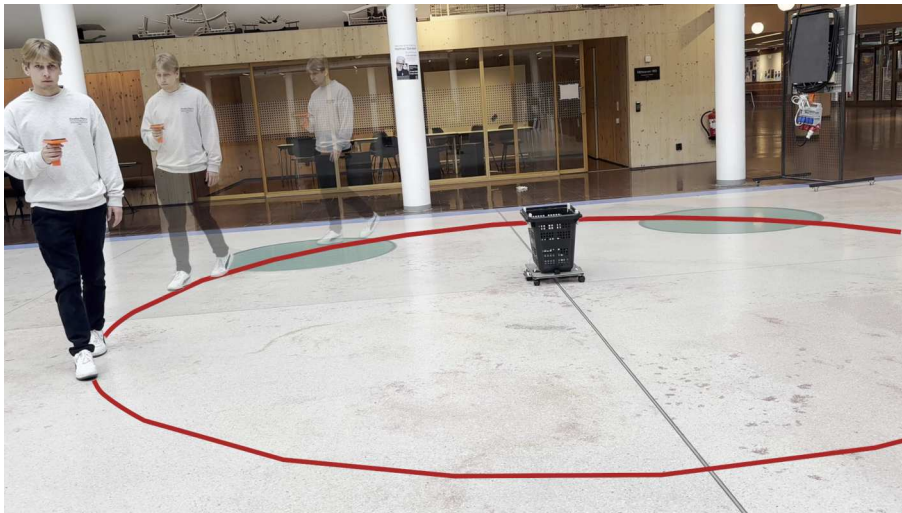
Tabell 5.3: Feluppskattning för NLOS. Vänster tabell visar medelvärdet av medelfel och standardavvikelse i x- och y-riktning, medan höger visar medianen av dessa.

Mått	Uppmätt fel (cm)	Mått	Uppmätt fel (cm)
Genomsnittlig μ_x	62,7	Median μ_x	74,7
Genomsnittlig μ_y	-5,5	Median μ_y	-24,8
Genomsnittlig σ_x	14,3	Median σ_x	12,9
Genomsnittlig σ_y	14,5	Median σ_y	13,0

Resultaten från både LOS- och NLOS-testerna visar att det uppmätta felet i positionsuppskattningen överskred den tillåtna gränsen, då det uppmätta medelfelet i samtliga fall översteg 30 cm.

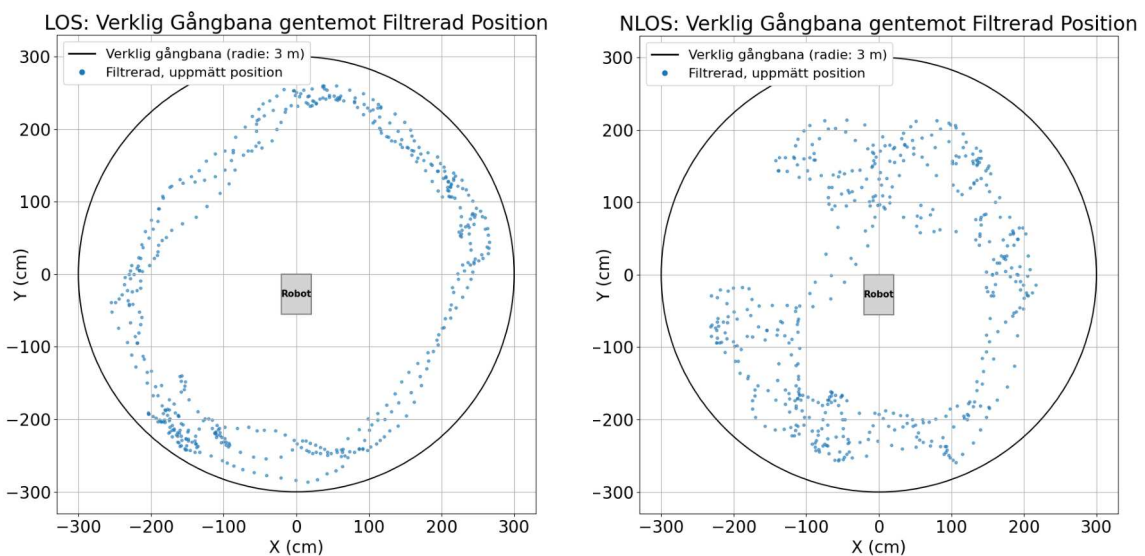
5.3 Positionsmätning genom UWB i cirkulär bana

En cirkel med radien tre meter tejpades upp på ett golv, och roboten placerades så att referenspunkten (mittpunkten precis framför LiDAR-sensorn) var precis ovanför cirkelns mittpunkt. En person höll i UWB-taggen och gick runt cirkelns kant (se Figur 5.4), fram tills att 500 mätpunkter samlades in. Syftet med testet var att se hur noggrann avståndsmätning blev när taggen rörde sig. Precis som testet i Avsnitt 5.2 gjordes även detta test i två delar, först med LOS och sedan med NLOS, där svarta trälådor placerades ut och inkapslade roboten. De uppmätta positionerna jämfördes med den faktiska cirkulära gångbanan.



Figur 5.4: Uppställning för test av UWB-mätning i cirkulär bana.

Figur 5.5 visar resultatet för testet. För båda testen syns det tydligt att UWB-modulerna underskattar det verkliga avståndet. Figur 5.5a visar dock en mer definierad cirkel, där mätpunkterna inte är lika utspridda som de är i Figur 5.5b. Med en blockerad vy kan man också se att en del punkter avviker från de andra punkterna (uppe till vänster i figuren).



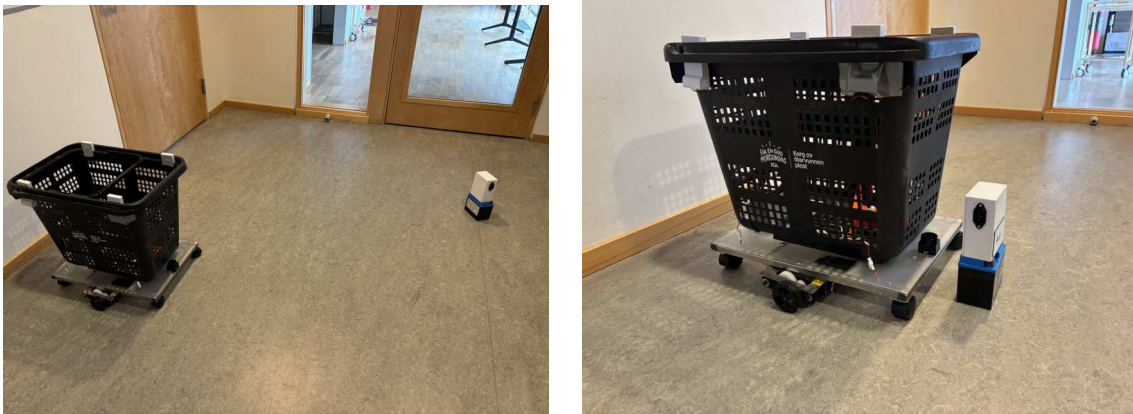
(a) Utan blockerad vy mellan robot och mål (b) Med blockerad vy mellan robot och mål

Figur 5.5: Verklig bana (heldragen svart linje) jämfört med filtrerad, uppmätt position (blå prickar) för gående person med UWB-tagg.

