



Konenäön hyödyntäminen robotilla tehtävissä maalaus- ja liimaussovelluksissa

Joona Joutsimäki

OPINNÄYTETYÖ
Toukokuu 2021

Konetekniikka
Koneautomaatio

TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Konetekniikka
Koneautomaatio

JOUTSIMÄKI, JOONA:

Konenäön hyödyntäminen robotilla tehtävissä maalaus- ja liimaussovelluksissa

Opinnäytetyö 55 sivua, joista liitteitä 0 sivua
Toukokuu 2021

Opinnäytetyön tavoitteena oli tutkia konenäön hyödyntämismahdollisuuksia erilaisissa robotin ohjaussovelluksissa. Työn toimeksiantajana toimi Nipema Oy, joka on erikoistunut liimaus- ja pintakäsittelylaitteisiin sekä -tuotantolinjoihin. Konenäkö on yhä suosiotaan kasvattava ja kykenevä apuväline monissa eri teollisuuden sovelluksissa, myös robotin ohjauksessa. Konenäkökameroissa voi olla hahmontunnistusta kappaleen paikantamiseen kuvausalueelta tai käyttäjä voi erilaisilla reunantunnistus- tai pituudenmittaustyökaluilla laskea arvoja, joilla voidaan ohjata robotin liikkeitä.

Tutkittava tilanne pohjautuu asiakaslähtöiseen ongelmaan. Projektissa saadut tulokset tähän yritykselle suhteellisen vieraaseen aihealueeseen olivat laitteiston käyttömahdollisuuksiin tutustuminen ja sitä kautta saadut näkemykset aiheesta. Näitä yritys voi hyödyntää tulevaisuudessa. Toteutus jakautui kolmeen osa-alueeseen, joita olivat Omronin konenäkökamera ja Yaskawan robotti sekä kokonaisuutta ohjaava Siemensin Step 7 -logiikka. Laajasta kokonaisuudesta konenäkö oli tämän opinnäytetyön pääaihe. Projekti aloitettiin tutustumalla yleisesti aihealueiden teorioihin, minkä jälkeen tutustuttiin tarkemmin käytössä olevaan laitteistoon ja niiden ominaisuuksiin. Loppuvaiheilla tehtiin matka yrityksen pääkonttorille, jossa päästiin kokeilemaan osa-alueista yhdistettyä kokonaisuutta.

Lopputuloksena opinnäytetyössä saatiin pystytettyä demolaitteistoa, jota voidaan hyödyntää jatkossa palatessa aiheen pariin. Lisäksi saatiin tutkittu kokonaisuus, pientä hienosäätöä lukuun ottamatta, toimimaan valmiin järjestelmän kaltaisesti. Raportin lopussa on lisäksi esitetty ajatuksia aiheen jatkokehityskohteista, kuten miten järjestelmää voitaisiin laajentaa tai millä kuvauspuolta voitaisiin helpottaa.

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Mechanical Engineering
Machine automation

JOUTSIMÄKI, JOONA:
Utilisation of Machine Vision in Robotic Painting and Gluing Applications

Bachelor's thesis 55 pages, appendices 0 pages
April 2021

The aim of this thesis was to study the possibilities of utilising machine vision in different robotic applications. The thesis was made in association with a company called Nipema Oy, which specialises in surface treatment and gluing devices and assembly lines. Machine vision has become more and more common and it is a capable tool in many industrial applications, including robotics. It can be used in pattern recognition for part locating within the vision area of the camera. In addition, user can place edge detection or distance measurement tools to calculate values that control the robot's movement.

The studied situation was based on a client-oriented problem. The results of the project on this relatively unfamiliar topic for the company were to get acquainted with the possible usage of the equipment and to gain insights on the topic. The implementation was separated into three parts: Omron's machine vision camera, Yaskawa's robot, and Siemens' Step 7 logic, which operates the whole system. From the vast entirety, the machine vision was the main focus of this thesis. The project started by studying the theories of the subject areas. This led into studying the equipment and their features used in the project. The last step was to travel to the company's headquarters to test the built-up entity of the system consisting all the different sections.

The outcome of this thesis was the assembled setup of the subsystems, that can be used in the future research among the subject. In addition, the studied entity was tested with the actual equipment and was proven to be operational with minor limitations. At the end of the report, subject's further development ideas, regarding the solutions for easing the system's operation and expansion, are shared.

Key words: machine vision, robotics, PLC

SISÄLLYS

1	JOHDANTO	6
2	NIPEMA OY	7
3	KONENÄKÖ	8
	3.1 Yleistä teoriaa	8
	3.2 Konenäön käyttökohteita	11
	3.3 2D-konenäkö	12
	3.3.1 Tekniikkaa	12
	3.3.2 Esimerkkitapaus tehokkaasta konenäköjärjestelmästä	13
4	OHJELMOITAVA LOGIIKKA	14
5	ROBOTIIKKA	16
6	OMAT TUTKIMUKSET	20
	6.1 Konenäkö	20
	6.1.1 Case 1: Kuvaus kohtisuoraan takaapäin	21
	6.1.2 Case 2: Kuvaus viistosta	22
	6.1.3 Case 3: Kuvaus laatikon sisältä	25
	6.1.4 Case 4: Kaksi kuvaa	26
	6.2 PLC	26
	6.3 Robotti	29
7	TOTEUTUS	32
8	JATKOKEHITYS	38
	8.1 Järjestelmän laajentaminen	38
	8.1.1 Useampi kappale	38
	8.1.2 Kappaleen vapaampi asento	40
	8.2 3D-konenäkö ja sen eri tekniikat	40
	8.2.1 Stereonäkö	41
	8.2.2 Struktuurivalo	43
	8.2.3 Laser kolmiomittaus	45
	8.2.4 Time of flight	47
	8.2.5 3D-konenäön hyödyntäminen	49
9	POHDINTA JA YHTEENVETO	51
	LÄHTEET	53

ERITYISSANASTO tai LYHENTEET JA TERMIT (valitse jompikumpi)

PLC	Programmable logic controller (ohjelmoitava logiikka)
ToF	Time of flight (pituudenmittausmuoto, joka perustuu signaalin kulkuajan mittaukseen)
FBD	function block diagram (ohjelmointikieli)
ST	Structured text (ohjelmointikieli)
VGR	vision guided robotics (konenäkö ohjattu robotiikka)
HDR	High dynamic range (laaja dynaaminen alue, käytettäessä näyttää kuvissa tummat ja kirkkaat alueet paremmin)
HMI	Human-machine interface (näyttöpääte)
FC	Function (Tia Portalin aliohjelmalohko)

1 JOHDANTO

Opinnäytetyössä tutkittavaksi aiheeksi saatiin Nipema Oy:ltä konenäön hyödyntäminen maalaus- ja liimaussovelluksissa. Aihe syntyi yrityksen omasta tarpeesta erääseen esimerkkitilanteeseen, jossa traverssimaalauksessa olleen laatikon mallisen kappaleen sisäkulmat eivät tule riittävän hyvin maalatuksi. Tämän seurauksena työntekijä joutuu suorittamaan paikkamaalauksen, jotta saavutetaan hyväksyttävä maalipinta. Yrityksellä oli ajatuksena saada konenäkökamera tunnistamaan kappaleet ja välittämään sen paikkatiedot robotille, joka tekisi paikkamaalauksen ihmistyöntekijän sijaan. Tilanteessa on kuitenkin haasteita, sillä kuljettimella liikkuvat kappaleet ripustetaan koukuilla sen kyytiin ja nämä koukut useasti aiheuttavat pientä muutosta kappaleen asentoon.

Toteutus jakautuu kolmeen osa-alueeseen, jotka ovat konenäkökamera, logiikka sekä robotti. Alussa käsitellään teoriaa ja yleistä tietoa, jotta lukijalle muodostuu käsitys käytössä olevista aihealueista. Tämän jälkeen käydään läpi tarkemmin omia tutkimuksia, joilla tilannetta lähdettiin selvittämään. Konenäkökamera ja logiikka oli alusta alkaen fyysisesti kokeiluissa käytössä, mutta robotin sijaitessa yrityksen pääkonttorin tiloissa, piti siihen tutustua simulointiohjelman avulla.

Opinnäytetyön aihe on hyvä koneautomaation opiskelijalle, sillä kaikkia sen aihealueita sisältyy koulutussuunnan opetussuunnitelmaan. Aihe on myös hyvin ajan-kohtainen, sillä vaikka yksittäiset osa-alueet ovatkin olleet jo useampia vuosia olemassa ja käytössä, ovat ne silti koko ajan kehittymässä. Erityisesti automaattikka, joustavuus ja jonkin tason autonomisuus ovat haluttuja piirteitä tulevaisuuden koneilta ja järjestelmiltä, minkä mahdollisuutta myös tämä opinnäytetyö osaltaan tutkii.

2 NIPEMA OY

Nipema Oy on vuonna 1986 perustettu perheyritys ja se suunnittelee ja valmistaa teollisuuden erikoiskoneita. Yrityksen vahvinta osaamisaluettaan on pintakäsittelyyn ja liimaukseen liittyvät koneet, lisälaitteet ja tuotantolinjat. Projektit toteutetaan tiiviissä yhteistyössä asiakkaan kanssa, jolloin kokonaisuus voidaan räätälöidä asiakkaan tarpeiden mukaan. Koneiden ja linjojen tuotannossa Nipeman ammattitaitoinen henkilökunta hoitaa suunnittelun, kokoonpanon, sähköistyksen, asennuksen ja käyttöönoton. Toiminnan tukena on tunnetut kansainväliset päämiehet. (Nipema 2021).

Nipeman tarjoamia tuotteita on laaja skaala maalaamo-, liimaus- ja lujitemuovilaitteista aina robotteihin asti. Maalaamolaitteista Nipema toimittaa monia erilaisia Aabo-Ideal pintakäsittelylinjoja sekä Craco-maalauslaitteita. Liimaus- ja annostelutekniikan osalta tarjotaan 1- ja 2-komponenttiliimalaitteita sekä kuumaliimalaitteita ja niiden tarvikkeita. Lujitemuovilaitteisiin liittyen tarjotaan geeli- ja hartsiruisuja sekä niiden lisätarvikkeita. Robottien puolelta Nipema tarjoaa eri kokoluokan Yaskawa Motoman sarjan robotteja. Lisäksi yrityksellä on tarjota PLC- ja PC-ohjauksia koneiden ja tuotantolinjojen ohjaukseen. (Nipema 2021).

Vuoden 2019 liikevaihto yrityksellä oli 1,75 miljoonaa euroa ja se nousi 17,8 % edelliseen vuoteen verrattuna (Finder 2021). Nipemalla on viime vuosien aikana sujunut hyvin ja kasvua on tapahtunut tasaisesti. Se näkyy liikevaihdon nousussa sekä kasvaneessa henkilöstön määrässä. Vuoden 2018 alussa yhtiön pääkonttori Savitaipaleella muuttikin suurempiin tiloihin. Yrityksellä on toimipisteet Savitaipaleen lisäksi kahdessa kaupungissa; Keravalla sijaitsee myyntikonttori ja Kangasalla tapahtuu ohjausjärjestelmien valmistus. (Nipema 2021).

3 KONENÄKÖ

3.1 Yleistä teoriaa

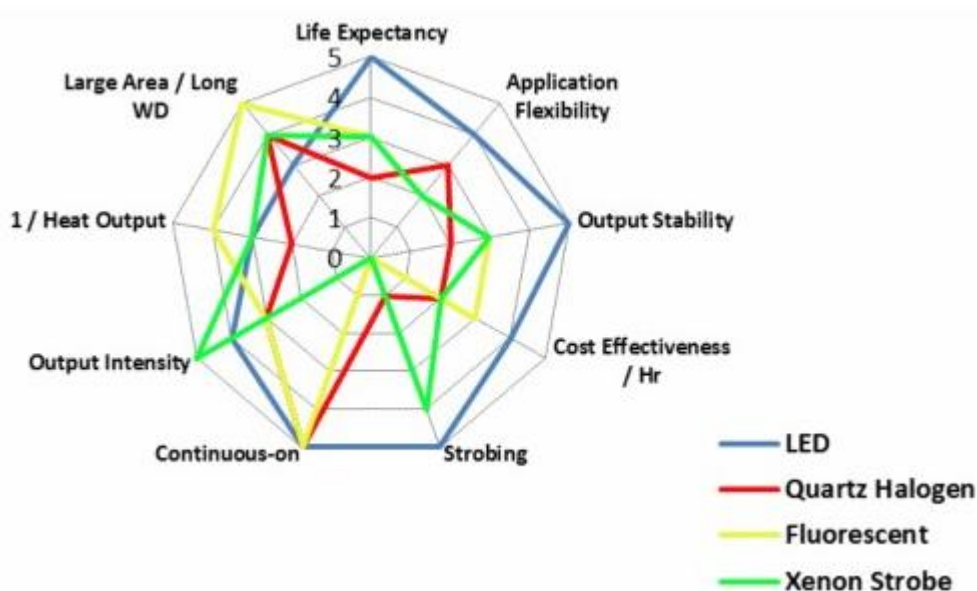
Konenäkö on eräänlainen anturoinnin muoto siinä missä esimerkiksi lämpötilaa tai läsnäoloa tunnistavat anturit. Toisin kuin muissa yleisimmissä anturoinnin muodoissa, on konenäkö hyvin laaja ja joustava tapa kerätä tietoa, ja tämän takia se on yleistynyt nopeasti eri teollisuuden aloilla viime vuosikymmenen aikana. Konenäköjärjestelmä osaa tulkita digitaalisen kuvan pikseliarvoja ja tehdä toimenpiteitä tähän tietoon perustuen. Tätä tietoa voidaan lähettää eteenpäin logiikalle tai kameralta saadaan kytkettyä suoraan outputteja toimilaitteisiin tai logiikkaan.

Tärkein komponentti konenäköjärjestelmässä on itse kamera (KUVA 1). Kamera määrittelee hyvin pitkälle tilanteesta saatujen kuvien laadun ja tunnistuksen tarkkuuden. Oikean kameran valintaan vaikuttaa mihin käyttötarkoitukseen konenäköjärjestelmä tulee. Tarvitaanko käyttökohteeseen värin tunnistusta vai riittääkö harmaasävykamera? Vaaditaanko kameralta hyvää tarkkuutta esimerkiksi pienten yksityiskohtien kuten tekstin erottamiseen? Halutaanko, että kamerassa itsessään on säätöjä esimerkiksi aukon koon tai suljinajan suhteen, millä saadaan vaikutettua kuvan kirkkauteen. On myös kameroita, joihin voidaan vaihtaa linssi ja sillä voidaan muokata kameran optiikkaa ja näin ollen vaikuttaa kameran näkökenttään. Kameraa valittaessa ei kannata valita automaattisesti markkinoiden parhaiten varustelluinta, sillä se ei monesti ole tarpeellista. Yksinkertaiseen toteutukseen voi riittää pelkkä harmaasävykamera ilman säätöominaisuuksia, joka tulee olemaan huomattavasti edullisempikin vaihtoehto. Lisäksi kyvykkäämissä kameroissa saattaa kuvan käsittely kestää pidempään ja se voi aiheutua pullonkaulaksi suurina nopeuksia vaativissa kohteissa.



KUVA 1. Basler ACE 2 Pro konenäkökamera (Basler)

Seuraavaksi suurin vaikuttaja konenäköjärjestelmän onnistumisen takaamisessa on valaistus. Valaistuksella saadaan aikaan se, että kamera näkee kappaleen sekä sen piirteet. Valaistuksen onnistumiseen voi vaikuttaa valon tyypillä tai valaisun tyylillä. Yleisimmät käytössä olevat valonlähteet ovat ledit, xenonit, loisteputket ja halogeenilamput. Eri valonlähteillä on omia hyötyjä ja haittoja, kuten kuvassa (KUVA 2) näkyy. Valaisutyyliä taas on esimerkiksi tausta-, suora-, rengas- ja palkkivalo. Joissain kameramalleissa on integroitu led valaistus linssin ympärillä. Se, mitä valaistus kombinaatiota kannattaa käyttää riippuu paljon kuvattavasta kohteesta, kuvaus ympäristöstä sekä mitä piirteitä kuvasta halutaan saada esille. Väärällä valaistuksella voidaan aiheuttaa se, ettei haluttuja piirteitä saada näkyviin tai jopa saadaan haitalliset piirteet näkyviin, kuten kuvassa (KUVA 3) näkyy.



KUVA 2. Eri valonlähteitä ja niiden ominaisuuksia (ni)



KUVA 3. Vasemmalla onnistunut valaistus, oikealla epäonnistunut (ni)

Huomioon otettava elementti konenäköjärjestelmää suunniteltaessa on kuvattavan kappaleen tausta. Oikeanlaisella taustalla saadaan korostettua kontrastieroa kappaleen ja taustan välillä, mikä yleisesti ottaen helpottaa kappaleen tunnistamista. Tummalle kappaleelle ei kannata valita tummaa taustaa, sillä se voi johtaa siihen, että tunnistettavan kappaleen piirteet hukkuvat taustaan. Tausta olisi hyvä saada mahdollisimman yksinkertaiseksi ja häiriöttömäksi. Esimerkiksi taustalevyssä oleva saumakohta saattaa aiheuttaa kameralle haamutunnistuksia eli häiriöin aiheuttamaa vahingollista tunnistusta. Lisäksi jo varjon muodostuminen kuvattavan kohteen päälle ja taustalle voi aiheuttaa häiriöitä tunnistukseen.