5.4 Avståndsmätning för LiDAR-enheten

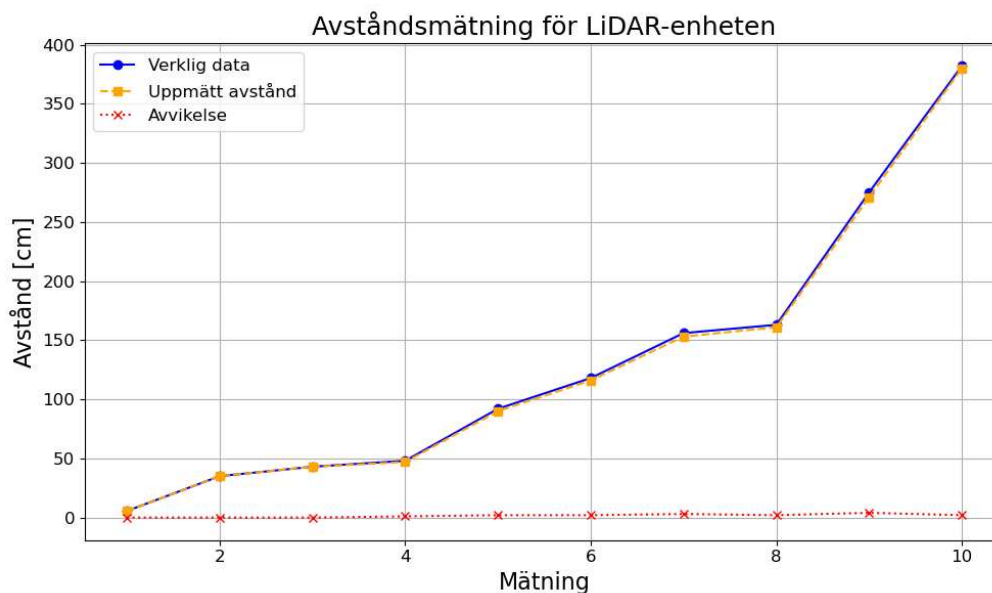
För att mäta LiDAR-enhetens precision i att mäta avstånd till objekt inom sitt synfält, gjordes ett test likt det som genomfördes för UWB-modulerna. Roboten

placerades på en öppen yta. Två batterier användes som godtyckliga objekt och placerades på varandra. Det gjordes 10 mätningar där objektet placerades på olika avstånd och vinklar från roboten, men inom LiDAR-sensorns synfält. För varje mätning mättes avståndet mellan sensorns centrum och objektet och jämfördes med det avstånd sensorn mätte upp. Jämfört med testet med UWB modulerna gjordes detta test med kortare avstånd av den anledning att LiDAR-sensorns syfte är att upptäcka närliggande objekt.



Figur 5.6: Exempeluppställningar för test av LiDAR-enheten.

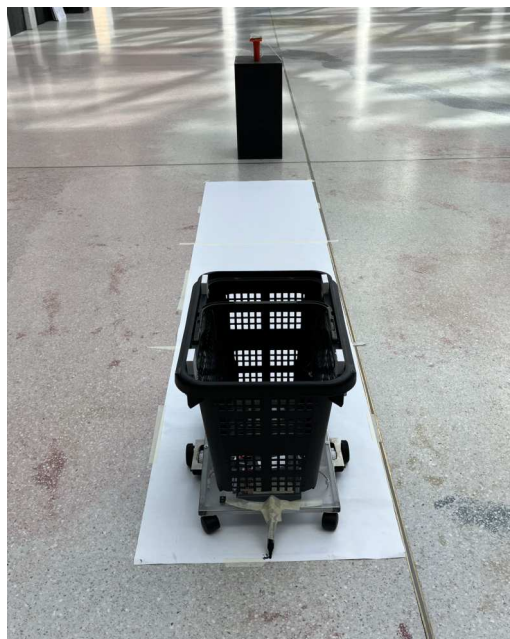
Avstånden som uppmättes av LiDAR-sensorn hade en generell avvikelse på 1-2%. Den maximala avvikelsen var 2,1% vid 48 cm avstånd. Det användes 10 mätvärden mellan 5,5 - 380 cm. Vid alla de 8 tillfällen där det uppmättes en avvikelse var den uppmätta längden kortare än den faktiska längden.



Figur 5.7: Grafisk presentation av resultatet från LiDAR-testet där även avvikelsen mellan verklighet och uppmätt distans är presenterad.

5.5 Mätning av avvikelse vid färd rakt fram

För att utvärdera hur bra roboten var på att köra rakt fram mättes avvikelser från en mittlinje (se Figur 5.8). Stora vita papper tejpades fast på ett golv och en penna sattes fast bak på roboten. Sträckan roboten körde var på 282 cm, vilket motsvarade den totala längden på pappersarken. Längden ansågs vara tillräcklig för att uppfatta avvikelser, samtidigt som det motsvarar ett rimligt avstånd mellan tagg och robot. Taggen ställdes på en pedestal en bit framför slutet på pappersarken, ett avstånd motsvarande längden på roboten. Roboten startade vid början av pappersarken och markerade med hjälp av pennan sin väg fram till taggen. Det gjordes 10 mätningar för detta test, och för varje mätning mättes avståndet från papprets mittlinje till den punkt på robotens spår som låg längst bort från mittlinjen.



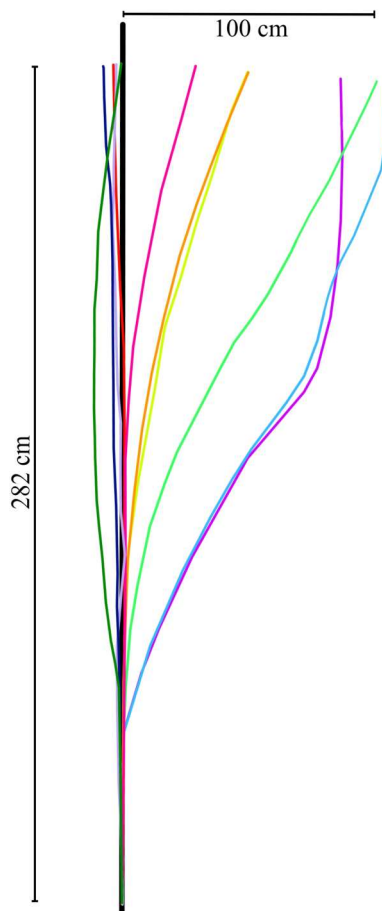
Figur 5.8: Uppställning inför test vid mätning av avvikelse vid färd rakt fram.

Avvikelser i färdbanan i förhållande till den kortaste vägen uppvisades. Mätresultaten presenteras i Tabell 5.4.

Tabell 5.4: Mätresultat från mätning av avvikelser vid färd rakt fram, där största avvikelsen [cm] över en färdad sträcka på 282 centimeter noterats. Positivt värde är förskjutning åt höger.

Nummer	Största avvikelse (cm)
1	-4
2	32
3	107
4	-2
5	23
6	0
7	101
8	31
9	-6
10	81

Median-avvikelsen var 27 cm, vilket motsvarade 0,1 cm förskjutning åt höger per cm som färdades rakt fram. Den största uppmätta avvikelse var 107 cm, vilket motsvarar en avvikelse på 0,4 cm per cm i färdriktningen. Färdbanorna presenteras i Figur 5.9, där det framgår att roboten antingen kör nära en rak färd eller avviker till höger.



Figur 5.9: Resultat från testning av avvikelse i rak färdriktning, digitalt återskapat, där varje enskild färg markerar vägen roboten tog under en iteration av testet och centrumlinjen är markerad i svart.

5.6 Mätning av hastighet

Eftersom roboten inte hade en komponent monterad på sig som kunde mäta hastighet gjordes ett praktiskt test. Testet gjordes på en sträcka på fyra meter. En timer startades när roboten passerade startlinjen och stoppades när den korsade mållinjen. Testet genomfördes först tre gånger med roboten inställd på att köra 100% av sin maximala hastighet. Därefter genomfördes testet tre gånger till, fast denna gång med hastigheten på 80% av sin maxhastighet. Denna inställning används också i den implementerade koden eftersom roboten annars kör för snabbt, vilket gör att den inte hinner svänga eller manövrera på ett tillförlitligt sätt.

Den högsta uppmätta hastigheten var 2,4 km/h. För 80 % av maximal hastighet uppmättes en verklig hastighet på 1,9 km/h.

Enligt krav 1.10 i kravspecifikationen ska robotens körhastighet ligga mellan 4 och 6 km/h. Eftersom samtliga uppmätta värden låg under detta intervall kan det konstateras att kravet inte uppfylldes.

5.7 Belastningstest

Enligt krav 1.2 i kravspecifikationen skulle roboten klara av last på upp till 15 kg. För att testa detta gjordes ett belastningstest. Först vägdes den färdiga roboten utan någon last för att bestämma dess egenvikt. Efteråt lastades korgen med stegvis ökande vikter: först 5 kg, sedan 7,5 kg och till sist 10 kg. Efter varje viktillägg observerades hur robotens körförmåga påverkades.

Robotens egenvikt var 8 kg. Vid last på 5 kg märktes inte någon märkbar förändring på körförmåga. Vid 7,5 kg last noterades det att roboten kunde köra framåt emellanåt, att den började slira och fick problem med att backa. Vid 10 kg last kunde roboten vid vissa tillfällen flytta på sig framåt, men förlorade helt förmågan att backa eftersom hjulen inte fick något grepp. Roboten klarade dock strukturellt alla viktnivåer under testet. Sammanfattningsvis uppfyllde roboten inte kravet eftersom den inte klarade att köra med en last på 15 kg.

5.8 Mätning av bromssträcka

För att kontrollera att roboten uppfyller krav 1.7 i kravspecifikationen, som fastställer att bromssträckan ska vara kortare än 0,5 meter, genomfördes en mätning av bromssträckan. Roboten startade utan last med maxhastighet, en bit bakom en bestämd linje och när den passerade linjen stängdes roboten av manuellt. Testet gjordes 3 gånger och avståndet mellan linjen och roboten mättes upp efter varje mätning.

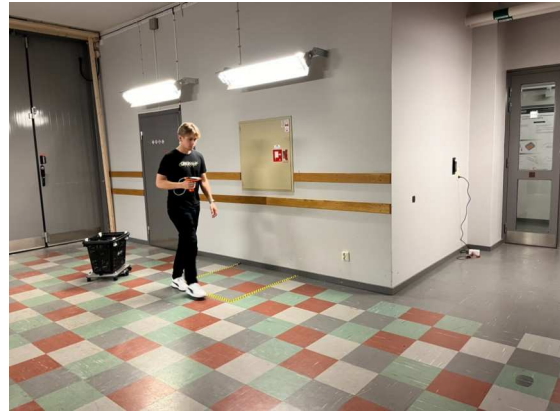
Den bromssträcka som uppmättes vid maximal hastighet var i genomsnitt 44 cm. Den högsta uppmätta bromssträckan var 47 cm. Notera att detta resultat gäller för den maximala uppmätta hastigheten, 2,4 km/h. Då samtliga mätvärden låg under gränsvärdet på 0,5 meter kan det konstateras att kravet uppfylldes.

5.9 Svänga runt hörn

För att utvärdera robotens förmåga att navigera i mer verklighetstroga miljöer där vägen inte alltid är rak, gjordes ett test där roboten fick följa efter en person som bar UWB-taggen runt ett hörn. Två olika varianter av testet gjordes (se Figur 5.10), och båda varianterna genomfördes tre gånger vardera. Det första scenariot innebar att tagg-bäraren redan stod bakom ett hörn när roboten sattes igång, och det andra scenariot av testet gick ut på att tagg-bäraren stod framför roboten när den sattes igång, och efter starten gick runt hörnet.



(a) Start bakom hörn.



(b) Start framför korg.

Figur 5.10: Testuppställning vid svängning av hörn.

Testerna visar varierande resultat beroende på startsituation. Resultatet från testerna framgår i Tabell 5.5 nedan.

Tabell 5.5: Resultat från tester av robotens förmåga att svänga runt hörn.

Mätning	Test	Kommentar	Kollisioner	Stopp
1	Bakom hörn från början	Vände, kolliderade med vägg, korrigerade kurs och fortsatte stabilt	1	0
2		Tog ~20 sekunder att hitta taggen, körde rakt fram, undvek kollision och svängde korrekt	0	0
3		Snabbare start, körde rakt och svängde utan problem	0	0
1	Gå till hörn efter start	En kollision, följde målet okej med liten fördröjning	1	1
2		Följde målet väl, men med stor svängradie och fördröjning	0	0
3		Samma beteende som i föregående mätning	0	0

Figur 5.11 visar den körväg roboten tog för mätning med tagg-bärare bakom hörn vid start. Bilden visar att roboten klarade av att svänga, dock med en väldigt stor svängradie.



Figur 5.11: Körväg av robot för mätning 2 vid test med tagg-bärare bakom hörn vid start.

5.10 Avstånd från ledaren

Krav 1.1 i kravspecifikationen angav att roboten ska stanna när avståndet till den som bär taggen är mindre än 70 cm. Testet utfördes genom att ledaren, det vill säga personen som bar taggen, först gick en bit bortom roboten och stannade. Roboten skulle därefter automatiskt följa efter och stanna när den bedömde att det angivna avståndet till ledaren hade uppnåtts. Det kortaste avståndet mellan ledaren och roboten mättes vid varje testtillfälle. Testet upprepades fem gånger.

Medelavståndet mellan ledaren och roboten var 50 cm, med värden mellan 31 och 65 cm. Därmed uppfylldes det uppställda kravet.

5.11 Följa ledare bland hinder

Detta test genomfördes för att undersöka robotens förmåga att navigera i både statiska miljöer och i enlighet med önskemål 1.12 ("att klara av dynamiska miljöer").

För att efterlikna en mer realistisk miljö placerades föremål ut på en öppen yta för att representera stillastående hinder, samtidigt som en person rörde sig runt i området för att simulera rörliga, dynamiska hinder. De stillastående hinderna var i samma storleksordning som roboten. Testledaren promenerade runt i området, och det observerades om roboten kolliderade med något hinder eller stötte på andra problem under testets gång.

När roboten skulle undvika fasta hinder inträffade kollisioner sällan. Dock, på grund av många hinder och smala passager, hände det ibland att roboten stannade för att undvika en krock. I de flesta fall lyckades roboten ta sig fram till testledaren.

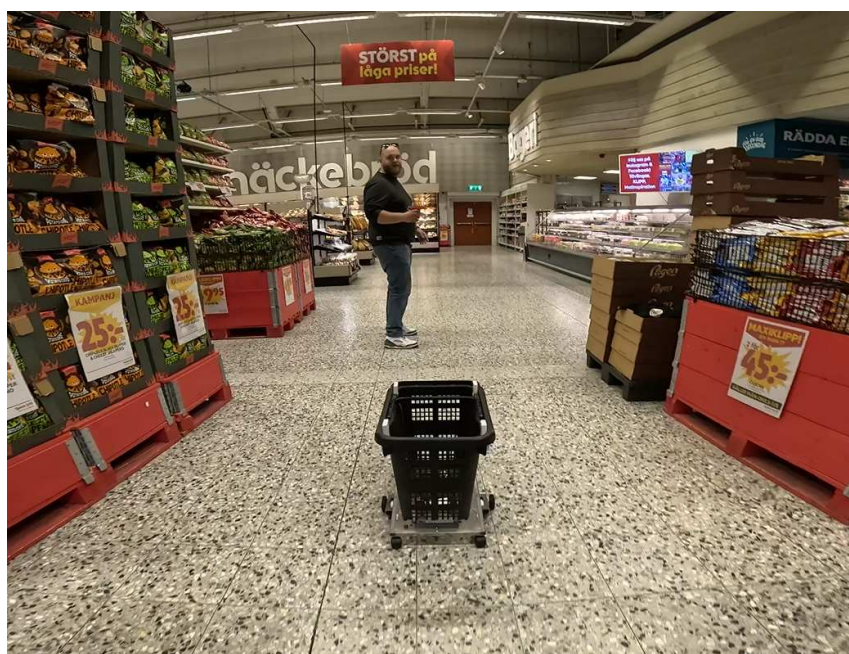
I en dynamisk miljö undvek roboten i regel att kollidera med den rörliga personen, så länge foten inte pekade direkt mot roboten. Detta eftersom LiDAR-sensorn inte identifierar så låga objekt. När foten däremot pekade mot roboten ökade risken för kollision, även om detta inte alltid ledde till en faktisk krock.