Konenäköjärjestelmän fyysisen rakenteen jälkeen, sille täytyy vielä konenäköohjelmassa opettaa tarkasteltava kohde eli referenssikuva. Tätä varten tarvitaan tilanteesta esimerkkikuva tai -kuvia. On aina kameras malli- ja tilannekohtaista,

tarvitaanko opetuskuvia yksi vai useampia. Kameravalmistajilla on mallikohtaisia ohjelmia, joilla käyttäjä voi asettaa tunnistusparametrit mallikuvaan. Järjestelmä vertaa myöhemmin ottamiaan kuvia tähän referenssikuvaan ja osaa jopa sekunnin kymmenyksen aikana käsitellä kuvan ja saada siitä halutun tiedon selville. Tieto voi olla esimerkiksi laaduntarkistus tai kappaleen aseman selvitys. Näitä yksittäisiä tunnistusohjelmia, joita kutsutaan valmistajakohtaisesti esimerkiksi jobeiksi tai sceneiksi, voi olla kameralla useampia, jolloin yhdellä kameralla voidaan tunnistaa useampia kappaleita tai tilanteita. Kuvien käsittely vaatii laskentatehoa ja kameroita onkin kahdenlaisia. On olemassa niin sanottuja älykameroita, joissa on integroitu prosessori ja muisti, mitkä osaavat toimia itsenäisesti. On myös kameroita, jotka vaativat ulkoisen älyn ja nämä kytketään esimerkiksi tietokoneeseen.

3.2 Konenäön käyttökohteita

Konenäön käyttökohteita voidaan jakaa karkeasti neljää osa alueeseen, joita on: tarkistus, lukeminen, paikoitus ja laskeminen. Nämä neljä kategoriaa pitää sisällyttää suurimman osan konenäkösovelluksista. (machinevision.co.uk).

Tarkistussovelluksia on esimerkiksi laadun- ja läsnäolontarkistukset. Laaduntarkistuksessa voidaan nopeasti selvittää, onko esimerkiksi piirilevyssä kaikki tarvittavat komponentit kiinni ja onko ne oikein asennettu. Myös naarmut, rypyt ja muut epämuodostumat nähdään. Läsnäolontarkistuksessa voidaan esimerkiksi tarkistaa, puuttuuko pakkauksesta tuotteita.

Lukemisen sovelluksissa voidaan lukea tekstiä tai erilaisia koodeja. Koodit voivat olla viiva-, QR- tai datamatriisikoodeja. Lukemalla kappaleesta sen tunniste, saadaan selville mikä se on. Tätä tietoa voidaan hyödyntää esimerkiksi kappaleiden lajittelussa, seurannassa tai kirjanpidossa.

Paikoitussovelluksessa selvitetään kappaleen paikka ja/tai asento. Tällä voidaan esimerkiksi varmistaa, että liukuhihnalla liikkuva kappale on oikeassa asennossa ja se saa päästä eteenpäin kokoonpanovaiheeseen. Konenäkö osaa myös paikantaa kappaleen koordinaatit ja rotaation, sekä parhaimmillaan kiertymät. Nämä

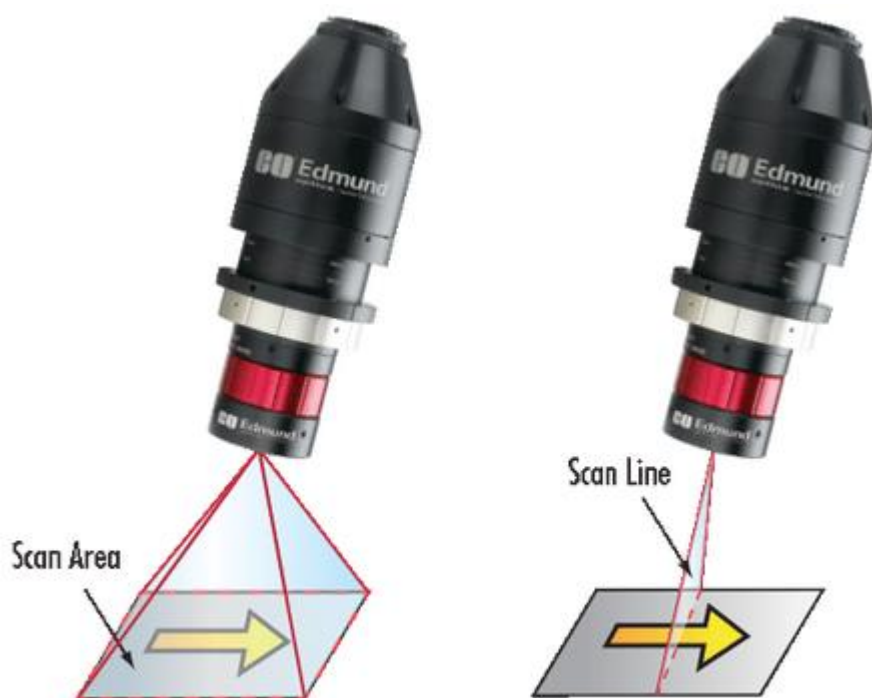
tiedot voidaan siirtää esimerkiksi robotille ja robotti osaa asemoitua kappaleeseen nähden oikealla tavalla. Tämä onkin yksi kasvava robotiikan ala, jota kutsutaan VGR:ksi (vision guided robotics).

Laskemissovelluksessa voidaan laskea yksinkertaisesti kuvassa esiintyvien kappaleiden määrä. Konenäöllä voidaan laskea myös esimerkiksi naarmujen tai muiden vastaavien määrät, jos sellaiselle on tarvetta. Myös tiettyjä pituuksia ja etäisyyksiä saadaan laskettua. Voidaan määrittää helposti kappaleen reunojen välinen etäisyys eli kappaleen pituus tai voidaan määrittää reikien keskipisteiden välinen etäisyys. Laskenta tapahtuu kuvan pikseleiden avulla ja kalibroimalla pikselitiedot, voidaan ne tiedot muuttaa halutuiksi pituusyksiköiksi.

3.3 2D-konenäkö

3.3.1 Tekniikkaa

2D-konenäkö pystyy tulkitsemaan kahden kohtisuoran akselin, useimmiten x- ja y-akseleiden, muodostamaa tasomaista aluetta ja siinä esiintyvien pikseleiden arvoja. Kappaleen syvyysuuntaista sijaintia kameraan nähden ei tällä menetelmällä kuitenkaan saada selville. On myös mahdollista luoda 2D-konenäkö kuvamalla vain yksiulotteisesti. Tällöin kuvataan viivan/linjan mukaisesti yhden tai muutaman pikselin korkuisia pikselirivejä kerrallaan. Tällöin joko kameran tai kappaleen täytyy liikkua kuvien välissä aina seuraavalle riville ja konenäköohjelma osaa siten näistä riveistä osa osalta kasata kokonaisen kuvan käsittelyä varten. Näiden kahden tekniikan erot hahmottuvat alla olevassa kuvassa (KUVA 4). (Manufacturing automation).



KUVA 4. Vasemmalla alue-kuvaus ja oikealla linja-kuvaus (Edmundo optics)

2D-konenäöllä saadaan toteutettua kaikkia edellisessä kappaleessa mainittuja sovellutuksia, ainakin jossain määrin. 2D-konenäköjärjestelmä on myös parhaimmillaan yksinkertainen toteuttaa, sillä siihen riittää vain yksi kamera.

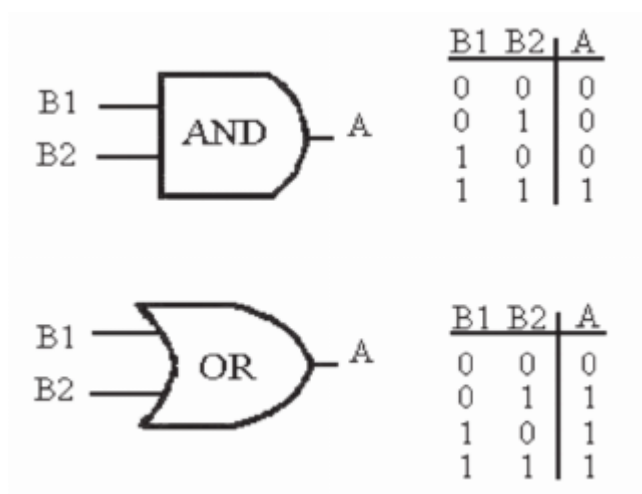
3.3.2 Esimerkkitapaus tehokkaasta konenäköjärjestelmästä

Hyvä esimerkki tehokkaasta konenäköjärjestelmästä on ihmissilmälle liian nopeasti tapahtuva laadunvalvonta. Tällainen tilanne löytyy Ranskasta, Heinekenin pullotuslinjalta, jolla kulkee 22 pulloa sekunnissa eli noin 80000 pulloa tunnissa. Siellä käytössä on Allied Visionin Mako G-030 kamera. Kameran sensori on vain 0,3 megapikseliä, mikä pieneltä kuulostaessaan on kuitenkin aivan riittävä kyseiseen sovellukseen. Täydellä resoluutiolla kamera pystyy ottamaan yli 300 kuvaa sekunnissa, mutta tässä tapauksessa kuva-aluetta oli pienennetty mahdollistaen kameran ottavan yli tuhat kuvaa sekunnissa. Tällä asetelmalla järjestelmä pystyy tallentamaan kuvia tunnin ajalta muistiin. Näitä tallenteita tarkastellen huoltotiimi voi tunnistaa ja korjata vian syyn keskeyttämättä tuotantoprosessia tarpeettomasti. Tällaisella kattavalla konenäköjärjestelmällä Heineken saavuttaa käytännössä epäonnistumisprosentin, joka on nolla. (Carroll).

4 OHJELMOITAVA LOGIIKKA

Ohjelmoitava logiikka eli PLC (programmable logic controller) on eräänlainen tietokone, jota käytetään erilaisissa automaation sovelluksissa ja laitteiden ohjauksissa. PLC-järjestelmä koostuu tuloista eli inpueteista ja lähdöistä eli outputeista, mitä PLC ohjelma ohjaa. Inputit voivat olla esimerkiksi nappeja, kytkimiä tai antureita eli järjestelmää ohjaavia komponentteja. Outputit voivat taas olla esimerkiksi valoja, venttiileitä, moottoreita tai muita vastaavia toimilaitteita eli järjestelmän ohjattavia komponentteja. (Control systems & automations).

PLC voi käsitellä kahden tyyppisiä input/output signaaleja. Yksinkertaisempi ja selvästi yleisempi on digitaalinen signaali. Tällä voi olla ainoastaan kaksi arvoa, jotka ovat 1 tai 0. Näistä voidaan puhua myös nimikkeillä HIGH tai LOW sekä ON tai OFF. Tämä signaali on nimestään päätellen joko aktiivinen tai ei aktiivinen. Digitaalisia signaaleja ohjataan Boolean algebran eli niin sanotun logiikan mukaisesti. Näitä loogisia toimintoja on esimerkiksi JA- sekä TAI-portit, joilla voidaan muodostaa ohjelmallisia ehtoja, joita noudattaen järjestelmä toimii. Alla esitetty kuva näistä kahdesta yksinkertaisesta portista totuustaulukoineen (KUVA 5). Näitä portteja on monenlaisia muitakin, millä saadaan omanlaisia toiminnallisuuksia, mutta nämä mainitut ovat välttämättömästi jokaisen ohjelman perustaa.



KUVA 5. JA sekä TAI-portit, sekä niiden totuustaulukot (Researchgate)

Toinen signaalityyppi on analoginen. Toisin kuin digitaalisessa signaalissa, analogisessa signaalissa on ääretön määrä arvoja tietyllä välillä ja sen tarkkuus rajoittuu analogia-digitaalimuunnoksen resoluution perusteella. Signaali voi olla esimerkiksi käyttäjän asettama arvo potentiometrillä tai anturoitava arvo. Anturoitava signaali voi olla esimerkiksi virtaviesti 4 – 20 mA tai jänniteviesti 0 – 10 V. Signaali täytyy vielä muuttaa PLC-ohjelman käyttämään muotoon, jotta sitä pystytään käyttämään osana ohjelmaa. Useasti analogisille signaaleille asetetaan tietyt rajat, joiden ylittyään tai sisällä ollessaan logiikkaohjelma ohjaa digitaalista signaalia.

Ohjelmoitavan logiikan ohjelmarakenteet saattavat erota toisistaan jo valmistajaja mallikohtaisesti. Esimerkiksi Siemensillä on LOGO! sekä Simatic S7, jotka eroavat ohjelman ulkoasultaan paljonkin. Tämän lisäksi on olemassa muutamia standardoituja ohjelmointikieliä, joita PLC:issä käytetään. Nämä vaihtelevat FBD:n (function block diagram) graafisesta selkeämmästä ulkoasusta ST:n (structured text) perinteisempää koodausta vastaavaan kirjoitettavaan ulkoasuun. Se, mitä ohjelmointikieltä käyttää, on pitkälti ohjelmoijasta kiinni. Tosin on myös tilanteita, missä tietty ohjelmointikieli helpottaa ja selkeyttää tietyn toiminnallisuuden tekemisessä.

5 ROBOTIIKKA

Robotiikka on tieteenlaji ja käytäntö robottien suunnitteluun, valmistamiseen ja soveltamiseen. Robotti on standardien mukaan käyttömekanismi, joka on ohjelmoitavissa kahden tai useamman akselin suhteen ja jossa on tietty autonomia, ja se liikkuu ympäristössään määrättyjen tehtävien suorittamiseksi (ISO/TC 299).

Perinteisemmin robottien ajatellaan kuuluneen teollisuuden suuriin tehtaisiin tekemään itsenäisesti töitä. Tämä oli totta vielä vuonna 2005, jolloin jopa 90 % kaikista roboteista sijaitsivat autotehtailla. Isoa muutosta on tapahtunut, sillä nykyäänä sama luku on jo alle 50 %. Robotit alkavat olemaan osana tekemistä useilla eri aloilla aina terveydenhuollosta maatalouteen ja niitä on jopa kotitalouksissa. Covid-pandemian jälkeen palvelurobottien sektorin uskotaan kasvavan teollisuusrobotteja nopeammalla tahdilla (Anandan). (Daley).

Robottien hankkimiselle on paljon syitä. ”Likaiset, tylsät ja vaaralliset, kuvastavat tehtäviä, mitä haluamme robottien tekevän” sanoo Esben Østergaard, eräs Universal Robots yhtiön perustajista (Østergaard). Ennen kaikkea robotilla voi parantaa työn tehokkuutta, nopeutta ja laadukkuutta. Robotit eivät tarvitse taukoja samanlailla kuin ihmiset, ja jos työn muut edellytykset ovat kunnossa, voi robotti työskennellä tauotta kellonympäri. Hyvin suunniteltuna robottisolun nopeus voi olla paljonkin ihmistä nopeampi. Roboteilla on myös äärimmäisen hyvä toistotarkkuus, lähes poikkeuksetta alle 1 millimetri, ja roboteilla ei tapahdu niin sanottuja inhimillisiä virheitä, joten robottien työn laatu on erinomainen. Tämän kaiken lisäksi robotteja voidaan lähettää työskentelemään ihmisille haitallisiin tiloihin tai haitallisten kappaleiden kanssa. (HowToRobot).

Teollisuudessa käytettäviä robotteja on muun muassa nivel-, rinnakkais-, scara- ja gantry-robotit. Alla esitetty kuvat edellä mainituista robottimalleista (KUVA 6 – KUVA 9).



KUVA 6. Nivellettyrobotti (ABB)



KUVA 7. Rinnakkaisrobotti (ABB)



KUVA 8. Scararobotti (Allied automation)



KUVA 9. Gantryrobotti (FLT)

Teollisuudessa robotteja käytetään muun muassa kappaleiden siirtelyssä. Se voi olla raskaiden kappaleiden siirtämistä työpisteeltä toiselle, pienien kappaleiden todella nopea lajittelu tai CNC-koneen lastaaminen uudella materiaalilla ja valmiin kappaleen purkaminen koneesta. Roboteilla voidaan myös hoitaa kokoonpanolinjojen töitä. Näitä voi olla esimerkiksi osien pulttaaminen toisiinsa, kappaleiden yhteen hitsaaminen tai komponenttien juottaminen piirilevyyn. Roboteilla voidaan hoitaa myös kappaleiden työstäminen. Se voi olla kappaleiden hiominen, kiillottaminen, purseenpoisto, reikien poraaminen tai muu vastaava. Roboteilla voidaan hoitaa jälkikäsittelyä, kuten maalaamista sekä erilaisia tiivistys- tai liimaustoimenpiteitä. Gantry- ja rinnakkaisrobotit soveltuvat parhaiten kappaleiden siirtelyyn. Scararobotti soveltuu siirtelyn lisäksi kokoonpano- sekä tiettyihin kappaleenkäsittelytehtäviin. Nivellettyrobotti on kaikkein parhaiten sovellettavissa eri tehtäviin yleisesti laajemman vapausastemäärän vuoksi. (Shake).