5.12 Test i matvarubutik

Eftersom roboten är ämnad att användas av kunder i matvarubutiker, gjordes ett test på Maxi ICA Stormarknad Göteborg. Butiken har både öppna ytor och smalare gångar mellan hyllor. En stor del av varorna i butiken står på antingen Europapallar eller hyllor vars ben befinner sig en bit in från hyllans kant. Vidare gjordes testet en tid då väldigt få kunder besökte butiken. Det gjordes inga mätningar i testet. Istället observerades den generella körförmågan.

Vid testkörningen noterades att roboten vid ett flertalet tillfällen krockade in i hyllor, frysdiskar och pallar. I vissa fall stannade roboten till, trots att det var fri sikt. Roboten hade dessutom svårt att färdas rakt vid körning rakt fram, utan körde en slalomlikt bana för att ta sig fram. Detta visar tydliga tecken på kraftig oscilla-

tion. Däremot klarade roboten av att svänga runt hörn de flesta gånger, ibland utan problem och ibland med att roboten först höll på att köra in i ett hinder, men lyckades stanna, backa och sedan fortsätta svängen. Vid några av tillfällena en kollision inträffade, gick axeln mellan ett drivhjul och dess växellåda av och krävde byta till en ny axel. Eftersom roboten körde långsammare än gånghastighet ökade avståndet mellan roboten och användaren successivt under gång. Detta medförde att den UWB-baserade positioneringen blev mycket mindre tillförlitlig för längre avstånd än fem meter, och roboten navigerade i fel riktning som led in i hinder. Roboten hamnade i en cykel där den försökte köra i en vägg och justerade sig själv för att inte kollidera. Därmed uppfylldes inte önskemålet 1.9 att kunna följa ett mål inom 10 m. Sammanfattningsvis hade roboten brister i sin förmåga att både navigera till taggens position och att upptäcka hinder. I Figur 5.12 visas en fotografi från testet.



Figur 5.12: Ett fotografi tagen från testet i matvarubutik.

5.13 Resultat för resterande krav i kravspecifikationen

Eftersom alla tester utfördes i rumstemperatur och på plan yta ansågs kraven gällande temperatur i användningsmiljö och att klara av nivåskillnad vara uppfyllda. Dessutom ansågs önskemålet om att robotens komponenter skulle vara dolda, vara delvis uppfyllt. En majoritet av komponenterna blev dolda under roboten medan UWB-modulerna, LiDAR-enheten och batteriet blev synliga på ovasidan.

Krav 1.6 (svängradie), testades inte då robotens manövreringsförmåga redan verifierats genom dess konstruktion med differentialstyrning, vilken möjliggör rotation kring den egna axeln. Därför ansågs det inte nödvändigt att genomföra ett separat testmoment för detta krav.

Utifrån samtliga tester har det konstaterats vilka krav från kravspecifikationen som uppfylldes eller ej. Dessa är sammanställda i Figur 5.13 nedan.

5. Tester och Resultat

Chalmers	Dokumenttyp	Kravspecifikation		
	Projekt	Autonom kundkorg EENX16		
Utförare: Grupp 11 EENX16		Skapad: 2025-01-30		
		Modifierad: 2025-05-12		
Kriterier	Målvärde	K/Ö	Verifieringsmetod	
Funktion(er)				
Följa ledaren		K		
Navigera en statisk miljö		K		
Klara av att bromsa		K		
1 Prestanda				
1,1 Avstånd från ledaren då fordonet stannar	<0,7 m	K	Praktiskt test	
<i>Gäller endast vid körning med tom korg. (För att inte kollidera med ledaren)</i>				
1,2 Klara statiska laster	< 15 kg	K	Praktiskt test	
<i>Krafter som uppstår vid en storhandling</i>				
1,3 Klara statiska laster	< 25,6 kg	Ö	Praktiskt test	
<i>Säkerhetsfaktor på 1,6</i>				
1,4 Undvika kollision	Kolliderar inte med några hinder	K	Praktiskt test	
1,5 Maximal amplitud oscillering vid efterföljning	0,5 m	Ö	Praktiskt test	
<i>Maximal svängning vid efterföljning</i>				
1,6 Svängradie	< 0,5 m	K	Praktiskt test	
1,7 Bromssträcka	< 0,5 m	K	Praktiskt test	
<i>Gäller endast vid körning med tom korg</i>				
1,8 Maximalt fel i avståndsuppmätning	0,3 m	Ö	Praktiskt test	
<i>Maximalt fel i uppmätt avstånd till ledare som befinner sig 2 meter bort</i>				
1,9 Maximal följningsavstånd	10 m	Ö	Praktiskt test	
<i>Följer ett mål inom radien</i>				
1,10 Körhastighet	4-6 km/h	K	Praktiskt test	
<i>Hastighet vid arbetspunkten</i>				
1,11 Kan navigera dynamiska miljöer		Ö	Praktiskt test	
1,12 Kan identifiera ledaren bakom solida objekt		Ö	Praktiskt test	
2 Miljö (omgivande)				
2,1 Temperatur i användningsmiljö	15-25 °C	K	Praktiskt test	
2,2 Klara av nivåskillnad	± 2cm/m	Ö	Praktiskt test	
3 Konstruktion				
3,1 Dolda komponenter		Ö	Visuell inspektion	
<i>Samtliga komponenter dolda inmanför "skal"</i>				
4 Storlek				
4,1 Djup	≤ 550mm	K	Mätning	
4,2 Djup	≤ 530mm	Ö	Mätning	
4,3 Bredd	≤ 430mm	K	Mätning	
4,4 Bredd	≤ 410mm	Ö	Mätning	
5 Massa				
5,1 Tjänstevikt	≤ 4,4 kg	Ö	Mätning	

Figur 5.13: Sammanställning av vilka krav i kravspecifikationen som uppfyllts där grönt innebär uppfyllt, rött innebär ej uppfyllt, gult innebär delvis uppfyllt och blå innebär ej testat.

6

Diskussion

I detta kapitel analyseras resultat för att jämföras med kravspecifikationen. Det diskuteras förbättringsområden och etiska aspekter för framtida utveckling.

6.1 Testresultat

Resultaten diskuteras i sin helhet för tre olika områden: LiDAR, Ultra Wideband och körförmåga.

6.1.1 LiDAR

Utifrån de tester som utfördes i syfte att undersöka sensorerna är det tydligt att LiDAR-enheten utför väldigt precisa mätningar. Anledningen till att samtliga testvärden var kortare än de LiDAR-enheten mätte upp, kan vara kopplat till brister i metoden för mätning, det var svårt att urskilja var exakt som enheten invändigt gjorde sina mätningar. Vid testning i den tänkta miljön (en butik) visade systemet dock tecken på störningar i LiDAR-enhetens mätningar. Ibland upptogs mätningar som var mycket kortare än det verkliga avståndet. Roboten reagerade då som att det var ett hinder i vägen, även om det inte var det. Detta kan tyda på brister i konfidensfiltret eller att dammpartiklar uppmättes.

6.1.2 Ultra Wideband

Den andra sensortekniken, UWB, visade mindre önskvärd mätdata. Anledningen till detta kunde primärt vara att just denna tekniken har i detta projekt blivit tillämpad utanför den menade användningsområdet. Att UWB-ankarna satt så tätt som de gjorde kan varit den största anledningen till att mätningarna inte överensstämde med verkligheten. Felet var stort i genomsnitt mellan uppmätt värde till det faktiska värdet och i spridningen mellan uppmätta värden för samma faktiska värde. Utifrån testet med cirkulär rörelse kring prototypen framgick det att den uppmätta distansen tenderade att vara kortare än den verkliga, eftersom alla uppmätta punkter befann sig inom cirkeln i Figur 5.5. De andra precisionstesterna för UWB visade dock att vid fåtal tillfällen kan den uppmätta distansen vara längre än den faktiska. Vidare visar Tabell 5.1 - 5.3 att de uppmätta koordinaterna tenderade att vara mer än en halvmeter förskjutna positivt i x-led, medan y-koordinaternas absolutvärde var i regel lägre än verkligheten. Detta visade sig inte vara ett alltför stort problem

i praktiken, åtminstone för LOS-förhållanden, eftersom riktningen dit roboten ska köra är grovt densamma. Men de höga felen kunde ha varit en orsak till oscillationerna som uppstod under testet i matvarubutiken.

Något som framgår från Tabell 5.2 - 5.3 är att NLOS-förhållanden inte ger någon tydlig försämring av de uppskattade positionerna. Tvärtom så blir standardavvikelsen några centimeter mindre i medelvärde och median. Detta strider mot förväntan eftersom hinder borde störa radiosignalernas färd mellan ankare och tagg, vilket ger ökad brus för trianguleringen. Däremot visar Figur 5.5 en tydlig ökning av varians för NLOS. Detta kan tyda på att filtret predikterar olika bra beroende på om det är ett stillastående eller rörligt mål. Hur medelfelet påverkas av NLOS är svårtolkat utifrån befintlig mätdata och det behövs ytterligare data för att dra slutsatser.

Kalmanfiltret som implementerades hade ingen tydlig effekt på reduceringen av de ovannämnda medelfelen, vilket är förväntat då filtret saknar förmåga att prediktera att punkterna är kraftigt förskjutna i genomsnitt. Däremot minskade filtret standardavvikelsen av mätningarna med några centimeter eftersom den filtrerar bort brus. Bruset kunde ha minimerats ytterligare genom att minska på processbruset i filtret, men då hade systemet istället blivit ännu långsammare på att reagera på ändringar i målets rörelse.

I sin helhet visar UWB-resultaten ett mycket högre medelfel och varians än förväntat på grund av den närliggande uppsättning av ankare. Effekten av siktförhållanden och Kalmanfilter är inte lika tydlig och det behövs fler test för att komma till en konkret slutsats. Användning av endast 100 mätpunkter för varje position är för lågt, vilket synliggörs i de motsägande medel- och medianvärdena som både visar förbättring och försämring. Samtidigt kunde valet av positioner runt om roboten vara mer systematisk för att tydligare undersöka hur vissa avstånd och vinklar påverkar resultatet. En annan potentiell felkälla är att platsen för uppsamlingen av ofiltrerade positioner inte var densamma som för senare test med filter. Den förstnämnda hade fler objekt kring sig som kunde reflektera UWB-signaler och medföra störningar som ger högre varians.

6.1.3 Körförmåga

Testerna för hastighet, bromssträcka och maximal belastning visade sig alla inte uppfylla kraven. Den uppmätta bromssträckan nådde faktiskt upp till kravet men detta var vid omkring hälften av den hastighet som kravspecifikationen krävt. Brister i bromstestet är att för få mätningar gjordes och att testet innebar att bryta strömtillförseln till motorerna istället för att till exempel kortsluta motorerna. I testet som gjordes gavs motorn möjlighet att "ladda ur" det uppbyggda rörelsemomentet, i form av elektromotorisk kraft, genom motorkontrollern vilket troligtvis gav upphov till den långa stoppsträckan. Den maximala hastigheten som uppmättes nådde inte heller upp till kravet. Detta tros vara en direkt orsak av att den motor som var beräknad vara mest lämplig, inte var tillgänglig och istället ersattes med en liknande motor med samma effekt, men med annan utväxling i kombination med

en 3D-printad växellåda med okänd verkningsgrad. Implementationen av växellådan förklarar också delvis resultatet i belastningstestet. Då växellådan inte var en del av planeringen för slutprodukten gjordes inga försök att utveckla den för prototypen. Den sköra 3D-printade växellådan begränsade den fjädring som kunde läggas på drivaxlarna och därmed begränsade friktionskraften mellan däcken och golvet. Detta förklarar varför däcken börjar slira då 7,5 kg lastar roboten.

Från testet av avvikelser vid färd rakt fram påvisas en stark dragning åt höger. Detta gäller dock inte alltid. Tre möjliga förklaringar till detta har identifierats till att vara olikheter mellan motorerna som gör att samma pulsvidd inte ger samma rotation. Ytterligare kan olikheter i de 3D-printade växellådorna också ge upphov till olika rotation. Den sista faktorn tros vara fel i mätningar från UWB-modulerna, då målet tenderar att vara förskjutet en halv meter åt höger. Den sistnämnda tros vara den primära orsaken till de störta avvikelserna i färdriktning.

De tester som gjordes för att utvärdera roboten under användning visar att prototypen går att använda i en butiksmiljö. I situationer då roboten har fri sikt till ledaren lyckas den alltid navigera sig fram till ledaren. Vägen den tar är sällan rak och i vissa fall åker roboten ortogonalt mot färdriktningen mot ledaren. Trots detta är det möjligt att ”hämta upp” roboten genom att gå närmare och leda den i rätt riktning om den skulle åka åt fel håll och tappa bort ledaren. I fall då ledaren går utanför synfältet kan ändå roboten mäta upp en position nära nog för att navigera runt hindret med hjälp av LiDAR-enheten som förhindrar kollisionen med hindret. Under testet i butiksmiljö kolliderade dock roboten med väggar vid diverse tillfällen. Detta kunde i vissa fall förklaras av att geometrin i den nivån där LiDAR-enheten gör mätningar inte alltid stämmer överens med nivåerna som roboten med korgen tar upp. Väggar som når hela vägen ner till marken kolliderar roboten med betydligt mer sällan, men inte aldrig. Dessa kollisioner är direkt kopplade till programvaran då LiDAR-enheten bedöms ha en tillräcklig uppdateringsfrekvens. Fel med programmet kan vara att hinderdetekteringen uppdateras för sällan och/eller att hinderdetekteringsalgoritmen är bristfällig, det vill säga att parametrarna för hinderdetekteringen felinställda eller att en annan algoritm hade kunnat implementeras för att erhålla ett bättre resultat.

6.2 Avvikelser från planering

Den prototyp som konstruerades under projektet avvek från den ursprungliga planeringen i diverse aspekter. Anledningen till dessa orsaker skiljer sig åt för varje avvikelse och presenteras nedan:

Ursrungsplanen var att använda ett UWB-chip DWM3000 som är en direkt efterträdare till chipet DWM1000 inuti MDEK1001 [33]. Planen var att programmering på själva chipet skulle erbjuda större friheter än att programmera på en helhetslösning som MDEK1001. Det visade sig att den DWM3000 varianten var begränsat i stöd i form av programmeringsbibliotek och gick inte att implementera med gruppens bristfälliga kunskaper inom inbäddade system.

Till följd av de ändringar i motor som beskrivits tidigare blev designen annorlunda från den tilltänkta. Med en växellåda i drivlinan fanns det inte längre plats för drivpaketet under roboten, detta medförde att hjulen istället stack ut cirka 10 cm utanför roboten och därmed försvann en viktig del ur designaspekten. Även hållfastheten försämrades markant, då hjulen, axlarna och växellådan var utskrivna i PETG-plast var de väldigt sköra för stötar. Även UWB-modulernas storlek gjorde att de inte längre gick att fälla in i konstruktionen som tänkt. Detta medförde att de fick monteras synligt istället.

6.3 Jämförelse med tidigare kandidat- och examensarbete

Jämfört med föregående arbete som utvecklade en autonom kundkorg [5] har stora framsteg gjorts med denna prototyp. För det första kan roboten följa en specifik person utan att förväxla den med någon annan eftersom nuvarande system förlitar sig på en unik radiosignal istället för personigenkänning. För det andra fungerar spårningen av målet utan att den är i sikte på grund av UWB-tekniken. Slutligen har kollisionsundvikningen nått betydligt längre i denna iteration då LiDAR-tekniken samlar mycket mer distansdata än ultraljudssensorer som användes i tidigare projekt. Det gör att roboten kan svänga undan olika mycket beroende på avstånd till och storleken av ett hinder. På tal om körförmåga lyckas denna iteration inte nå gånghastigheten vilket är en försämring. Däremot lider den inte av lika stora oscillationsproblem eftersom vinkelregulatorn endast aktiveras för stora fel.