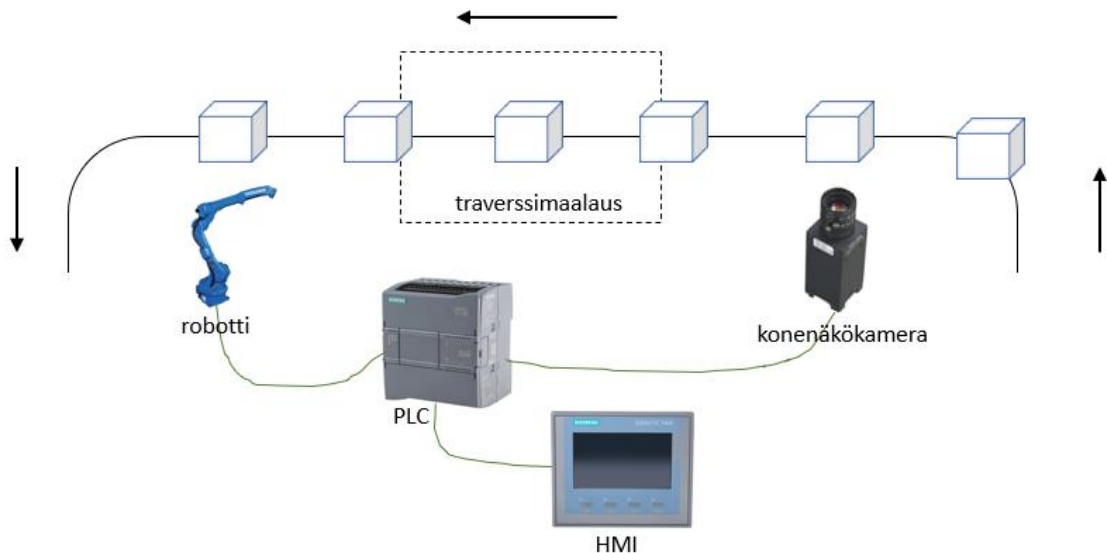
Teollisuudessa otetaan yhä enemmän käyttöön yhteistyörobotteja eli niin sanottuja kobotteja. Nämä kobotit on kehitetty olemaan turvallisia versioita roboteista, joiden työtilassa ihmisenkin on turvallista olla. Niiden kuormankäsittelykyky ja liikenopeedet on suunniteltu riittävän pieniksi, jolloin mahdollisissa törmäyksissä säästytään vahingoilta. Lisäksi kobottien ominaisuuksiin kuuluu pysähtyminen turvallisuussyistä, jos se kohtaa pienenkin vastuksen liikeradallaan (Pöysti). Vuonna 2019, noin 4,8 % asennetuista teollisuusroboteista oli kobotteja, mikä on 11 % kasvu vuoteen 2018 nähden (IFR). Niiden ennustetaan saavuttavan 29 % markkinaosuus kaikista teollisuusroboteista seuraavan vuosikymmenen aikana (Anandan).

Toinen kehityskohde teollisuusroboteille on ollut konenäkökameran yhdistäminen robotin ohjaukseen, josta mainittiin jo ohimennen aikaisemmin tässä raportissa. Tästä yhdistelmästä puhutaan nimellä VGR eli vision guided robotics. Robotti vaatii aina toimiakseen ohjelman. Robotti voidaan esimerkiksi ohjata poimaan kappaleita tietyistä paikasta. Ohjelma toimii hyvin niin kauan, kunnes kappaleiden asento muuttuu. VGR mahdollistaa esimerkiksi sen, että konenäkökamera antaa kappaleesta kappalekohtaisesti koordinaatit robotille. Ohjelma pysyy taustalla muuten samana, vain tarttumispisteen koordinaattien vaihdellessa. Tässä tapauksessa kappaleesta tarttuminen saadaan tapahtumaan aina oikeasta kohdasta ongelmia synnyttämättä. Hyöty moninkertaistuu, kun puhutaan esimerkiksi robotin tekemästä maalauksesta, hitsauksesta, liimaamisesta tai koneistamisesta, joissa tapahtuvissa virheissä on suuremmat vaikutukset. Ajatellaan esimerkiksi tilannetta, jossa runko on asettunut huonosti hitsausjigiin johtaen pieneen muutokseen sen asennossa. Tämä voi johtaa hitsauksessa tapahtuneen virheen seurauksena valmiissa tuotteesta rakenteelliseen heikkouteen. Jos valmis tuote on esimerkiksi auton runko, aiheutuu tästä vääjäämättä iso ongelma. Tässä tapauksessa konenäköjärjestelmä olisi havainnut rungon asennon muutoksen ja sopeuttanut hitsausohjelman kompensoimaan asento virhettä ja rakenteelliselta heikkoudelta olisi vältytty. Aiemman esimerkin ollessa epätodennäköinen ääripään tapaus, havainnollistaa se kuitenkin hyvin, mitä puutteita VGR voi aiempaan nähden paikata. (Anwekar).

Kolmas teollisuudessa ja myös muilla aloilla yleistynyt robottimalli on mobiilirobotit. Mobiilirobotit ovat mekanismeja, jotka kykenevät liikkumaan niiden tarkoituksen mukaisessa ympäristössä. Mobiilirobotit voivat liikkua joko maalla, ilmassa tai vedessä. Niissä on yleensä kauko-ohjauksen mahdollisuus ja ne voidaan lähettää tekemään tehtäviä paikkoihin, johon ihmiset eivät voi tai pääse menemään. Keskitytään tässä raportissa kuitenkin teollisuudessa käytettäviin mobiilirobotteihin. Liikkumista varten niille on joko jalat, telaketjut tai renkaat. Niitä on kahdenlaisia: autonomisia tai ei-autonomisia eli ohjattuja. Ohjatut mobiilirobotit ovat yksinkertaisempia, sillä niille opetetaan kulkureitit erikseen ja ne osaavat kulkea vain näitä pitkin. Vaikkakin tämä mahdollistaa paljon, on sillä omat ongelmansa. Esimerkiksi, jos ohjatun mobiilirobotin reitille tulee este, sanotaan vaikka pudonnut pahvilaatikko hyllyltä, mobiilirobotti odottaa esteen takana niin pitkään, kunnes este siirretään pois, ennen kuin se pääsee taas jatkamaan matkaansa. Autonominen mobiilirobotti eroaa siinä ohjatuista, sillä se osaa itsenäisesti liikkua ympäristössään ja reagoida muutoksiin. Aiemman esimerkkitapauksen sattuessa autonomiselle mobiilirobotille, se osaisi kiertää esteen ja pääsisi jatkamaan matkaansa. Molempien tyylien mobiilirobotit on varustettu erilaisilla antureilla, jotta ne osaavat liikkua ympäristössään törmäilemättä. Autonomisessa mobiilirobotissa on antureita enemmän, sillä se vaatii lähtökohtaisesti enemmän tietoa ympäristöstään liikkuaakseen siellä itse. Mobiilirobottien ohjaamiseen voidaan hyödyntää jopa konenäköä. (Brush).

6 OMAT TUTKIMUKSET

Tutkittava esimerkkitalanne jakautuu kolmeen osa-alueeseen, jotka ovat: konenäkökamera, PLC ja robotti. Alla esitetty hahmotelmakuva (KUVA 10) miltä lopullinen tilanne voisi näyttää.



KUVA 10. Hahmotelma esimerkkitalanteesta

Näistä yksittäisistä osa-alueista kerrottiin raportin alussa yleistä teoriaa ja seuraavissa osioissa syvennytään aiheisiin käytännön osalta. Kuvassa näkyy lisäksi HMI (human-machine interface), jota käsitellään PLC:n puolella. HMI-näytöltä loppukäyttäjä näkee laitteiden tilat ja tiettyä dataa kappaleista.

6.1 Konenäkö

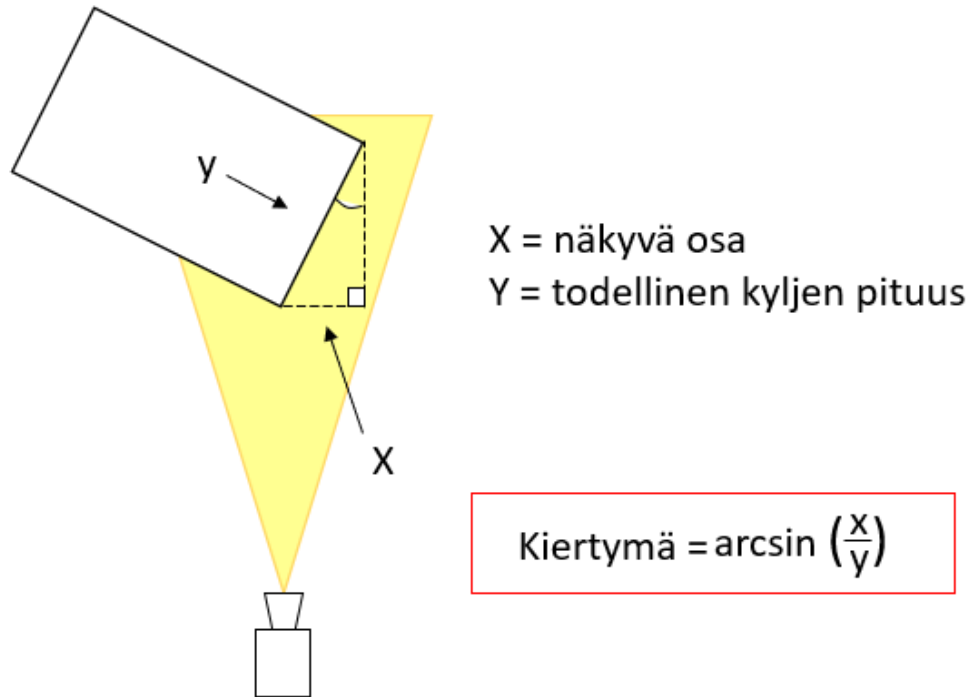
Tutkimuksessa käytettävä konenäkökamera on Omronin FQ2-S35-13 ja sen parametrit asetetaan TouchFinder for PC -tietokoneohjelmalla. Kamera valikoitui tutkimuksessa käytettäväksi sen takia, että yrityksellä oli sen käyttämisestä historiaa ja nyt sen laajempia käyttömahdollisuuksia haluttiin selvittää. FQ2-S35-13 konenäkökamera on vuoden 2017 1,3 megapikselinen 2D-värikamera, jossa on 6mm linssi. Siinä on käsin säädettävä aukonkoko kuvien kirkkauden säätämi-

seen, jonka lisäksi ohjelmallisesti saa säädettyä suljinaikaa sekä valittua esiase-
tettujen HDR tasojen väliltä, jolloin vielä tilannekohtaisemmat asetukset olivat
mahdollisia. Siinä ei ole integroitua valaistusta, joten testeissä käytettiin erilaisia
kotoa löytyviä valaisimia, jotta kameran sensori sai tarpeeksi valoa.

6.1.1 Case 1: Kuvaus kohtisuoraan takaapäin

Kappaleen ollessa oletettavasti metallinen laatikko, tehtiin oletuksiin perustuva
johtopäätös siitä, että parhain kuvakulma olisi suoraan takaapäin. Ajatuksena oli,
että näin ollen laatikosta näkyisi tasainen pohja sekä reunat, jos kappaleen asen-
nossa olisi kiertymää. Valaistukseksi kokeiltiin sivulla sijaitsevaa valonläh-
dettä, jotta kameraan ei tulisi suoraa heijastusta aiheuttaen ylivalotusta. Etänä
suorittaessa tutkimustilojen ollessa suhteellisen rajalliset, käytettiin koekappa-
leena suhteellisen pientä valkoista pakettia, joka hahmotteli todellisen tilanteen
laatikkomaista muotoa. Kokeiluissa huomattiin, että hahmontunnistustyökalu
Shape Search III oli hyvin pätevä. Sille opetettiin referenssikappale, jonka se pyr-
kii etsimään kameran jatkossa ottamista kuvista. Työkalu myös osaa kertoa kap-
paleen keskipisteen sijainnin kuvausalueellaan eli kappaleen x/y-koordinaatit, jo-
ten sillä saatiin kameratunnistuksen helpoin osuus tehtyä.

Seuraavaksi lähdettiin miettimään, miten mahdolliset kiertymät saisi selville. Käy-
tössä oli kuitenkin 2D-kamera, joten kiertymien selvitys ei ollut kameralla itses-
tänselvyys. Pääteltiin, että kiertymät saisi varmasti geometrialla ja trigonometri-
alla laskettua, kun tiedetään tiettyjä pituuksia. Tiedossa oli laatikon ja kameran
välinen etäisyys, laatikon kyljen todellinen pituus sekä kameran näkemä pituus
kyljestä. Kameran näkemä osuus kyljestä saatiin Edge Width -työkalulla, joka
kertoo paikantamansa kahden reunan välisen etäisyyden. Hahmoteltiin tilanne,
joka näkyy alla olevasta kuvasta (KUVA 11). Tilanteesta sai yksinkertaisella sini-
lauseella kiertymäkulman selville. Seuraavaksi tulostettiin mitta-asteikko kapp-
leen alle ja verrattiin todellista kulmaa kameran avulla laskettuun kulmaan. Huo-
mattiin, että laskettu kulma oli noin $\pm 1^\circ$ todellisesta kulmasta, jota pidettiin siedet-
tävänä tarkkuutena. Ongelmana oli, että kulman laskiessa alle 7° , rupesi luotet-
tavuus kärsimään. Tällöin Edge Width ei löytänyt aina tarvittavia reunoja, jolloin
kulmaa ei saatu laskettua.



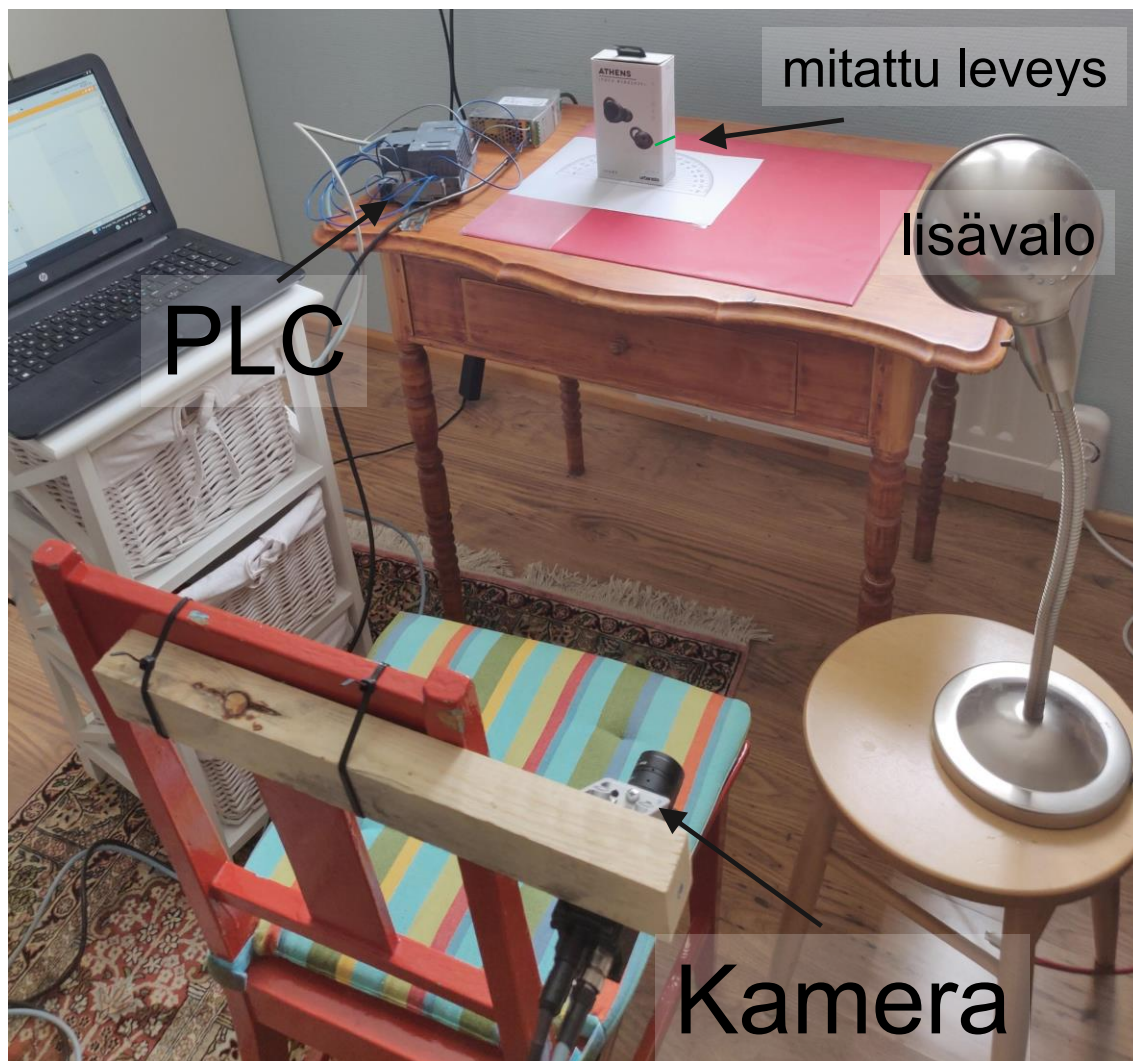
KUVA 11. Kiertymän laskeminen

Kokeilut avasivat silmiä myös muille ongelmille. Tutkimuksissa käytettiin pientä ja helppoa kuvauskohdetta. Tämä pisti miettimään idean toteutusmahdollisuutta isommalle ja yksityiskohdattomalle laatikolle. Ensinnäkin kameran ollessa kohtisuoraan laatikon keskipisteestä, pitää leveässä laatikossa olla suurehko kiertymä, ennen kuin laatikon kylki tulee näkyviin. Eli toisin sanoen pieniä, todennäköisemmin tapahtuvia, kiertymiä ei saa selville. Tästä viisastuneena mietittiin uutta vaihtoehtoa toteutukselle ja keksittiinkin niitä kaksi. Molemmissa kuitenkin on jo lähtökohtaisesti omat haasteensa. Ensimmäinen idea olisi kuvata laatikkoa viistosta, jolloin kappaleen kylki olisi aina näkyvässä kameralla. Toinen idea olisi kuvata laatikkoa kahdesta kuvakulmasta: sen lähestyessä kameraa, jolloin kappale näkyisi kuva-alueen sivussa ja sivukylki näkyisi kunnolla, sekä kohtisuoraa kameran edestä, jolloin saataisiin tarkemmin laatikon x/y-koordinaatit.