Å andra sidan är UWB-efterföljningen en tydlig försämring jämfört med tidigare efterföljande robot som använder samma teknik [17]. Examensarbetet visar att mätnoggrannhet under tio centimeter kan uppnås då ankarna är placerade 40 cm ifrån varandra. I denna prototyp är avstånd mellan ankare högre men medelfelet är runt 50-70 cm. Det tidigare arbetet använder ett UWB-chip med samma antenn som på MDEK1001, men skillnaden är förmodligen annorlunda konfiguration vilket gör sensorerna olika lämpade på att mäta positioner utanför området som ankarna spänner ut. Hur exakt konfigurationen skiljer sig är komplicerat att avgöra eftersom MDEK1001 innehåller inbäddad kod och omprogrammeringen av detta låg utanför projektets ramar.

6.4 Vidareutveckling

UWB-testen visar att det finns tydliga förbättringsområden för mätnoggrannheten och precisionen av den ofiltrerade datan. Samma uppställning av ankare på korg hade kunnat användas i fortsättningen, men då bör andra UWB-moduler som är inställda för den uppställningen användas. Ett annat alternativ är att följa databladets rekommendation och placera ett flertal MDEK1001-moduler utspridda över ett rum, samtidigt som taggar ställs på varje robot. På så sätt uppnås förmodligen

högre mätnoggrannhet och precision för positionsmätningarna.

Om utspridning av MDEK1001-moduler uppnår bättre mätvärden hade omfattningen kunnat skalas upp markant utan större förändringar, med två taggar på varje robot för att avgöra riktning och position, och en tagg på varje person kan man para ihop kund och korg och låta dessa samspela i butiken. Alternativt kan det vara möjligt att endast använda en tagg per robot och istället utgöra riktning enbart genom att jämföra position med den senaste uppmätta positionen i kombination med iakttagelse av styrsignalen mellan mätningar. Med ytterligare uppgradering med planeringsalgoritmer och kartläggning av butiken via LiDAR-mappning kan man dessutom optimera robotarnas rörelser och färdvägar i butiken.

Förutom förbättringar i mjukvaran finns det flertal utvecklingsområden inom hårdvara. Nuvarande motor bör ersättas med den korrekt dimensionerade motor som tidigare benämnts för att undgå växellåda. På så sätt hade hjulen undvikit stötar genom att de placeras innanför ramen. Om vidareutveckling sker med nuvarande motor bör växellådan uppgraderas med exempelvis stålaxlar för att klara påfrestningen av framdrivningen. Processorn i Raspberry Pi 4B är underdimensionerad för den här applikationen, med detta följer att roboten inte gör tillräckligt snabba justeringar. Vidare bör någon form utav visuell representation av robotens handlingar implementeras för att underlätta för användaren att förstå vad roboten försöker göra. Detta kan vara genom exempelvis en skärm eller olika färgade dioder.

6.5 Samhälleliga och etiska aspekter

Att utveckla och skapa en autonom kundkorg innebär både möjligheter och utmaningar utifrån dels samhälleliga- men också etiska aspekter. Ur ett samhälleligt perspektiv kan tekniken bidra till ökad tillgänglighet, särskilt för äldre eller personer med nedsatt rörlighet. En självkörande kundkorg ger alltså butiker möjlighet att inkludera fler kunder.

En produkt som görs autonom kan i många fall ersätta ett jobb som en människa har kunnat göra tidigare. Idag kan butiksanställda behöva flytta på kundkorgar som kunder inte ställer tillbaka på rätt plats, men att ersätta en sådan uppgift genom att korgen återvänder till sin station ersätter bara en arbetsuppgift och inte ett jobb. Detta skapar mer tid för de anställda att göra andra arbetsuppgifter.

Det är också viktigt att tänka på hur denna teknik på kundkorgar påverkar miljön. För att minimera nyproduktion bör tekniken kunna integreras på kundkorgar som redan existerar eftersom en kundkorg ofta är tillverkad av plast. Dessutom är det viktigt att ta hänsyn till var och hur de nödvändiga komponenterna tillverkas, inte bara utifrån ett miljöperspektiv utan också med tanke på arbetsförhållandena i fabrikena där de produceras. Vidare är valet av batteri, samt dess återvinningshantering två viktiga aspekter att ha i åtanke för att säkerställa en miljömässigt hållbar produkt.

Mindre butiker kan till att börja med ha svårt att investera i denna teknik, vilket gör att den främst riktar sig till större butiker. Detta skulle kunna skapa konkurrens, där kunder väljer att gå till den butik som har dessa korgar. Däremot kan införskaffandet av autonoma kundkorgar också resultera i att kunder väljer att inte använda dem, alternativt väljer att avstå från att besöka butiken helt. Detta kan bero på att tekniken upplevs som komplicerad, vilket gör att en del kunder kan känna obehag. Äldre personer kan särskilt vara tveksamma till att använda en autonom kundkorg, vilket innebär att den möjliga lösningen, som har målet att inkludera fler människor, istället skapar svårigheter. Det är därför viktigt att tänka på hur denna teknologi kan implementeras på ett sätt som blir enkel att förstå för alla, och att det tydligt framgår hur produkten fungerar.

Eftersom ingen kamera används som personigenkänning påverkas inte kundernas integritet genom någon form av datainsamling. Om en kundkorg skulle köra in i en kund eller på ett föremål kan frågor om vems ansvaret ligger på uppkomma, är det då butiken eller kunden som bär ansvaret? Det är frågor som borde besvaras innan en butik implementerar tekniken.

7

Slutsats

Syftet med projektet var att utveckla en prototyp av en autonom kundkorg. Prototypen utvärderades genom tester baserade på kravspecifikationen, men också andra tester ämnade att utvärdera prestanda. Resultaten visade att samtliga krav inte uppfylldes, men att de centrala funktionskraven för personföljning och hinderdetektering uppnåddes. Slutligen testades prototypen i en matvarubutik där den uppvisade sagda funktionalitet, men även vissa brister.

Sammantaget uppnåddes en fungerande prototyp med grundläggande funktionalitet för personföljning och hinderdetektering. Dock kvarstår vissa utmaningar för att uppnå robust och tillförlitlig autonom navigering i mer komplexa miljöer.

Referenser

- [1] E. Landt, "Butik i Nyköping börjar med digitala mynt till kundvagnar", *SVT Nyheter*, [Online], okt. 02, 2023. URL: <https://www.svt.se/nyheter/lokalt/sormland/butik-i-nykoping-borjar-med-digitala-mynt-till-kundvagnar--4c8r09> (hämtad 2025-02-05).
- [2] L. Dunn, "AI-Powered Shopping Carts Are Here," *Consumed*, Nov, 19, 2024, [Online]. URL: https://consumed.substack.com/p/ai-powered-shopping-carts-are-here?utm_campaign=post&utm_medium=web (hämtad 2025-02-05).
- [3] R. Redman, "Kroger tests 'smart' shopping cart from Caper," *Supermarket News*, Jan, 19, 2021, [Online]. URL: <https://www.supermarketnews.com/grocery-technology/kroger-tests-smart-shopping-cart-from-caper> (hämtad 2025-02-05).
- [4] Binary Tech, "World's first self driving smart shopping cart ELI | New invention technology", *YouTube*, Nov. 26, 2018, [Video]. URL: https://www.youtube.com/watch?v=_held95qCQ4 (hämtad 2025-02-06).
- [5] K. Agneholm, A. Ahmad, J. Ekarna, S. Jansson, M. Nilsson och H. Palm, "Elektriskt assisterad kundvagn," Opublicerad, 2022.
- [6] J. C. F. Diaz, W. E. Carter, R. L. Shrestha och C. L. Glennie, *Lidar remote sensing*. 2013, vol. 2, s. 757–808. DOI: 10.1007/978-1-4419-7671-0_44. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84905918298&doi=10.1007%2f978-1-4419-7671-0_44&partnerID=40&md5=74f170e8c80025b4e963dbb614285498.
- [7] RS, "Okdo Lidar Module with Bracket Development Kit for LiDAR_LD06 Raspberry Pi SBC," 2025 [Online]. URL: <https://se.rs-online.com/web/p/sensor-development-tools/2037609> (hämtad 2025-03-25).
- [8] S.-J. Xing, F.-M. Zhang, S.-Y. Cao, G.-W. Wang och X.-H. Qu, "Arbitrary and absolute length measurement based on femtosecond optical frequency comb," *Wuli Xuebao/Acta Physica Sinica*, årg. 62, nr 17, 2013. DOI: 10.7498/aps.62.170603. URL: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-84884528536&doi=10.7498%2faps.62.170603&partnerID=40&md5=9161d70b77d88dec60b1a82ea9ca6b04>.
- [9] Qorvo US, Inc., *Getting Back to Basics with Ultra-Wideband (UWB)*, White Paper, [Online], 2021. URL: <https://www.qorvo.com/-/media/files/qorvopublic/white-papers/qorvo-getting-back-to-basics-with-ultra-wideband-uwband-white-paper.pdf> (hämtad 2025-03-25).
- [10] R. S. Kshetrimayum, "An introduction to UWB communication systems," *IEEE Potentials*, årg. 28, nr 2, s. 9–13, 2009. URL: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4803808> (hämtad 2025-03-25).

- [11] O. Norrby och J. Friberg, "Inomhuspositionering med UWB och Bluetooth: Studie av kommersiella produkter och konstruktion av eget system," examensarb., Chalmers tekniska högskola, Göteborg, Sverige, 2018. [Online]. URL: <https://odr.chalmers.se/server/api/core/bitstreams/1f4e60b1-2c9e-4f28-8cd1-81bb998f1e04/content>.
- [12] NETGEAR Team, "Difference Between 2.4 GHz, 5 GHz, and 6 GHz Wireless Frequencies," URL: <https://www.netgear.com/hub/wifi/routers/difference-2-4-ghz-5-ghz-and-6-ghz/> (hämtad 2025-03-25).
- [13] W. Chantaweesomboon, C. Suwatthikul, S. Manatrinon m. fl., "On performance study of UWB real time locating system," 2016, s. 19–24. DOI: 10.1109/ICTEmSys.2016.7467115. URL: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-84974594018&doi=10.1109%2fICTEmSys.2016.7467115&partnerID=40&md5=c5366fcfc80febf7fa2d6a2b65a6efa5>.
- [14] Decawave Ltd., *MDEK1001 Kit User Manual Module Development & Evaluation Kit for the DWM1001*, Decawave Ltd., 2017. URL: <https://www.qorvo.com/products/d/da007995> (hämtad 2025-05-09).
- [15] E. Teoman och T. Ovatman, "Trilateration in indoor positioning with an uncertain reference point," 2019, s. 397–402. DOI: 10.1109/ICNSC.2019.8743240. URL: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85068791997&doi=10.1109%2fICNSC.2019.8743240&partnerID=40&md5=125dd8247a15391ad776bc4473a9b307>.
- [16] N. M. o. I. S. S.S Anikanov, "Simple adaptive filters for data pre-processing," i (2012) *8th International Topical Meeting on Nuclear Plant Instrumentation, Control, and Human-Machine Interface Technologies 2012, NPIC and HMIT 2012: Enabling the Future of Nuclear Energy*, San Diego, USA, 2012, s. 1104–1111. URL: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-84880486082&partnerID=40&md5=495ff3b91e373228df4dce5b250423fe> (hämtad 2025-04-19).
- [17] G. Betancur och F. Treven, "A Framework for a Relative Real-Time Tracking System Based on Ultra-Wideband Technology," Examensarbete, examensarb., Chalmers tekniska högskola, 2017. URL: <https://odr.chalmers.se/items/73cfb119-9ca1-4550-8bbb-7d4da96d4715>.
- [18] G. W. och G. Bishop, *An Introduction to the Kalman Filter*, Chapel Hill, NC, USA: Department of Computer Science University of North Carolina at Chapel Hill, 2006.
- [19] A. Becker. "Multivariate Kalman Filter," Kalman Filter Tutorial. (2024), URL: <https://www.kalmanfilter.net/kalmanmulti.html> (hämtad 2025-05-03).
- [20] A. Becker. "Overview," Kalman Filter Tutorial. (2024), URL: <https://www.kalmanfilter.net/default.aspx> (hämtad 2025-05-03).
- [21] O. Krejcar, I. Spicka, R. Frischer och M. Heger, "Incremental PIC controller with handled limit states and manual settings," vol. 1, 2010, s. V1379–V1383. DOI: 10.1109/ICMEE.2010.5558526. URL: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-77957794774&doi=10.1109%2fICMEE.2010.5558526&partnerID=40&md5=0dac81345b66529385967f6236c3c67e>.
- [22] J. Fredriksson, *L13 - Actuator III.pdf*, Chalmers tekniska högskola, Maskinteknologisektion, 2024.

-
- [23] *RS PRO, 12 V dc, 6000 gcm, Brushed DC Geared Motor, Output Speed 143 rpm*, RS Components AB. URL: <https://docs.rs-online.com/7575/A700000007082249.pdf> (hämtad 2025-04-28).
- [24] *SAFETY DATA SHEET Valve Regulated Lead-Acid (VRLA) Battery*, United Kingdom, Yuasa Battery Europe Ltd, 2011. URL: <https://docs.rs-online.com/fef4/0900766b81506fde.pdf> (hämtad 2025-04-10).
- [25] R. P. Press, *Raspberry Pi 4 Model B specifications*, u.d. URL: <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/specifications/> (hämtad 2025-03-19).
- [26] R. P. Press, *Buy a Raspberry Pi Pico*, u.d. URL: <https://www.raspberrypi.com/products/raspberry-pi-pico/> (hämtad 2025-03-25).
- [27] *PRODUCT BRIEF: MDEK1001*, USA: Qorvo, 2017. URL: <https://www.qorvo.com/products/p/MDEK1001#documents> (hämtad 2025-03-19).
- [28] *PRODUCT INFORMATION: DWM1000*, USA: Qorvo, 2018. URL: <https://www.qorvo.com/products/p/DWM1000#documents> (hämtad 2025-03-19).
- [29] L. Cao, J. Chen och J. Li, “Working principle and application analysis of UART,” 2023, s. 255–259. DOI: 10.1109/EEBDA56825.2023.10090571. URL: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85154604711&doi=10.1109%2fEEBDA56825.2023.10090571&partnerID=40&md5=cf555af9db947f16cec05d>
- [30] T. Matsuda, Y. Su, Y. Xiang, K. Nguyen och H. Sekiya, “Raspbian Versus LineageOS: An Experimental Study on Raspberry Pi 4,” 2024, s. 94–98. DOI: 10.1109/ISCIT63075.2024.10793669. URL: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85216573597&doi=10.1109%2fISCIT63075.2024.10793669&partnerID=40&md5=1e9b9f5191ec45319b43decb8df6ea08>.
- [31] Y2Nk4, *LiDAR_Obstacle_Detection*, GitHub repository, 2023. URL: https://github.com/Y2Nk4/LiDAR_Obstacle_Detection (hämtad 2025-03-13).
- [32] Decawave Ltd., *DWM1001 System Overview And Performance*, User Manual, Decawave Ltd., 2018. URL: <https://www.qorvo.com/products/d/da007974> (hämtad 2025-04-28).
- [33] *PRODUCT INFORMATION: DWM3000*, USA: Qorvo, 2021. URL: <https://www.qorvo.com/products/p/DWM3000> (hämtad 2025-05-09).
- [34] Ł. Wargula, B. Wiczorek och M. Kukla, “The determination of the rolling resistance coefficient of objects equipped with the wheels and suspension system – results of preliminary tests,” *MATEC Web of Conferences*, årg. 254, s. 01005, 2019. DOI: 10.1051/mateconf/201925401005.