6.1.2 Case 2: Kuvaus viistosta

Seuraavaksi kokeiltiin edellä mainittua ensimmäistä ideaa. Kappale siis sijoitettiin kameran kuvausalueeseen nähden hieman sivulle, jolloin kappaleen sivukylki oli

nähtävissä sen ollessa niin sanotussa 0-asennossa. Kokeiluissa pystyttiin käyttämään hyvin pitkälti kaikkia case 1:n metodeja, kuten valaistusta sekä Edge Width -työkalua. Kuitenkin kappaleen sijaitessa kuvausalueen sivussa, tarvittiin kulman laskemisessa vielä enemmän laskutoimituksia. Ensinnäkin kappaleen ollessa 0-asennossaan, on kylki näkyvillä ja kamera laskee jo tästä kappaleelle kiertymän. Siksi otettiin kuva kappaleen ollessa tässä 0-asennossa ja merkattiin saatu kulman arvo talteen. Kutsutaan tätä arvoa niin sanotuksi vakiokulmaksi. Tämä vakiokulman arvo vähennetään jatkossa kaikista lasketuista arvoista, jotta saadaan korjattua kappaleen sijainnista aiheutuva virhe. Näin tekemällä saatiin kappale 0-asennossa näyttämään 0° kiertymää. Sitten mitta-asteikkoa käyttäen kuvattiin kappaletta 10° kiertymässä molempiin suuntiin ja otettiin lasketut arvot ylös. Laskettuja arvoja verraten todelliseen kulman arvoon saatiin muodostettua korjauskerroin, jolla saatiin kompensoitua kappaleen sijainnista johtuma vääristymä laskutulokseen. Korjauskertoimella kerrottiin lopuksi viimeisin arvo, jolloin saatiin laskettu arvo vastaamaan todellista kulman arvoa jo melko tarkasti. Alla kuva (KUVA 12) kokeilun kuvausasetelmasta.

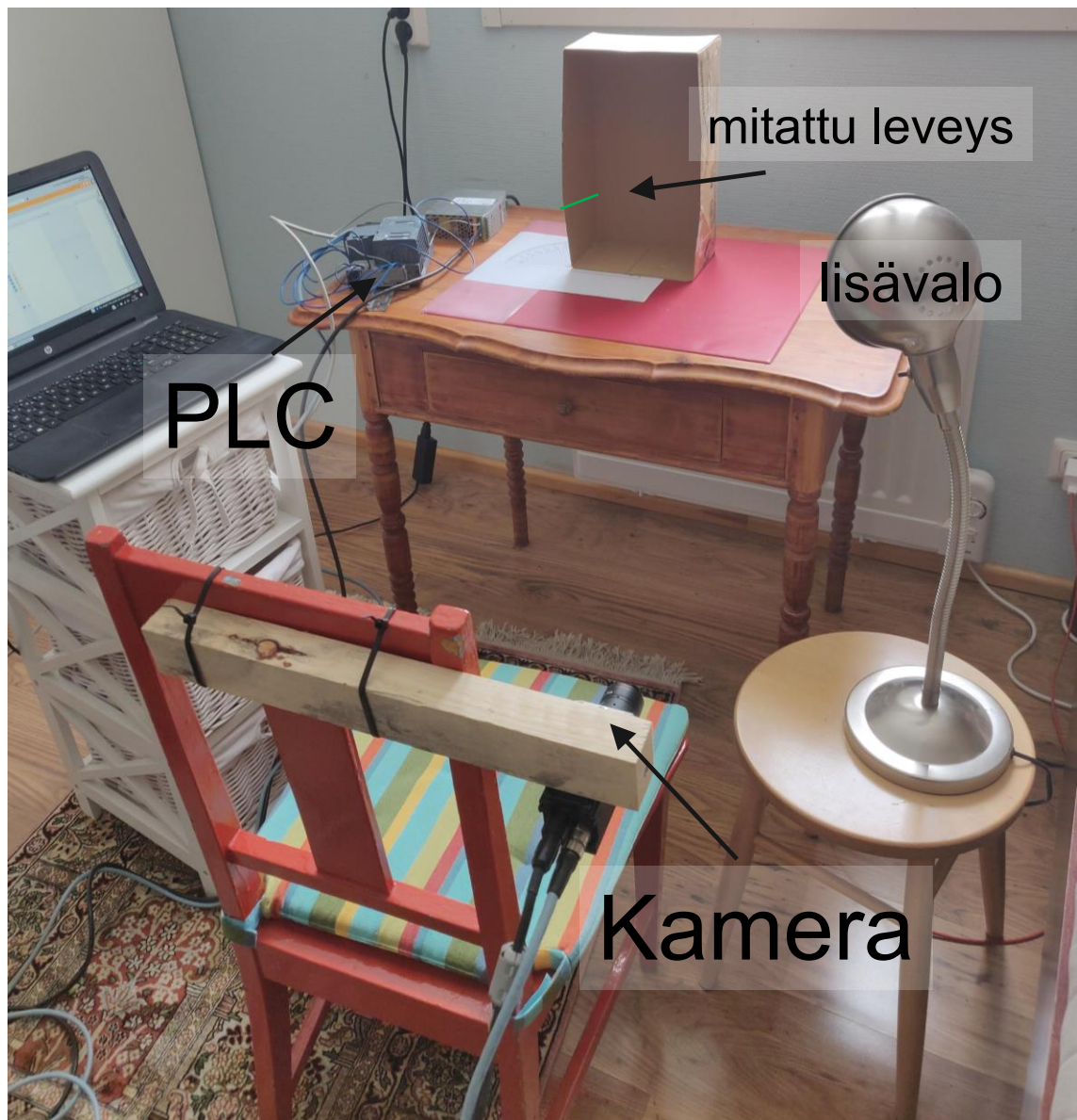


KUVA 12. Kuvaustilanne, pieni kappale

Verrattuna edelliseen kokeiluun, tällä tavalla saatiin kappaleen kiertymä myös 0° ympäriltä selville. Tässä myös käteväksi muodostui se, kun lasketusta kulmasta vähennetään vakiokulman arvo, on toiseen suuntaan tapahtuva kiertymä valmiiksi negatiivinen. Edellisellä metodilla tarvittiin toinen Edge Width -työkalun toisen puolen kyljelle ja tuloksen ollessa positiivinen, kiertymän suunta tarvitsi selvittää toisin keinoin. Kaikin puolin kokeilujen perusteella vaikutti, että viistokuvaaminen on kiertymän selvittämisessä parempi tapa.

6.1.3 Case 3: Kuvaus laatikon sisältä

Vaikkakaan kotiolosuhteista ei löytynyt sopivaa metallista laatikkoa, haluttiin tutkia mahdollisuutta kuvata kappaletta myös sisältäpäin. Testeissä käytettäväksi kappaleeksi valikoitiin pahvilaatikko, joka oli tarkoituksella myös huomattavasti suurempi kuin edellinen koekappale. Kappale asetettiin yhä kuvausalueen reunamille, jotta case 1:n ongelmilta säilyttiin. Tällä kertaa kiertymäkulma laskettaiisiin kappaleen sisäpuolen näkyvästä kyljestä jo edellä mainitulla tavalla muuttamalla kulman korjausarvot oikeiksi. Alla kuva (KUVA 13) kokeilun kuvausasetelmasta.



KUVA 13. Kuvaustilanne, iso kappale

Kokeiluissa oletettiin jo alun alkaen olevan ongelmia sopivan valaistuksen muodostamisessa kappaleen sisäpuolelle. Omissa kokeiluissa ei saatu pahvilaatikon ruskeasta väristä erotettua sisäkulman reunaa luotettavasti kaikista kuvista selville. Kuitenkin kokeiluja jatkaakseni, lyijykynällä korostettiin sisäkulman reunaa, jotta se olisi paremmin kameralla nähtävissä. Tällä tavoin saatiin kokeilut vietyä loppuun asti, jotka osoittivat sen käytännössä yhtä toimivaksi kuin kuvauksen taakapäin. Erona edelliseen tapaukseen on valaistuksen haasteet, jotka voivat korostua yhä entisestään kuvatessa metallista kappaletta. Nämä päästään kuitenkin todentamaan vasta kuvatessa todellista kappaletta.

6.1.4 Case 4: Kaksi kuvaa

Tässä ideana on kahden aikaisemman casen hyvien puolien yhdistäminen yhteen toteutukseen. Eli case 1: x/y-koordinaattien saaminen sekä case 2/3:n kiertymän selvittäminen. Käytetyssä kamerassa on mahdollisuus useammalle sceneelle eli tunnistustilanteen tekemiselle. Täten toteutuksen kulku olisi tehdä scene 1, joka hoitaa kiertymän selvittämisen Edge Width -työkaluun perustuen. Sen jälkeen vaihdetaan kamera sceneen 2, jossa x/y-koordinaattien selvitys on Shape Search III -työkalulla mahdollinen.

Tämä todettiin mahdolliseksi toteutustavaksi kotikokeiluissa, mutta tässä ajateltiin rajoittavana tekijänä olevan lopullisen tilanteen kappaleiden ripustus kuljettimelle sekä kuljettimen liikenopeus. Kappaleiden liian tiheään asettaminen voi aiheuttaa ongelmia kahden eri kuvan ottamiseen. Myös kuljettimen liian nopea liikenopeus voi johtaa siihen, että kamera ei kerkeä pysymään kuljettimen tahdissa, sillä sen täytyy hoitaa kuvan ottaminen, sen käsittely, datan lähetys ja scenen vaihto kahteen kertaan. Tämä pidetään lähtökohtaisesti mahdollisena lähestymistapana, jos tilanne on sille sopiva.

6.2 PLC

Tutkimuksessa käytettiin Siemensin Step 7 -logiikkaan perustuva ET200SP lojiikkaprosessoria ja sitä ohjelmoitiin Tia Portal v16 -ohjelmaversiolla. Käytössä

ei ollut nappeja tai muita ulkoisia inputteja, joten kokeiluvaiheessa ohjelmaa yksinkertaisesti simuloitiin. Sitä varten tietokone yhdistettiin logiikkaan Ethernet-kaapelilla, asetettiin logiikka online-moodiin ja Tia Portalilla hoidettiin ohjelman ajaminen. Tia Portalissa on monitorointitila, josta pääsee näkemään ohjelmakoodin toimintaa ja näkemään kameran logiikalle lähettämät datat. Käytössä oli ohjelman watch table näkymä, jossa pystyi asettamaan muistibittejä päälle tai pois, joilla ohjelmakoodia ajettiin.

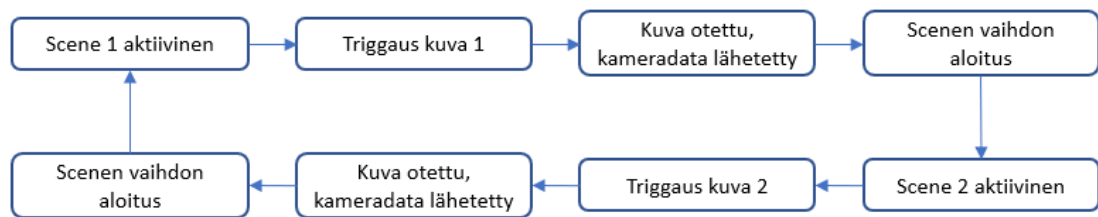
Tutkimusten alkuvaiheessa oli tärkeää saada konenäkökamera ja logiikka keskustelemaan. Tiedonsiirto laitteiden välillä haluttiin hoitaa profinet-väylällä, joka on eräs standardimuotoinen tiedonsiirtotapa teollisuudessa. Jotta kamera saatiin tuotua näkyviin logiikalle, täytyi sen valmistajan sivuilta käydä lataamaan laitekohtainen GSDML-tiedosto. Tällä saadaan konenäkökameran tiedonsiirto määritelmät tuotua logiikalle, jotka eivät automaattisesti Siemensin kirjastosta löydy. Tämän jälkeen konenäkökamera ja logiikka pystyi kommunikoimaan profinet-väylän kautta ja ohjelmakoodin tekeminen pystyttiin aloittamaan.

Ohjelmakoodiin rakennettiin oma FC (function-lohko) triggaukselle eli kuvan ottamiselle sekä datan pyytämiselle. Kamerassa oli mahdollisuus niin sanotulle kättelylle laitteiden välillä. Kättely tarkoittaa sitä, että logiikan pyytäessä kameralta toimintaa, toiminnon jälkeen kamera antaa varmistuksen siitä, että toiminto suoritetaan, eli laitteet niin sanotusti kättelevät keskenään. Tämä ominaisuus ei ole pakollinen toiminnan kannalta, mutta yleisesti järkevä käyttää, koska se vian tapahtuessa osoittaa mistä vikaa voisi lähteä selvittämään. Tämän ominaisuuden käyttäminen monimutkaistaa koodia hieman, mutta tekee siitä samalla myös luotettavamman. Ohjelmakoodi muodostettiin Omronin oman manuaalin mukaan, joka saatiin muutaman matkan varrella tehtävän korjauksen jälkeen toimimaan kiitettävästi.

Kuten aikaisemmin esitetyssä tilateen hahmotelma kuvassa näkyi, saattaa tilanteessa olla useampi kappale kameran ja robotin välillä, joten kameradata tarvitsee saada väliaikaisesti muistiin. Siksi seuraavaksi muodostettiin siirtorekisteri. Se on eräänlainen väliaikaismuisti, jolla kameradata saadaan pidettyä muistissa tietyn ajan ja hyödynnettyä tarvittaessa, jonka jälkeen se häviää. Siirtorekisterin

toimintaperiaate on askeltaa dataa muistipaikalta toiselle komennon tapahtuessa. Muistialue voi olla esimerkiksi 0 – 100, kuten tässä tapauksessa. Kuljetinta seuraamaan asetetaan pulssianturi, joka antaa siirtorekisterille komennon siirtyä muistipaikalta seuraavalle. Täten saadaan kameradata liikkumaan ohjelmallisesti askelittain kuljettimella kulkevan kappaleen rinnalla. Siirtorekisteri toteutettiin muista poiketen ST-ohjelmointikielellä, koska sillä saatiin koodinpätkä tiiviiseen ja suoraviivaiseen muotoon.

Aiemmassa kappaleessa (kappale 6.1.4) esiintyvän case 4:n vuoksi täytyi ohjelmakoodia laajentaa merkittävästi. Ensinnäkin täytyi luoda uusi oma FC scenen vaihdolle. Tässä tarvitsi lähettää kameralle haluttua sceneä vastaava tietty tuplasananmittainen komentokoodi heksadesimaaleina. Lisäksi myös tähän kohtaan muodostettiin kättelytoiminto, jotta saatiin koodiin toimintavarmuutta. Tämän puolen ollessa kunnossa, muodostettiin pääohjelmarakenteelle oma FC, jotta vaadittava sekvenssi tapahtuu oikein ja oikeassa järjestyksessä. Alla esitetty pääohjelman vaiheet (KUVA 14).



KUVA 14. Kahden scenen ohjelmakierto

Ohjelmarakenne on esitetty hyvin yksinkertaistettuna, mutta pääpiirteittäin se vastaa todellista koodia. Vaiheissa 2 ja 6 tapahtuvat triggaukset hoidetaan todellisessa tilanteessa ulkoisesti, kuten kuljettimelle asetetuilla antureilla, mutta kokeiluvaiheissa ne hoidettiin käsin simuloimalla. Ohjelmankesto kuvan triggauksesta scenen vaihtumiseen on hieman alle sekunnin, joten jos kuljettimen nopeus ja antureiden etäisyys mahdollistavat sitä suuremman ajan kuvien ottamisen välille, pitäisi ohjelman olla toimiva sen suhteen.

Tia Portalilla ohjelmoitiin myös järjestelmään suunniteltu HMI-paneeli. Kun käytettävästä paneelistä ei ollut tietoa, valittiin simuloinnissa käytettävä paneeli Sie-

mensin kirjastosta sattumanvaraisesti. Näyttöön suunniteltiin näkymään järjestelmän tilatietojen lisäksi kameradata ja toiminnot käsiajolle. Tilatietoja varten tehtiin kaksi näyttöä, toinen kameraa ja toinen robottia varten. Alla esitetään kuva (KUVA 15), missä näkyy kameraa varten muodostettu näyttö. Kuva on Tia Portalin näkymä, joten siinä ei kaikki graafiset piirteet näy, mutta siitä saa ajatuksen mitä HMI-paneeliin tulisi. Näyttöä pystyttiin simuloimalla kokeilemaan ja päästiin todentamaan triggauksien toimivan sekä tilatietojen ja kameradatan näkyvän näytöllä.



KUVA 15. Näyttö kamerantiedoille, 2:n scenen ohjelma

6.3 Robotti

Tutkimuksessa hyödynnettiin Yaskawa Motoman GP25 nivelrobottia. Fyysisten robotin sijaitessa yrityksen pääkonttorilla, piti ennakkoon tutustuminen hoitaa MotoSim EG-VRC simulointiohjelman välityksellä. Simulointiohjelmalla tutustuminen robottiin ei ole ideaalia, mutta tässä tilanteessa se oli kuitenkin järkevin vaihtoehto. Ohjelmalla kuitenkin pääsi simuloimaan oikean robotin liikkeitä, robotin ohjaamisen perusteita ja Yaskawan ohjain-pendanttia. Näistä oli paljon apua, kun lopulta pääsi fyysistä robottia ohjaamaan paikanpäälle.

MotoSimillä tutustuttiin yksinkertaisen liikeradan muuttamiseen muuttujien avulla, joka hahmotteli tilannetta, jossa konenäkökamera antaisi nämä sijainnin muutokseen johtavat muuttujat. Alussa tutkittiin liikeradan muuttamista käyttäjäkoordinaatistoa muuttamalla. Tämä olisi siinä mielessä ilmeisin ratkaisu, sillä se

osaisi kerralla muuttaa kaikki käyttäjäkoordinaatistoon sidotut paikkapisteet kerralla uusiksi ja pisteiden väliset liikeradat päivittyisivät siinä samalla. Yaskawan robotteihin tämä koordinaatisto kuitenkin luotiin kolmen paikkapisteen mukaisesti, joten muutokset täytyisi saada kaikille kolmelle pisteelle laskettua kolmiulotteisesti, joka osoittautui vaikeaksi.

Seuraava vaihtoehto oli shiftaus-toiminto, jolla saadaan siirrettyä kaikkia paikkapisteitä kyseisen komentoalueen sisällä yhden paikkapisteen suhteen. Yksinkertaisuudessaan tämä tarkoitti yhden paikkapisteen luomista, johon saatiin muuttujien avulla lisättyä siirtymä korkeus- ja leveyssuunnassa. Paikkapisteet luodaan kuuden arvon avulla, joita ovat x-, y- ja z koordinaatit, sekä kiertymät vastaavien akselien ympäri. Tässä tapauksessa kaikki pisteet saatiin liikutettua esimerkiksi 100 millimetriä oikealla ja 50 millimetriä ylöspäin muodostamalla yksi paikkapiste: $P(100, 50, 0, 0, 0, 0)$. Paikkapisteessä siis muutetaan sitä koordinaattiarvoa, johon halutaan muutosta, ja jätetään muut nolliksi.

Seuraavaksi tutustuttiin conveyor tracking eli kuljettimen seuranta -toimintoon. MotoSimmiin tuotiin oman valikon kautta kuljetin, johon tarvitsi asettaa sen liikenopeus ja -suunta. Tämän jälkeen ohjelmaan tarvitsi lisätä komento aloittamaan synkronointi luodun kuljettimen suhteen sekä päivittää liikekäskyt synkronoiduiksi kuljettimen suhteen. Tällä tavoin robotti saatiin suhteuttamaan omat liikkeensä yhdessä kuljettimen liikkeen kanssa. Simulointiohjelmassa tämä toiminto oli yllättävän helppo ottaa käyttöön.

Alla näkyy esimerkki erään ennakkoon tutustuessa tehdyn kokeilun robottiohjelmasta kommentteineen:

```

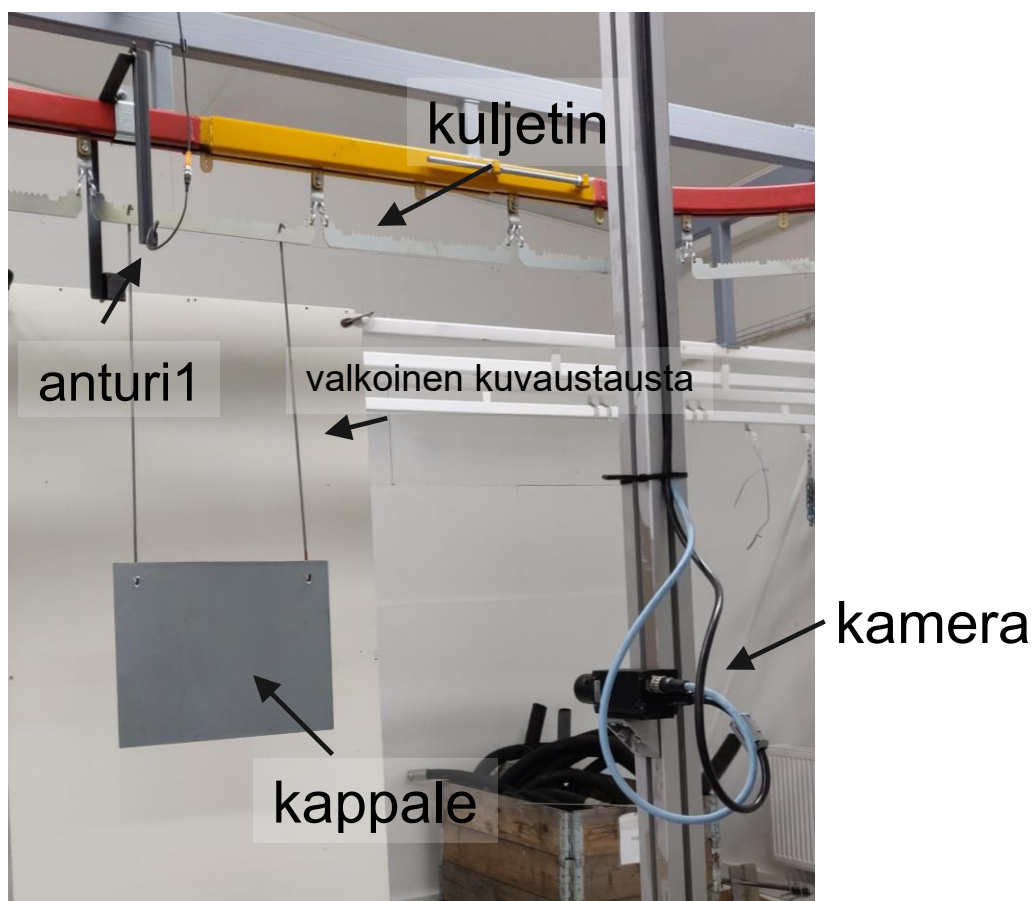
NOP          //Ohjelman alku
MOVJ C00000 VJ=100.00          //Ajo kotiasemaan
MOVJ C00001 VJ=100.00
SYEND CV#(1)          //Varmistetaan kuljettimen seurannan lopetus
SYSTART CV#(1)          //Kuljettimen seuranta kuljettimen 1 suhteen
SFTON P000          //Paikkapisteen shiftaus muuttujan P000 suhteen
SYMOVL C002 V=50.0 CV#(1)
SYMOVL C003 V=50.0 CV#(1)

```

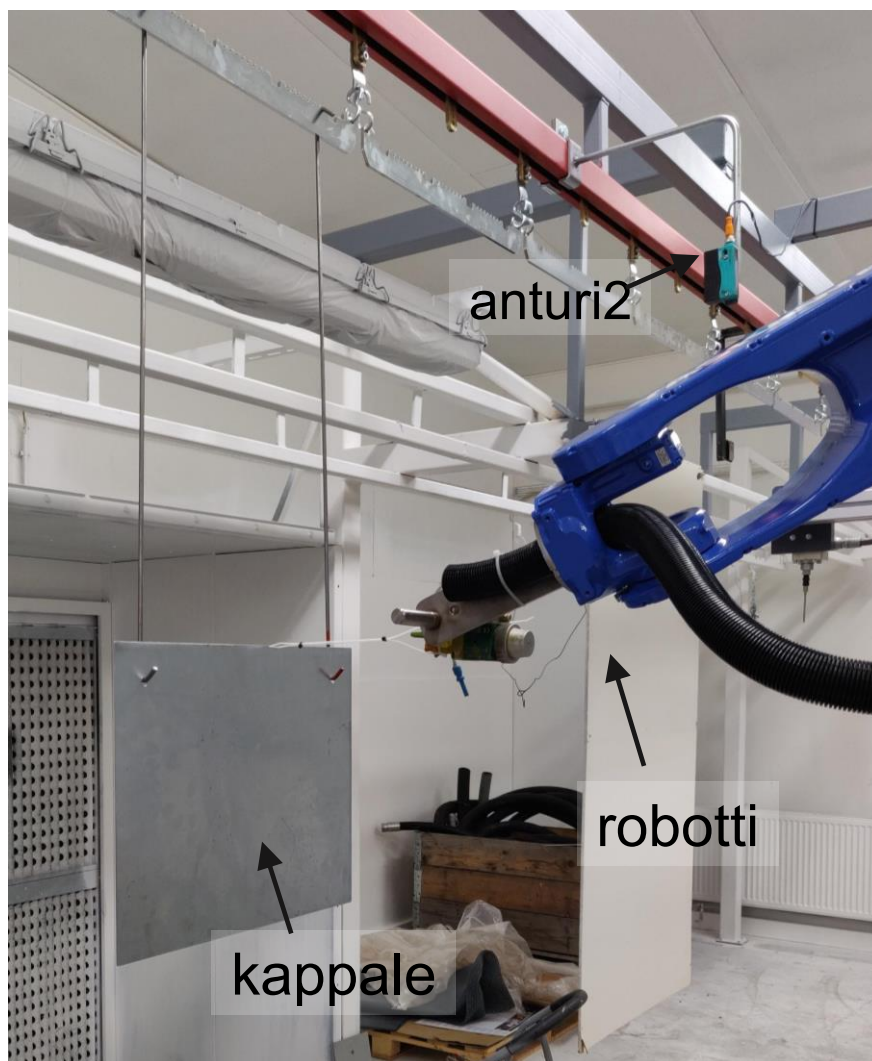
```
SYMOVL C004 V=50.0 CV#(1) //Synkronoidut liikkeet
SYMOVL C005 V=50.0 CV#(1)
SYMOVL C006 V=50.0 CV#(1)
SFTOF //Shiftauksen lopetus
SYEND CV#(1) //Kuljettimen seurannan lopetus
MOVJ C00007 VJ=100.00 //Ajo kotiasemaan
END //Ohjelman loppu
```

7 TOTEUTUS

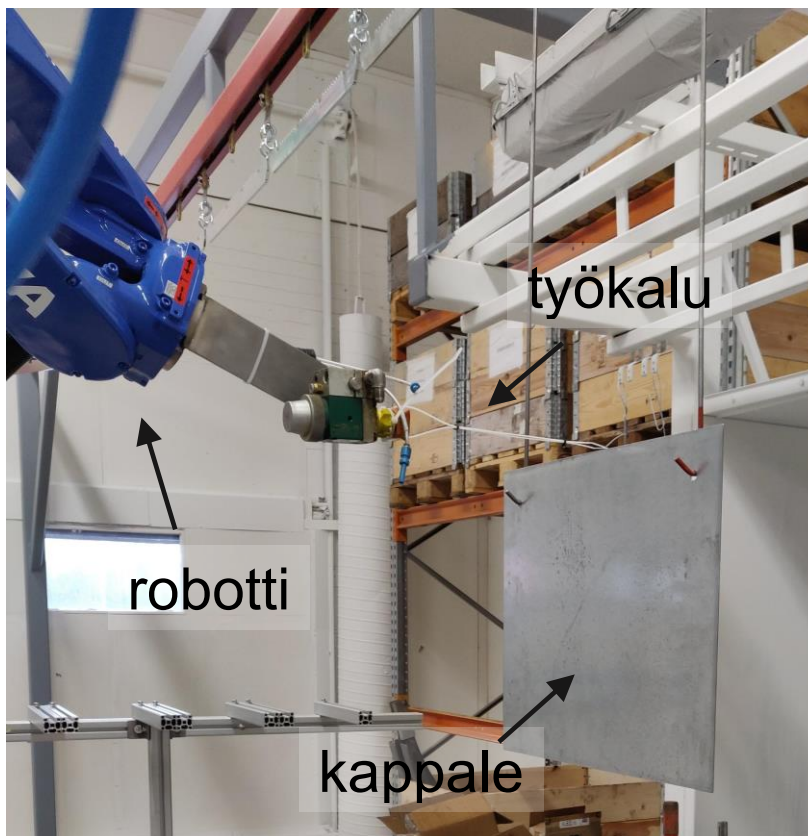
Opinnäytetyöprojektin loppuvaiheilla tehtiin työreissu yrityksen pääkonttorille Savitaipaleelle, jonne kasattiin demolaitteisto lopullisia testauksia varten. Paikan päällä oli valmiina kuljetin ja robotti, kun taas konenäkökamera ja logiikka tuotiin mukana. Konenäkökameralle tehtiin teline sekä sen laukaisuun asennettiin valokenno. Kuljettimeen kiinnitettiin pulssianturi, jolla saatiin kuljettimen liikkumisesta tietoa. Lisäksi kuljettimen yhteydessä oli jo toinen valokenno, jolla saatiin laukaistua kuljettimen seuranta toiminto robotille. Robotin ranteeseen kiinnitettiin maali-ruiskun asentoa hahmotteleva viritelmä, jotta saatiin ohjattua robottia realistisemmin. Alla näkyy kuvia testilaitteistosta (KUVA 16 - KUVA 18).



KUVA 16. Kamera, kuljetin, anturi ja kuvattava kappale



KUVA 17. Robotti, anturi ja kappale

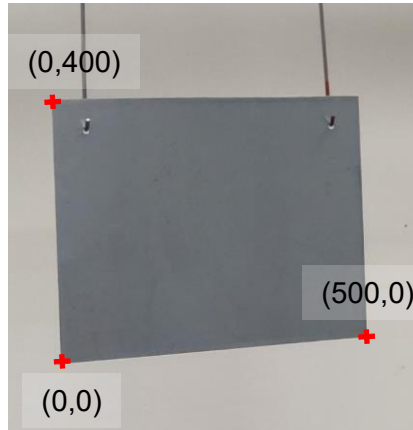


KUVA 18. Robotti ja kappale

Lähtökohtaisesti pyrittiin yksinkertaiseen ratkaisuun, jossa saadaan järjestelmä toimimaan kuitenkin kokonaisuudessaan, ja jos aikaa riittää, kokeiltaisiin toteuttaa vaikeampia ideoita. Tämä tarkoitti aluksi yksinkertaisempaa kuvauskohdetta ja pientä robottiohjelmää. Kuvauskohteeksi valikoitui 50 cm x 40 cm pellinpala ja ohjelmassa robotti kiertää suorakulmaisen pellin ympäri reunoja pitkin yhden kieroksen verran. Tätä varten riitti yhden kuvan ottaminen kohtisuoraa edestäpäin. Lopulliseen kokeiluun ei myöskään saatu ajateltua HMI-paneelia, joten siihen liittyvät toimenpiteet jäivät pois. Paneelin puuttumisella ei kuitenkaan ollut toiminnallisuuden kannalta merkitystä.

Kameralle luotiin asetukset aluksi ottamalla kappaleesta kuva sen niin sanotussa 0-asennossa. Tässä sen pystyi kalibroimaan hyväksi käyttäen pellin tiedettyjä mittoja. Kalibroinnissa valitaan vähintään kolme, tai halutessaan myös useampi, pistettä, jolle syötetään tiedetyt arvot ja näiden perusteella kamera osaa luoda loput kalibrointi parametrit. Kalibrointi tehtiin ottamalla pisteet vasemmasta ylä- ja alakulmasta sekä oikealta alakulmasta. Näin saatiin kameran koordinaatiston

origo kappaleen vasempaan alakulmaan ja muutettua pituuden pikselitiedot millimetritiedoiksi. Alla näkyy hahmotteleva kuva kameran kalibroimisen pisteistä (KUVA 19).



KUVA 19. Kameran kalibrointi kappaleen suhteen

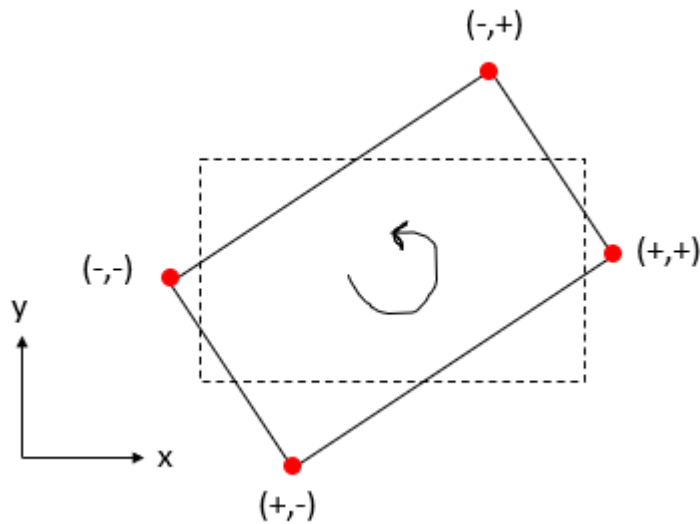
Seuraavaksi lisättiin kuvaan hahmontunnistustyökalu Shape Search III. Se perustuu reunojen löytämiseen, joten kuvattavan kappaleen taakse asetettiin tasavalkoinen tausta häiriöiden välttämiseksi. Työkalu kertoo tunnistamansa kohteen keskipisteen sijainnin x- ja y-koordinaatistolla ja näitä arvoja käytettiin referenssiarvoina siirtymien laskemisessa. Tässä tapauksessa kappaleen keskipiste oli (250, 200). Kappaleen siirtyessä kuva-alueella eri paikkaan, tunnistustyökalu kertoo kappaleen sen hetkisen keskipisteen sijainnin ja näistä miinustetaan referenssiarvot. Näin ollen saadaan siirtymä ja siirtymän suunta molempien koordinaattien suhteen laskettua.