A

Kravspezifikation

A. Kravspecifikation

Chalmers	Dokumenttyp <i>Projekt</i>	Kravspecifikation Autonom kundkorg EENX16		
Utförare: Grupp 11 EENX16		Skapad: 2025-01-30 Modifierad: 2025-05-12		
Kriterier	Målvärde	K/Ö	Verifieringsmetod	
Funktion(er)				
Följa ledaren		K		
Navigera en statisk miljö		K		
Klara av att bromsa		K		
1 Prestanda				
1,1 Avstånd från ledaren då fordonet stannar <i>Gäller endast vid körning med tom korg. (För att inte kollidera med ledaren)</i>	<0,7 m	K	Praktiskt test	
1,2 Klara statiska laster <i>Krafter som uppstår vid en storhandling</i>	< 15 kg	K	Praktiskt test	
1,3 Klara statiska laster <i>Säkerhetsfaktor på 1,6</i>	< 25,6 kg	Ö	Praktiskt test	
1,4 Undvika kollision	Kolliderar inte med några hinder	K	Praktiskt test	
1,5 Maximal amplitud oscillering vid efterföljning <i>Maximal svängning vid efterföljning</i>	0,5 m	Ö	Praktiskt test	
1,6 Svängradie	< 0,5 m	K	Praktiskt test	
1,7 Bromssträcka <i>Gäller endast vid körning med tom korg</i>	< 0,5 m	K	Praktiskt test	
1,8 Maximalt fel i avståndsuppmätning <i>Maximalt fel i uppmätt avstånd till ledare som befinner sig 2 meter bort</i>	0,3 m	Ö	Praktiskt test	
1,9 Maximal följningsavstånd <i>Följer ett mål inom radien</i>	10 m	Ö	Praktiskt test	
1,10 Körhastighet <i>Hastighet vid arbetspunkten</i>	4-6 km/h	K	Praktiskt test	
1,11 Kan navigera dynamiska miljöer		Ö	Praktiskt test	
1,12 Kan identifiera ledaren bakom solida objekt		Ö	Praktiskt test	
2 Miljö (omgivande)				
2,1 Temperatur i användningsmiljö	15-25 °C	K	Praktiskt test	
2,2 Klara av nivåskillnad	± 2cm/m	Ö	Praktiskt test	
3 Konstruktion				
3,1 Dolda komponenter <i>Samtliga komponenter dolda innanför "skal"</i>		Ö	Visuell inspektion	
4 Storlek				
4,1 Djup	≤ 550mm	K	Mätning	
4,2 Djup	≤ 530mm	Ö	Mätning	
4,3 Bredd	≤ 430mm	K	Mätning	
4,4 Bredd	≤ 410mm	Ö	Mätning	
5 Massa				
5,1 Tjänstevikt	≤ 4,4 kg	Ö	Mätning	

Figur A.1: Kravspecifikation

B

Beräkning för val av motor

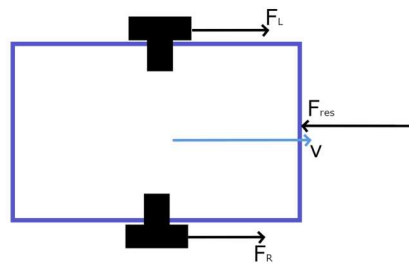
Nedan följer de beräkningar som gjordes för att bestämma den motor som behövdes till projektet.

Utifrån kravspecifikationen (se Appendix A, Figur A.1) fastställdes följande parametrar:

- **Massa:** $m = \text{Tjänstevikt} + \text{Önskad last} = 4,4 + 25,6 = 30 \text{ kg}$
- **Hastighet:** $v = 5 \text{ km/h} \approx 1,4 \text{ m/s}$
- **Acceleration:** $a \approx 0,7 \text{ m/s}^2$

Radien r_{hjul} för drivhjulen bestämdes till att vara 4,5 cm, vilket är mindre än casterhjulens totala höjd på 5,8 cm. På så sätt bär främst casterhjulen lasten och onödig belastning på drivaxlarna undviks.

I Figur B.1 visas den friläggning av roboten som användes som grund för dimensioneringen av motorerna. Figuren illustrerar roboten med dess motorer och drivhjul med de krafter som verkar på den.



Figur B.1: Friläggning av kundkorg med hjul placerade för differentialstyrning, samt de krafter som uppstår vid rörelse.

Jämviktsekvation

Summan av de krafter som verkar på vagnen ges av

$$F_L + F_R = ma + F_{\text{res}}, \quad (\text{B.1})$$

vilket kan skrivas om som

$$ma = \frac{2T_{\text{hjul}}}{r_{\text{hjul}}} - F_{\text{res}}, \quad (\text{B.2})$$

där F_L och F_R representerar kraften [N] från vänster respektive höger hjul. Luftmotståndet försummas på grund av den låga hastigheten. Rullmotståndskraften F_{res} beräknas enligt

$$F_{\text{res}} = f_t \cdot m \cdot g, \quad (\text{B.3})$$

och här representerar g tyngdaccelerationen och f_t rullningsmotståndets koefficient. Ett typiskt värde för ett icke-pneumatiskt hjul är $f_t \approx 0,035$ [34].

Moment och vinkelhastighet

Det moment T_{hjul} som verkar vid hjulet kan bestämmas genom omskrivning av ekvation (B.2)

$$T_{\text{hjul}} = \frac{r_{\text{hjul}}}{2} \cdot (ma + F_{\text{res}}). \quad (\text{B.4})$$

Den resulterande vinkelhastigheten ges av

$$\omega_{\text{hjul}} = \frac{v}{r_{\text{hjul}}}. \quad (\text{B.5})$$

Driftfaser

För roboten uppkommer två driftfaser, accelerationsfas och kontinuerlig drift. Genom att sätta in Ekvation (B.3) i Ekvation (B.4) kan momentet T_{hjul} vid accelerationsfasen beräknas till

$$T_{\text{hjul}} = 0,70 \text{ Nm.}$$

Vinkelhastigheten ω_{hjul} vid accelerationsfasen kan med den fastställda hastigheten v och hjulradien r_{hjul} sättas in i Ekvation (B.5) och beräknas till

$$\omega_{\text{hjul}} = 0 \rightarrow 30,86 \text{ rad/s.}$$

För kontinuerlig drift används samma ekvationer som för accelerationsfasen, men $a = 0$, för att beräkna T_{hjul} och ω_{hjul} . De beräknades till

$$T_{\text{hjul}} = 0,23 \text{ Nm}$$

$$\omega_{\text{hjul}} = 30,86 \text{ rad/s} = 294,9 \text{ RPM.}$$

Effekten under kontinuerlig drift beräknas till

$$P = T_{\text{hjul}} \cdot \omega_{\text{hjul}} = 7,15 \text{ W} \quad (\text{B.6})$$

Med beräknat krav för moment och vinkelhastighet för accelerationsfas och för nominell drift, samt nominell effekt kan dessa jämföras med specifikationer för kommersiella motorer för att hitta en motor som uppfyller kraven.

C

Kostnader för prototypens komponenter

De komponenter som köptes in för projektet presenteras i Tabell C.1. De komponenter som inte nämns här har antingen skänkts eller lånats ut till projektet.

Tabell C.1: Komponentlista med inköpspriset för varje komponent samt det sammanlagda priset för alla komponenter. Samtliga priser är uttryckt i SEK.

Komponent	Antal	Styckpris (inkl moms)	Sammanlagt pris
Raspberry pi 4 4GB	1	891	891
Okdo Lidar module	1	681	681
RS PRO Brushed Geared DC Geared Motor, 19.8 W, 4.5 → 15 V dc, 59 Ncm, 120 rpm, 6mm Shaft Diameter	2	455	910
Casterhjul (4 pack)	1	39	39
Raspberry pi pico	1	79	79
		Totalt pris:	2600

INSTITUTIONEN FÖR ELEKTROTEKNIK
CHALMERS TEKNISKA HÖGSKOLA
Göteborg, Sverige
www.chalmers.se



CHALMERS