Seuraavaksi siirryttiin robotin pariin ja tehtiin yksinkertainen robottiohjelma, jolla lähdetäisiin kokeilemaan järjestelmän toimivuutta paikallaan olevaan kappaleeseen. Robotin ja PLC:n välisessä tiedonsiirrossa oli kuitenkin hieman hankaluuksia, sillä robotin vastaanottama data jakautui erikoisesti sen IO-alueelle. Lopulta datan löydyttyä robotin puolelta, se kuitenkin erosi lähetetystä datasta. Tarkemmin asiaa tutkiessa huomattiin, että data on oikein, mutta se oli kääntynyt. Data lähetettiin sanoina eli kahtena tavuna ja robotti käänsi näiden kahden tavun järjestyksen sanan sisällä. Tähän löytyi kuitenkin helppo ratkaisu Tia Portalin puolella, jossa oli toiminto sanan kääntämiseen logiikan puolella. Tämän tehtyä lähe-

tetty ja vastaanotettu data vastasivat toisiaan. Lopulta robotille luotiin vielä aliohjelma, jossa luodaan lähetetyn datan avulla paikkamuuttuja, jonka suhteen liikerata saatiin shiftattua. Sitten manuaalisesti testattiin paikallaan olevalla kuljettimella järjestelmää laukaisemalla kamera käsin ja siirtämällä data robotille ja seuraamalla liikeradan tarkkuutta. Järjestelmä toimi hyvin ja se tunnisti kappaleen keskipisteen liikkeessä jopa lähes 20 senttimetriä suuntaan tai toiseen.

Sitten kiinnitettiin huomiota kuljettimen seuranta toimintoon, joka osoittautui yllättävän hankalaksi. Toisin kuin MotoSimmissä, piti kuljettimen asetuksiin asettaa mikrometriä per pulssi arvo liikenopeuden sijaan. Robottiin oli kytketty oma erillinen pulssianturi kuljettimelta sekä ulkoinen valokenno tätä varten. Valokennolla sekä nollattiin aikaisemmat kuljettimen pulssiarvot sekä laukaistiin kuljettimen seuranta toiminto päälle. Valokennolla nollattiin aluksi pulssiarvot ja kuljettimen annettiin liikkua tietty matka ja mitattiin sen fyysinen siirtymä. Sitten katsottiin pulssien määrä tälle siirtymälle. Näiden avulla saatiin mikrometriä per pulssi arvo asetettua. Tämän jälkeen päivitettiin itse robottiohjelmaan liikekäskyt synkronoiduiksi kuljettimen suhteen. Kaiken ollessa oletetusti kunnossa, järjestelmää testattiin, mutta kuljettimen seuranta toiminto aiheutti aina robotille häiriötilan. Tästä alkoi pitkä selvittely siitä, miksi häiriötila syntyy ja miten sen saisi poistumaan ja jopa Yaskawaan otettiin yhteyttä. Lopulta ongelma saatiin poistumaan hieman jopa yllättäen tekemällä kuljettimen kalibrointi ja käyttäjäkoordinaatisto uudestaan. Lopulta saatiin järjestelmä kokonaisuudessaan toimimaan ja robotti muuttamaan liikerataansa kameran perusteella jatkuvassa liikkeessä olevan kappaleen mukaan.

Työreissun ajan käydessä vähiin, kerettiin lopussa tekemään vain pieni kokeilu haastavamman tilanteen kuvauksesta eli kokeiltiin ripustaa kappale eripituisiin koukkuihin, jolloin kappale on kiertynyt syvyys akselin suuntaisesti. Konenäkökameran Shape Search III työkalu kertoo suoraan tämän kiertymän, joten seuraavaksi täytyi miettiä miten tämä kiertymä vaikuttaa tässä tapauksessa kulmapisteiden sijaintiin. Alla on hahmoteltu kuva (KUVA 20) missä nähdään kiertymän vaikutus kulmapisteiden koordinaatteihin.



KUVA 20. Kappaleen kiertymän vaikutus kulmanpisteiden koordinaatteihin

Näiden mietintöjen pohjalta siirryttiin miettimään, kuinka suuri siirtymän arvo on ja miten sen saisi laskettua. Nämä saatiin lopulta yksinkertaisella trigonometrialla laskettu, kun tiedettiin kiertymäkulman suuruus ja kappaleen mitat. Seuraavaksi uudet lasketut arvot tarvitsi siirtää robotille ja muodostaa siellä tarpeeksi muuttujia, jotta neljä paikkapistettä saatiin muodostettua oikeilla arvoillaan. Lopulta nämä tarvitsi lisätä shiftaus-komennoiksi robottiohjelmaan ennen kuin päästiin idean toimivuutta kokeilemaan. Testatessa huomattiin, että robotti tekee oikean muotoisen liikeradan, mutta se ei liiku täysin kappaleen reunoja pitkin. Tämä toistui muutamalla kokeilukerralla ennen kuin huomattiin virheen syy. Virhe oli yksinkertainen, sillä kappaleen kiertymän oletettiin tapahtuvan keskipisteen ympäri. Tilannetta tarkemmin ajatellessa tajuttiin, että kiertymä tapahtuu todellisuudessa pidemmän ripustuskoukun suhteen, eikä kappaleen keskipisteen. Todettiin, että idea oli taustalla oikein, mutta valitettavasti ajan loppumisen takia ei keretty hienosäätää järjestelmää paremmaksi.

8 JATKOKEHITYS

Lähtökohtaisesti tutkittavan ongelman ratkaisu oli haasteellista, mutta ajan kanssa päästiin kokeilemaan eri ideoita ja lopulta osittain jopa toteamaan ne käytännössä. Vaikka aikataulu aiheen parissa tehtävien kokeilujen parissa oli käymässä umpeen, heräsi matkan varrella ideoita järjestelmän kehittämiseen ja seuraavaksi esitellään niitä tarkemmin.

8.1 Järjestelmän laajentaminen

Teoriassa järjestelmää pystyisi laajentamaan kahteen eri suuntaan kokeilujen pohjalta. Ensimmäinen vaihtoehto olisi tehdä järjestelmä siten, että kappaleita voisi olla useampia, mutta suuria asennonmuutoksia ei sallittaisi. Toinen vaihtoehto olisi tehdä järjestelmästä vapaampi kappaleen asennon suhteen, mutta tällöin voitaisiin käyttää vain muutamia selkeästi toisistaan eroavia kappaleita. Molemmat perustuisivat useammalla kameran scenellä tehtävään tulkintaan, jossa aluksi selvitettäisiin yhdellä scenellä mikä kappale on kyseessä ja sen perusteella osataan vaihtaa kameralle toinen tunnistettua kappaletta vastaava scene, joka osaa tehdä tarkemmat asennon tunnistukset. Ensimmäiseen sceneen asetettaisiin korkeuden ja leveyden mittaus, jonka perusteella logiikka tietää mikä kappale on kyseessä ja asettaa kameran vastaavaan sceneen. Seuraavaksi on molemmille vaihtoehdoille omat kappaleet, joissa ideaan pureudutaan tarkemmin.

8.1.1 Useampi kappale

Tämä idea perustuu, kuten edellä jo mainittiin, melko pieniin sallittuihin asennon muutoksiin, erityisesti kiertymien suhteen. Käytössä oleva kamera pystyi mittaamaan hyvinkin luotettavasti, jopa muutaman millin tarkkuudella, kappaleen pituuden tai korkeuden. Järjestelmän toiminta voisi olla täten seuraavanlainen:

Valokenno vaikuttaa kappaleen tullessa kuvausalueelle, josta saadaan logiikalle tieto ja tämä käskee kameraa ottamaan kuvan. Kamera voisi mitata esimerkiksi

kappaleen korkeudeksi 302 millimetriä ja leveydeksi 467 millimetriä. Logiikka vertaisi näitä pituuksia esiasetettuihin rajoihin ja osaisi kertoa, että korkeuden ollessa 290 – 310 millimetriä ja leveyden ollessa 460 – 480 millimetriä, on kyseessä kappale x. Sitten logiikka lähettäisi kameralle tiedon, että vaihtaa scenen numeroksi x:n ja ottaa sitten uuden kuvan. Uuden kuvan käsittelyssä käytetään hahmontunnistustyökalua, Shape Search III:sta, joka selvittää kappaleen keskipisteen sijainnin ja sen kiertymän syvyysakselin mukaan. Näiden kaikkien tietojen pohjalta logiikka kertoisi robotille mikä kappale tulisi maalata ja kuinka paljon sen keskipiste on siirtynyt. Robotti kuittaa tiedot vastaanotetuiksi ja logiikka lähettää lopuksi kameralle käskyn siirtyä alkuperäiseen sceneen, joka mittaa kappaleen korkeutta ja leveyttä.

Vaihtoehtoisesti kappaleen kokoluokka tunnistettaessa ja vaihdettaessa sitä vastaavaan sceneen, voidaan sinne asettaa hahmontunnistukset muutamillekin eri kappaleille, jolloin järjestelmää saisi laajennettua entuudestaan. Kappaleiden täytyisi tosin olla samaa kokoluokkaa, mutta jotenkin erota toisistaan, jotta ne saa tunnistettua tietyksi kappaleeksi. Lisäksi, jos kuvattavia kappaleita on vain muutamia, voidaan järjestelmää ajaa vain yhdessä scenessä, jossa on tarvittavat hahmontunnistuksen näille muutamalle kappaleelle.

Omronin konenäkökamerassa on 32 eri sceneä tehtävissä ja jokaisessa scenessä on mahdollista lähettää 32 eri datatietoa. Näitä lähetettäviä datatietoja on 4 per kappale, joita on tunnistuksen onnistuminen (OK / ei OK), keskipisteen x-koordinaatti, keskipisteen y-koordinaatti ja syvyysuuntaisen akselin mukainen kiertymä. Nämä yhteenlaskettuna tarkoittaa sitä, että tällä menetelmällä olisi teoriassa mahdollista tehdä yli 240 kappaleen tunnistusta, joka mahdollistaa jo hyvinkin suuren maalausjärjestelmän. Se, kuinka näin suuri järjestelmä olisi käytännössä toteutettavissa, onkin eri asia. Yksittäiset tunnistukset olisivat kuitenkin suhteellisen helppo toteuttaa, koska helpot asennontunnistukset eivät vaadi juurikaan vaativaa kalibrointia vaan tiedot saa lähes suoraan kameralta sellaisenaan. Tällaisen järjestelmän asetusten tekeminen olisi työlästä, mutta loppujen lopuksi sen pitäisi olla hyvin suoraviivaista tekemistä.

8.1.2 Kappaleen vapaampi asento

Toinen vaihtoehtoinen laajennustapa mahdollistaisi kappaleiden vapaamman riipustuksen kuljettimelle. Suuremmat asennonmuutokset aiheuttavat omat haasteet erikokoisten kappaleiden kuvaukseen sillä kappaleen kallistuminen vaikuttaa suoraan kameran näkemään kappaleen korkeuteen ja leveyteen. Järjestelmä voisi olla toiminnaltaan samankaltainen kuin edellä mainitussa ideassa muutamia muutoksia lukuun ottamatta. Tässä toteutuksessa kappaleiden täytyisi olla selvästi erikokoisia, mutta kuitenkin samankaltaisia järjestelmän toimiakseen. Kappaleen kallistuessa kameran näkemä etuseinämä pienenee, mutta kylki tulee näkyviin. Kameraan ei saa suoraan asetettua mistä reunasta pituus pitäisi mitata, vaan se mittaa kahden selkeimmin näkemänsä reunan välisen etäisyyden ja alustavasti ne on kappaleen ja taustan väliset reunat. Näin ollen täytyy ottaa huomioon kappaleen kiertymän sekä syvyyden vaikutus kameran näkemään pituuteen. Muuten toiminta voisi olla samankaltainen, jossa kappale aluksi tunnustetaan tiettyjen rajojen mukaan ja vaihdetaan kameran scene sitä vastaavaksi. Tarpeen mukaan asennon laskemiseen voitaisiin käyttää kahta otettua kuvaa, kuten kokeiluissa oli testattu, jos yhdellä kuvalla ei saada kaikkia laskemisessa tarvittavia piirteitä näkyviin.

Jokaiselle kappaleelle kuitenkin tarvitsisi suorittaa kattavat kalibroinnit, jotta kiertymät saataisiin laskettua. Opinnäytetyön aikana ei päästy kokonaisuudessaan todentaman, onko kappaleen täysin vapaa kolmiulotteinen asento lopulta selvittävissä luotettavasti, mutta alkuperäiset kokeilut kuitenkin viittasit, että teoriassa sen pitäisi olla mahdollista. Vaikka asento saataisiin täysin oikein selville, pitäisi sen perusteella kuitenkin laskea robotille huomattavan vaikeat kolmiulotteiset paikkapisteet liikekäskyjen shiftoimista varten. Lopulta tällä tavoin toteutettu järjestelmä ei välttämättä käytännössä ole kannattava tehdä.

8.2 3D-konenäkö ja sen eri tekniikat

Tutkimuksissa käytettävä 2D-konenäkökamera aiheutti omia haasteitaan kolmiulotteisen tilanteen tulkinnessa. Tämän johdosta mietittiin mitä muita ratkaisuja

tai kehitettävää tilanteessa olisi. Eräs vaihtoehto olisi päivittää konenäkökamera 3D-kameraksi ja seuraavaksi käsitelläänkin niiden eri tekniikoita.

3D-konenäkö eroaa 2D:stä nimensä mukaisesti tunnistamalla yhden ulottuvuuden lisää, joka on useimmiten syvyys. Tämän avulla kuvattavan kohteen muoto, paikka ja asento saadaan huomattavasti tarkemmin selville, jonka avulla voidaan tehdä robotin ohjaamisesta helpompaa. 3D-konenäkökamerat pystyvät muodostamaan kolmiulotteisen pistepilven, josta mahdollisesti voisi saada yksittäisiä pisteitä kerrottua robotille. Lisäksi 3D-konenäkö ei ole yleisesti yhtä riippuvainen valaistuksesta tai kappaleen pinnasta kuin 2D-konenäkö. Toisaalta 3D-konenäkö vaatii 2D:tä tehokkaamman prosessorin ja kuvankäsittelyohjelman. Kokonaisuutena 3D-konenäköjärjestelmä koostuu yhdestä tai useammasta kamerasta ja niiden lisälaitteista.

3D-konenäön varmasti tärkein käyttökohde on paikoitussovellutuksissa. Varsinkin robottien ohjaukseen se tuo paljon lisää, mitä 2D-konenäöllä ei saada aikaiseksi. Alla on esitetty neljä yleisintä 3D-konenäön tekniikkaa robottien ohjauksessa. (Zivid).

8.2.1 Stereonäkö

Stereonäkö (stereo vision) perustuu kahden tai useamman kuvan 2D-kuvan sekä tiedettyjen etäisyyksien ja kulmien avulla muodostettuun 3D-mallin tekemiseen. Kuvissa täytyy olla yhteisiä tunnistettuja pisteitä, jotta 3D-malli saadaan luotua. Stereonäön voi tuottaa useallakin eri tavalla, mutta peruseriaate sen taustalla säilyy aina samana.

Varmasti yksinkertaisin tapa stereonäön toteuttamiselle on ostaa valmis stereonäkökamera. Alla on esitetty kuva FLIR Bumblebee sarjan konenäkökamerasta (KUVA 21). Samanlaisia valmiita kameroita löytyy myös muilta valmistajilta ja näiden etuna on helpompi kalibrointi, kuin myöhemmin esitetyissä tavoissa. Sensorit ovat kiinteässä kamerarungossa lähellä toisiaan kuten ihmissilmätkin, joka poistaa osan kalibroinnissa vaadituista mittojen asettamisista. Muita etuja valmiin stereonäkökameran kanssa työskentelyssä on, että sen kuvankäsittelyohjelmassa

on kyseiselle kameralle helpottavia toimintoja. Valmis stereonäkökamera voi myös sopia liikkuvien kappaleiden kuvaamiseen helpommin kuin muut stereo näön toteuttamistavat, mutta asia on kuitenkin aina kamera kohtainen. (FLIR).



KUVA 21. Valmis stereonäkökamera, FLIR Bumblebee (VisionSystems Design)

Toinen vaihtoehtoinen tapa stereonäön toteuttamiselle on kaksi tai useampi erillistä kameraa. Tässä erona edelliseen on, että kamerat ovat irrallaan toisistaan. Täten voidaan kuitenkin sovittaa ne paremmin tarkasteltavaan tilanteeseen ja saada yksityiskohtaisempaa dataa, kuin valmiista stereonäkökamerasta ja sen kiinteästä sensoreiden välisestä etäisyydestä. Tämä toisaalta kasvattaa kalibroimisen vaatimuksia sekä vaatii enemmän taitoa kuvankäsittelyohjelman kanssa työskentelevältä. Tämä tyyli valmiin stereonäkökameran kaltaisesti voi sopia liikkuvien kappaleiden kuvaamiseen kameroiden kiinteän asennon vuoksi, mutta sopevuus on järjestelmä kohtainen.

On myös mahdollista toteuttaa stereonäkö yhdellä kameralla. Tällöin samalla kameralla tarvitsee ottaa useampi kuin yksi kuva kappaleesta. Kameraa täytyy liikuttaa eri kuvaus kohtiin esimerkiksi robotilla, kuljettimella tai muulla vastaavalla mekanismilla. Tärkeää kuitenkin on, että kamera saadaan tarvittavan suurella toistotarkkuudella samaan paikkaan, jossa kalibroinnit on suoritettu, jotta tulokset pysyvät luotettavina. Koska kuvaus tehdään yhdellä kameralla, on kuvien ottamisen välillä aina viive ja sen suuruus riippuu kameran liikuttamisen nopeudesta sekä liikutettavan matkan pituudesta ja viive aiheuttaa haasteita liikkuvien kappaleiden hahmotuksessa.

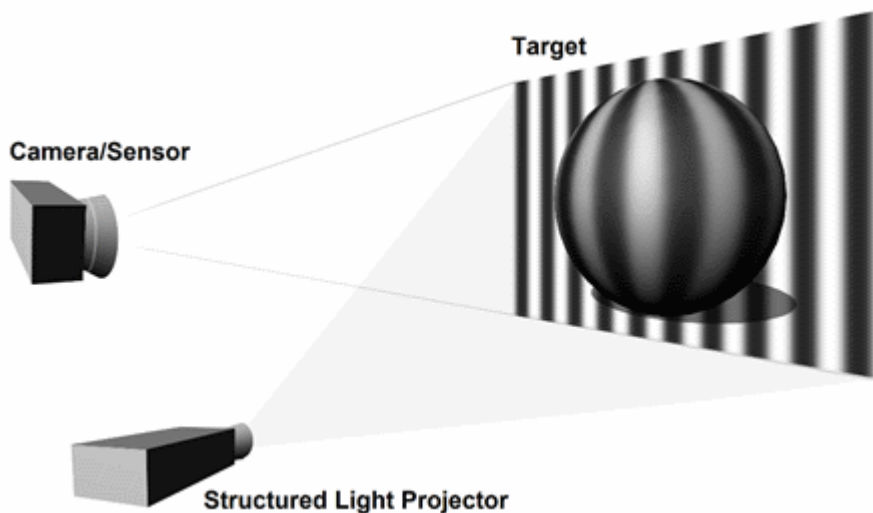
Stereonäön käyttökohteita on kappaleen tunnistus ja paikoitussovellutukset sekä erilaiset mittaukset (FLIR). Kappaleen tunnistus- ja paikkatietoja voidaan käyttää

robotin ohjauksessa esimerkiksi tilanteessa, missä trukkilavallinen sekalaisesti pinottuja laatikoita täytyy purkaa. Robotti tarvitsee tiedot laatikoiden asemista ja asennoista, jotta se osaa purkaa lavan oikeassa järjestyksessä sekä tarttua kappaleisiin oikeasta paikasta. Stereonäöllä voidaan tuottaa tarvittava 3D-tieto suhteellisen edullisesti (Nair).

Stereonäön heikkouksia on tekstuuriittomat kappaleet, joista ei löydy kiintopisteitä, kuten reunoja, kulmia, kirkkaita tai vaaleita kohtia. Lisäksi stereonäkö voi olla altis valaistuksen muutokselle, johtaen epävarmuuksiin tuloksissa. (García).

8.2.2 Struktuurivalo

Struktuurivalo (structured light) on tapa, jossa kappaleeseen kohdistetaan projektorilla tunnettu valokuvio, jonka heijastumasta voidaan tulkita kappaletta (MoviMED). Alla on havainnollistava kuva menetelmästä (KUVA 22). Seuraavaksi esitellään muutamia eri tapoja toteuttaa struktuurivalomenetelmä.

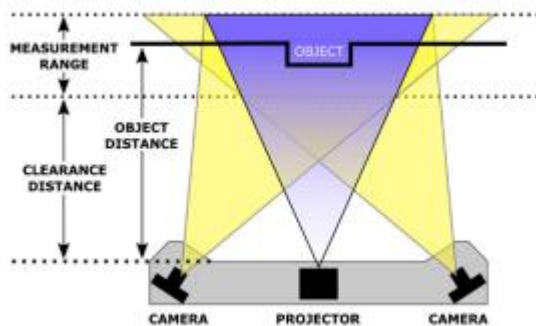


KUVA 22. Struktuurivalomenetelmä

Kuten jo edellä mainittiin, struktuurivalo menetelmä perustuu tunnetun valokuvion kohdistamiseen kappaleeseen. Kappaleeseen voidaan kohdistaa vain yksi valokuvio, jonka perusteella tulkinta tehdään. Valokuvio voi olla kappaleelle kuitenkin täysin uniikki, jolla muoto saadaan parhaiten esiin. Tällä tavalla ei välttämättä saada kaikkein tarkinta 3D-mallia kappaleesta, mutta tätä metodia voi soveltaa

liikkuvaan kappaleeseen. Toinen vaihtoehto on kohdistaa kappaleeseen sekvenssi valokuvioita, jotka eroavat toisistaan esimerkiksi raitojen tiheydessä. Koneköjärjestelmä osaa verrata kaikkia otettuja kuvia ja muodostaa näistä tarkemman 3D-mallin. Mitä enemmän sekvenssissä on erilaisia kuvioita, sitä tarkemmaksi lopputulos saadaan. Useampaa valokuvioita käyttäessä on kuvattava kappale ja kamerat pidettävä täysin paikallaan. (García).

Strukturivalo on tarkka menetelmä ja siitä voi tehdä entistä tarkemman yhdistäen siihen stereonäön, jolloin tilanne näyttää esimerkiksi alla olevan kuvan kaltaiselta (KUVA 23). Eräät konekökameravalmistajat myyvätkin strukturivalokameroita, joissa on yhdistetty kaksi kamerasensoria ja valoprojektori. Tästä saatu lisätarkkuus on tietenkin sovelluskohtaista ja tarjolla on myös kameroita, jossa on vain yksi kamerasensori ja valoprojektori. Valmiissa kameroissa on yleensä jonkin asteista esikalibrointia, mutta käyttäjän täytyy kuitenkin tehdä itse hienosäädöt lopullisen tilanteen mukaisesti. (García).



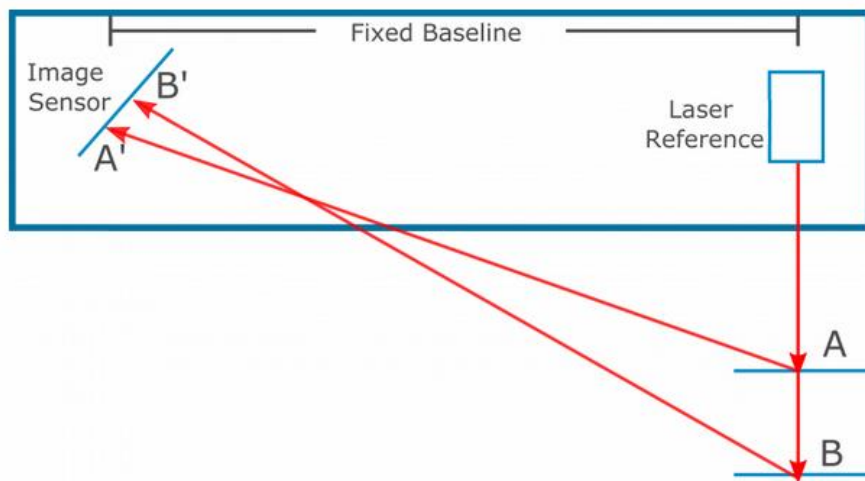
KUVA 23. Strukturivalo ja stereonäkö (García)

Menetelmän haasteina on kameroiden koko ja valaistus. Yleensä käytetty valo tässä menetelmässä on valkoista valoa. Valkoinen valo on kuitenkin altis ympäristön valaisuuden muutokselle johtaen mahdollisiin epävarmuuksiin mittauksissa. Valkoinen valo voi myös tietyn värisissä pinnoissa näkyä huonosti. Strukturivaloantureiden haasteena on ollut niiden suuri koko, jonka vuoksi se on huono asentaa esimerkiksi nivelrobotin käsivarteen. Tekniikan kehityskohteena onkin sensoreiden pienentäminen. (García).

Muodon tunnistus on struktuurivalon tärkeimpiä käyttökohteita sen tarkkuuden vuoksi ja siksi sitä käytetään myös paljon 3D-skannauksessa. Kappaleesta voidaan tehdä tarkka 360° -malli, jos kappaletta kuvataan useasta eri kuvakulmista yhden sekvenssin verran. Eräitä tunnettuja käyttökohteita tälle on myös Microsoft Kinect ja Applen Face ID (Hermany).

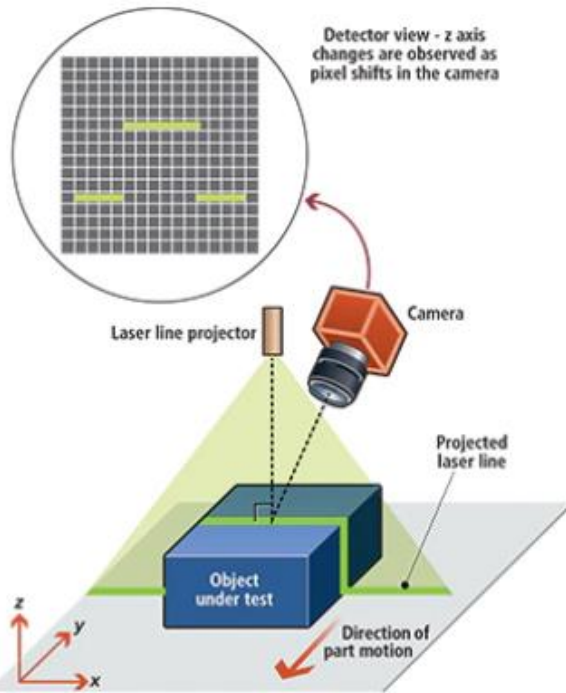
8.2.3 Laser kolmiomittaus

Laser kolmiomittaus (laser triangulation) perustuu laser-osoittimen, sensorin ja kappaleen välille muodostuneen kolmion tulkitsemiseen. Alla näkyy havainnollistava kuva menetelmästä (KUVA 24). Alla oleva kuva havainnollistaa yhden laserin valopisteen heijastumisen, jolla saadaan kappaleesta vain yksittäinen piste selville. Jotta saadaan koko kappaleen muoto selville, täytyy käyttää menetelmää laajemmin. (Hermany).



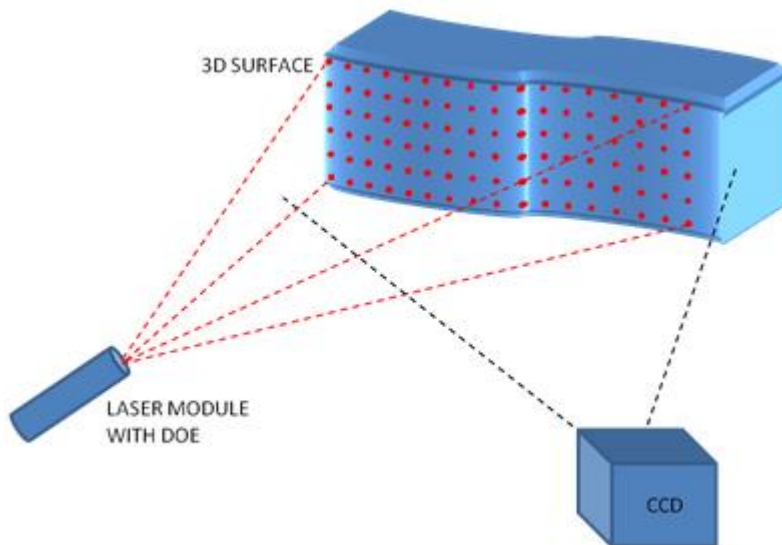
KUVA 24. Laser kolmiomittaus (Hermany)

Yksi tapa on kuvata liikkuvaa kappaletta. Tässä tavassa kappaleen yli lähetetään laser-viiva, kuten alla olevassa kuvassa (KUVA 25), ja viivan heijastuksesta saadaan kamera sensorille tieto kappaleen muodosta. Skannaamalla koko kappaleen ylitse, joko kameraa tai kappaletta liikuttamalla, saadaan muodostettua kappaleesta kokonainen 3D-malli. (VisionSystemsDesign).



KUVA 25. Laser kolmiomittaus, laserviiva -menetelmä (VisionSystemsDesign)

Edellistä edistyneempi tapa on valokuvio -menetelmä, josta näkyy esimerkkikuva alla (KUVA 26). Tässä tapauksessa laservalo hajautetaan useammaksi pisteeksi kappaleeseen, jolloin pystytään koko kappaleen profiilia tulkitsemaan kerralla. Muita käytettyjä valokuvioita on ruudukot, viivat ja ympyrät. Tämä on myös nopeampi menetelmä, sillä kuva voidaan ottaa niin sanotusti kerralla. (Global-lasertech).



KUVA 26. Laser kolmiomittaus, valokuvio menetelmä (Global-lasertech)

Laser kolmiomittaus sopii erityisesti tasaisesti mattapintaisille kappaleille, mutta heijastavat tai läpinäkyvät kappaleet tuottavat sille ongelmia. Heijastavien pintojen kuvaamisessa kohdennettu laser heijastuisi suoraan takaisin lähtöpisteeseen, joten niitä kuvatessa laser täytyy osoittaa pintaa vasten pienessä kulmassa. Läpinäkyvien kappaleiden kanssa ongelma on se, ettei heijastusta saada aikaan. Laservaloa käyttäessä täytyy myös olla tarkka, ettei se pääse heijastumaan ihmissilmään, koska se voi olla haitallista varsinkin käyttäessä tehokkaita lasereita. (MTIInstruments)

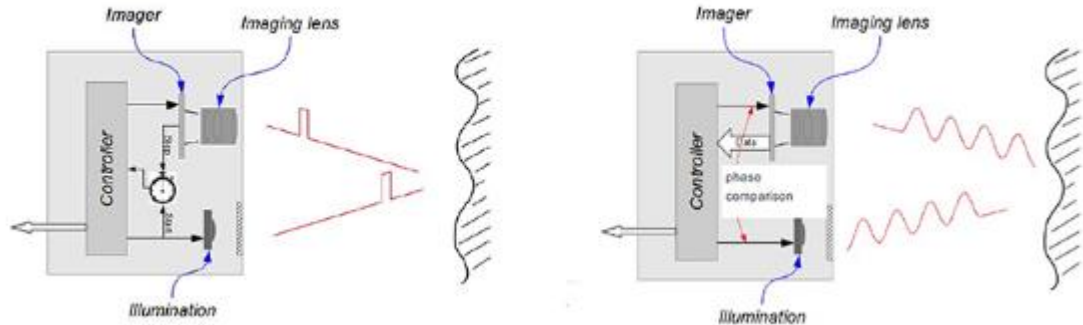
Laser kolmiomittausta hyödynnetään paljon robottien ohjauksessa. Esimerkiksi perinteisestä hitsausrobotista, joka vaatii kappaleilta tarkkoja muuttumattomia asentoja, voidaan tehdä joustavampi. Laser kolmiomittaus menetelmällä voidaan hahmottaa hitsausseaman sijainti sekä sen geometria ja robotti osaa mukautua siihen. Menetelmää voidaan hyödyntää myös perinteisemmissä robotin poimintasovelluksissa, jossa kappaleen asento ja asema tarvitsee selvittää. Lisätarkkuutta halutessa tässäkin menetelmässä voi hyödyntää useampaa kamerasesoria.

8.2.4 Time of flight

Time of flight (ToF) perustuu lähetetyn signaalin kulkuajan perusteella tehtyyn etäisyyden mittaukseen. ToF-kamera muodostuu signaalilähettimestä, esimerkiksi laser-osoittimesta, ja sensorista, joka reagoi laserin aallonpituuteen. Sensori mittaa viiveen valon lähetyshetken ja takaisin saapumishetken välillä ja laskee tämän avulla kameran ja kappaleen välisen etäisyyden. Valon sijasta on mahdollista käyttää signaalina myös esimerkiksi ultraääntä, mutta yleisesti ToF-sensorit perustuvat valoon. (Analog Devices).

On kahden tyyppisiä ToF-menetelmiä. Niin sanotussa suorassa menetelmässä valo lähetetään pulsseina, kun taas epäsuorassa menetelmässä valo lähetetään aaltomaisena, yleensä siniaaltona. Alla olevassa kuvassa (KUVA 27) näkyy eri

menetelmät. Pulssimenetelmässä mitataan yksinkertaisemmin lähtevän ja saapuvan pulssin välinen aikaviive. Aaltomenetelmässä mitataan lähtevän ja saapuvan aallon vaihe-ero. Molemmat menetelmät ovat yleisesti käytettyjä. (Yida).

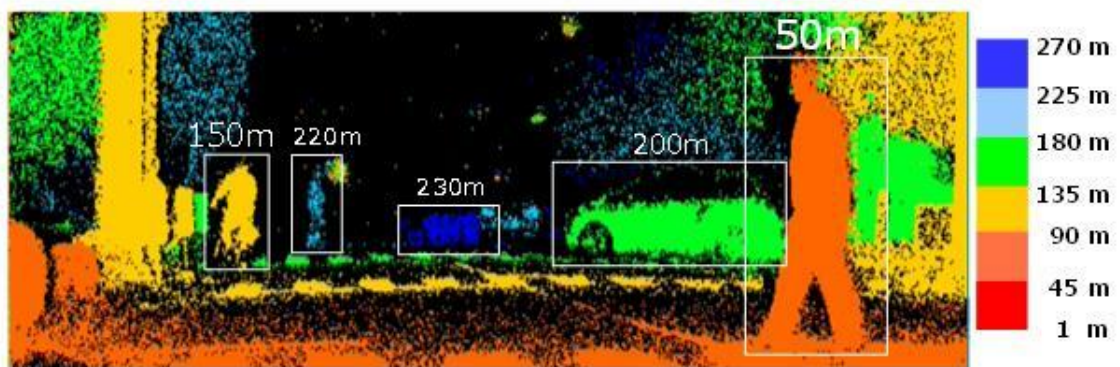


KUVA 27. ToF-menetelmät (Terabee)

ToF-kameran etuja on nopeus, toimintaetäisyys, turvallisuus sekä edullisuus. Kamera pystyy muodostamaan 3D kuvan hyvin nopeasti, jopa yhdellä otoksella. Lisäksi monet kamerat saavuttavat useamman kymmenen kuvan per sekunti nopeuksia. Nopeuden lisäksi ToF toimii myös pitkille etäisyyksille ja eri kokoisille kappaleille. Kamerat ovat myös hyvin joustavia, sillä niiden optiikkaa voi muuttaa halutun näkökentän saamiseksi. Monet ToF-kamerat käyttävät valonlähteenä pienitehoista infrapunalaseria, joten se on myös turvallinen ihmisilmille verrattuna esimerkiksi laser kolmiomittaus menetelmässä käytettyihin lasereihin. Muihin 3D-konenäkömenetelmiin verraten ToF on edullinen. Lisäksi ne saadaan hyvin pieneen tilaan ja niitä onkin kokeiltu joidenkin valmistajien toimesta nykypäivän älypuhelimien kamerajärjestelmissä. (Yida).

ToF -kameroille ongelmia muodostuu heijastavia pintoja kuvatessa. Tällaisissa tilanteissa valon säteet saattavat heijastua ja kimpoilla useiden pintojen välissä ennen kameraan saapumista, mikä vääristää tuloksia. Saattaa myös tulla tilanteita, jossa useammasta valosignaalista hajautuneet pienemmät valonsäteet yhdistyvät ja luovat valetunnistuksia sensorille. Lisäksi ulkona kuvatessa kirkas auringonvalo saattaa valottaa sensorin, jolloin oikeat heijastukset eivät rekisteröidy. (Yida).

ToF-kameroita käytetään erityisesti kappaleiden etäisyyksien selvittämisessä ja siitä johdetuissa sovellutuksissa. Kamera voidaan sijoittaa tilaan, jossa sekä ihmisiä että robotteja työskentelee. Kamera voi antaa käskyn hidastaa robotin liikkeitä ihmisen lähestyessä sitä ja jopa pysäyttää sen taaten ihmisen turvallisuuden. Eleohjaukset ovat myös kasvava kehityskohde ToF-kameroissa. Tämä tarkoittaa, että ihminen voi esimerkiksi kädellään tehdä tiettyjä eleitä ja liikkeitä kameran kuvausalueella ja näillä voidaan ohjata toimilaitetta. Myös 3D-skannauksessa sekä autonomisten kulkuneuvojen tai ajojärjestelmien apuvälineenä voidaan käyttää ToF-kameraa. Panasonic on esimerkiksi kehittänyt autoihin käytettäväksi kameran, joka näkee jopa pimeässä ja hankalissa näköolosuhteissa jopa 250 metrin päähän. Alla on esimerkkikuva kyseisellä kameralla otettuna (KUVA 28). Kuvasta näkee hyvin, kuinka eri etäisyyksillä olevat objektit värjäytyvät eri väreillä luoden selkeän tulkittavan syvyyskuvan. (Panasonic).



KUVA 28. *Panasonicin kehittämä pitkän matkan ToF-kamera (Panasonic)*

8.2.5 3D-konenäön hyödyntäminen

Edellä mainitut tekniikat ovat yleisimpiä 3D-konenäkötekniikoita robottien ohjauksessa. Se miten 3D-konenäköä voisi hyödyntää opinnäytetyössä tutkitussa tilanteessa on erityisesti kappaleen vapaamman asennon tulkinnassa. Kuten kappaleessa 8.1.2 Kappaleen vapaampi asento, käytiin tekniikkaa läpi olemassa olevaa kameraa hyödyntäen, on kyseisen järjestelmän teko haastavaa 2D-kameralla. Kappaleen asennon selvittäminen on haastavaa puhumattakaan robotin ohjaukseen tarvittavien paikkapisteiden laskemisesta. Lisäksi kappaleiden mitat täytyy tuntea, jotta kalibroinnit voidaan tehdä.

Toisaalta kappaleessa 8.1.1 esitetty menetelmä Useampi kappale, soveltuu suu-
relle määrälle kappaleita, mutta siinä vaaditaan joustamattomuutta kappaleiden
asennoilta. Tästä herää kysymys järjestelmän tarpeellisuudesta ja voisiko maa-
lauksen kuitenkin hoitaa toisella tavalla, jos kappaleen asento pysyy käytännössä
samana.

Tästä päästään 3D-konenäön hyödyntämiseen. Teoriassa sitä hyödyntäen voi-
taisiin sallia vapaampi kappaleen asento, mutta myös kuitenkin useampia kuvat-
tavia kappaleita. 3D-kamera osaisi muodostaa kuvatun kappaleen perusteella
kolmiulotteisen pistepilven, josta voitaisiin valita kolme tarkkaa koordinaattia ro-
botin käyttäjäkoordinaatiston tekemiseen. Käyttäjäkoordinaatistoa muokkaa-
malla saadaan koko liikerataa muokattua eikä yksittäisiä shiftauksessa käytettyjä
paikkapisteitä tarvitse laskea. Lopuksi robotille tarvitaan tieto, mikä kappale on
kyseessä, joko kameranlta, työjonosta tai muualta.

Oikean tekniikan valintaan vaikuttaa aina kuvattava kohde, käyttöympäristö ja
käytössä oleva budjetti. Budjetti määrittelee kameran ominaisuudet ja alustavasti
kuvaustulosten tarkkuuden. Tässä tapauksessa kappaletta lähtökohtaisesti ku-
vataan sen roikkuessa ripustuskoukuilla kuljettimesta ja kamera oletettavasti ase-
tetaan kuvaamaan näin ollen lähes vertikaalisesti. Struktuurivalo ja laser kol-
miomittauksessa erityisesti kuvausalue täytyisi sijoittaa erilleen tai eristää, jotta
projektorin valo tai laser ei aiheuta ongelmia. Yksinkertainen stereonäkö on taas
kaikkein riippuvaisin valaistuksesta. Oikean kameran toteutukseen löytääkseen,
kannattaa ottaa yhteyttä eri valmistajiin, kuunnella heidän ehdotuksiaan ja niitä
vertaillen valita paras vaihtoehto.

9 POHDINTA JA YHTEENVETO

Opinnäytetyön aihe oli opintoihin peilaten hyvä. Kaikista osa-alueista oli jo entuudestaan hieman kokemusta ammattikorkeakoulun laboraatioista ja tämän projektin aikana pääsi syventymään niihin täysin uudella tavalla. Matkan varrella oli haasteita, mutta ei mitään ylitse pääsemätöntä. Entuudestaan oletin konenäkö osuuden olevan haasteellisin ja logiikan sekä robotin puolien helpompia. Loppujen lopuksi tilanne olikin juuri päinvastainen.

Konenäkö oli lähtökohtaisesti vaikein osa-alue, koska se oli eniten aiheuttanut hankaluuksia koulun laboraatioissa ja asiaa ei helpottanut itselle täysin uusi Omronin kamera ja sitä ohjaava sovellus. Alussa sen suhteen olikin melkoinen oppimiskäyrä, ja asiaa vaikeutti netistä löytyvän tiedon niukkuus. Aika pian hahmottui, että kamera on lähinnä teollisuuteen tarkoitettu ja se selvisi vähäisen tiedonlöytymisen sekä suoraan sanottuna tönkön TouchFinder for PC ohjelman käyttöliittymän vuoksi. Aloittelijana parhaimmaksi koin valmistajan manuaalin lukemisen ja kokeilujen avulla tehtävän tutustumisen kameran toimintaan. Lopulta käyttö alkoi luonnistua sujuvasti ja ohjelmasta löytyi kaikki tarvittava toiminnallisuuden kannalta, mutta edelleen koen käyttöliittymän olevan heikompi, kuin muilla näkemilläni kameravalmistajilla. Matkan varrella pääsi myös hyvin näkemään teorian käytännössä, kuten erilaisien valaisimien ja valaisutyylien vaikutuksen kuvauksessa sekä hyvän kuvaustaustan vaikutuksen työkalujen toimivuuteen.

Tutkimuksissa käytettävä logiikka oli itselleni suhteellisen tuttu ja myös Tia Portal ohjelmakoodin tekeminen FBD ohjelmointikielellä sujui. Opettelun kohteena kuitenkin oli käytännöllisemmän ohjelmarakenteen tekeminen sekä osittain muut ohjelmointikielet, mitkä ovat myös hyvä oppia tulevaisuutta ajatellen. Ohjelmarakenteella tarkoitan Tia Portalin käytettäviä eri ”rakenne-blockkeja” (eli aliohjelmaa, joihin voi tehdä yksittäisiä ohjelmakokonaisuuksia) ja niiden hyötykäyttämistä ohjelman jaottelussa. Näistä oli koulun puolella teorian puolella mainittu, mutta niitä ei ollut juurikaan käytännössä käytetty. Ohjelmointikielistäkin oli käytetty koulussa ainoastaan FBD:tä, joten muut käytettävät kielet olivat ja ovat edelleen melko vieraita. Logiikan aihealueessa itselle kuitenkin haasteellisimmaksi muodostui juuri tämä hieman erilainen tapa ohjelman muodostamiseen sekä yksinkertaisesti

ohjelman laajuus. Lisäksi laitteiden väliset kättelyt ja siirtorekisterin tekeminen sekä yleisesti laajempi datablockkien käyttö oli itselle uutta asiaa.

Robotin osa-alueesta oli ehkä vähiten käytännön kokemusta, mutta silti koin sen olevan hallussa perustoiminnoiltaan. Aikaisempaa kokemusta oli kertynyt ABB:n sekä UR:n robottien ohjauksesta, mutta Yaskawan robotti aiheuttikin hieman hankaluuksia. Etukäteen robottiin tutustuessa MotoSimillä ei oikeastaan yksinkertaisista toiminnosta tuntunut onnistuvan. Osittain oli varmasti käyttäjässäkin vika, mutta lopulta niistäkin ongelmista päästiin eteenpäin. Loppua kohden myös robotin ohjaus rupesi sujumaan kohtalaisen hyvin ja ongelman kohdatessa osasi jopa hieman hahmottaa ongelman aiheuttavan syyn.

Kaiken kaikkiaan opinnäytetyöprojekti sujui hyvin ja saatiin tuloksia siitä, mitä jo olemassa olevalla kalustolla olisi mahdollista saada aikaan. Henkilökohtaisesti jäi hieman harmittamaan, että aika loppui kesken reissulla yrityksen pääkonttorille ja aivan lopulliseen muotoon järjestelmää ei saatu. Perustoiminnallisuus saatiin kuitenkin todennettua eli se, että kamera tunnistaa kappaleessa pienen leveys- ja korkeussuuntaisen asennonmuutoksen ja robotti osaa muuttaa esiopetettua liikerataansa sen mukaisesti. Projektin aikana opittiin paljon yksittäisistä osa-alueista ja myös siitä, mitä tällaiset suhteellisen yksinkertaisetkin järjestelmät vaativat. Oppimista ajatellen olisi hienoa, jos tällaisia mahdollisuuksia olisi opiskelijoilla mahdollista tehdä useammin, esimerkiksi kesätöiden tai harjoittelujaksojen aikana.

LÄHTEET

- ABB. Product images. Luettu 27.4.2021 https://cdn.productimages.abb.com/9IBA242872_400x400.jpg
- ABB. Product images. Luettu 27.4.2021 https://cdn.productimages.abb.com/9IBA242805_400x400.jpg
- Allied automation. Luettu 27.4.2021 https://www.allied-automation.com/wp-content/uploads/2017/07/rh-ch_robot-e1500509503571.jpg
- Analog Devices. 3D IMAGING WITH ADI TIME OF FLIGHT TECHNOLOGY. Luettu 18.5.2021 <https://www.analog.com/en/applications/technology/3d-time-of-flight.html>
- Anandan T. M. 2020. Industry Trends and Market Potential – What’s Next? Luettu 18.5.2021 https://www.robotics.org/content-detail.cfm/Industrial-Robotics-Industry-Insights/Industry-Trends-and-Market-Potential-What-s-Next/content_id/9391
- Anwekar, V. Vision-guided robotics. Luettu 18.5.2021 <https://blog.isa.org/vision-guided-robotics>
- Basler. Basler ace 2 Series. Luettu 27.4.2021 <https://www.baslerweb.com/en/products/cameras/area-scan-cameras/ace2/#product-line=ace2pro>
- Brush, K. 2019. Mobile robot (mobile robotics). Luettu 18.5.2021 <https://internetofthingsagenda.techtarget.com/definition/mobile-robot-mobile-robotics>
- Carrol, J. 2018. Machine vision system aids in high-speed beer bottle inspection process. Luettu 8.2.2021 <https://www.vision-systems.com/factory/article/16752025/machine-vision-system-aids-in-highspeed-beer-bottle-inspection-process>
- Chromasens. 2017. Manufacturing automation. Line-scan vs. area-scan cameras: What is right for your machine vision application? Luettu 18.5.2021 <https://www.automationmag.com/6989-line-scan-vs-area-scan-cameras-what-is-right-for-your-machine-vision-application/>
- Cognex. Components of machine vision. Luettu 18.5.2021 <https://www.cognex.com/what-is/machine-vision/components>
- Control systems & automations. PLC Programming Basics Part I. Luettu 15.2.2021 <https://www.controlsysteamsandautomation.com/learn/plc/plc-programming-basics-i/>
- Daley, S. 2019. Thanks a lot, Mr. Robot (no, really): Six companies shaping the future of automotive robotics. Luettu 18.5.2021 <https://builtin.com/robotics/automotive-cars-manufacturing-assembly>

Edmundo Optics. Imaging Resource Guide. 10.1. Luettu 27.4.2021 <https://www.ni.com/fi-fi/innovations/white-papers/12/a-practical-guide-to-machine-vision-lighting.html>

Finder. Nipema Oy. Luettu 1.2.2021 <https://www.finder.fi/Konetarvikkeet+teollisuustarvikkeet/Nipema+Oy/Savitaipale/yhteystiedot/159748>

FLIR. 2015. Stereo Vision Introduction and Applications. Luettu 18.5.2021 <https://www.flir.com/support-center/iis/machine-vision/application-note/stereo-vision-introduction-and-applications/>

FLT. AGR Area gantry robot. Luettu 27.4.2021 <https://flt-us.com/products/modular-axis-and-gantry-system/area-gantry-robot/>

García D. F., Pérez, L., Rodríguez I., Rodríguez N., & Usamentiaga R. 2016. Robot Guidance Using Machine Vision Techniques in Industrial Environments: A Comparative Review. Luettu 8.2.2021

Global-lasertech. Machine Vision with Laser Triangulation – Application Guide. Luettu 27.4.2021 <https://www.global-lasertech.co.uk/applications/machine-vision-with-laser-triangulation/>

Hermaty. Guide to Unlocking Your Automation Potential with 3D Scanning. Luettu 18.5.2021 <https://hermary.com/learning/guide-to-unlocking-your-automation-potential-with-3d-scanning/>

Hermaty. How 3D scanners work? Luettu 27.4.2021 <https://hermary.com/learning/principles-of-laser-triangulation/>

HowToRobot. 8 reasons why companies buy robots — and no, it's not just about replacing people. Luettu 17.2.2021 <https://www.howtorobot.com/expert-insight/8-reasons-why-companies-buy-robots-and-no-its-not-just-about-replacing-people>

International federation of robotics (IFR). Industrial robots. Luettu 18.5.2021 <https://ifr.org/industrial-robots>

ISO/TC 299. About ISO/TC 299 Robotics. Luettu 18.5.2021 <https://committee.iso.org/home/tc299>

machinevision.co.uk. Machine vision applications. Luettu 18.5.2021 <https://machinevision.co.uk/machine-vision-uses/applications/>

MoviMED. 2018. What is Structured Light Imaging? Luettu 18.5.2021 <https://www.roboticstomorrow.com/article/2018/04/what-is-structured-light-imaging/11821>

MTI Instruments. Laser triangulation sensors. Luettu 18.5.2021 <https://mtiinstruments.com/technology-principles/laser-triangulation-sensors/>

Nair, D. 2012. A Guide to Stereovision and 3D Imaging. Luettu 18.5.2021 <https://www.techbriefs.com/component/content/article/tb/pub/features/articles/14925>

ni. A practical guide to machine vision lighting. Luettu 27.4.2021 <https://www.ni.com/fi-fi/innovations/white-papers/12/a-practical-guide-to-machine-vision-lighting.html>

Nipema Oy. Nipema Oy kotisivut. Luettu 26.1.2021 <https://nipema.fi/fi>
Omron Microscan systems. Katsottu 1.2.2021 <https://www.microscan.com/en-us/resources/videos/introduction-to-machine-vision-part-1-definition-applications>

Panasonic. Panasonic Develops Long-range TOF Image Sensor. Luettu 27.4.2021 <https://news.panasonic.com/global/press/data/2018/06/en180619-3/en180619-3.html>

Pöysti, C. 2020. DigiSalama. Kobotti – kiva työkaveriko? Luettu 18.5.2021 <https://digisalama.metropolia.fi/kobotti-kiva-tyokaveriko-kokemuksia-kobottien-kanssa-tehdysta-yhteistyosta/>

Researchgate. Figure 3. Luettu 27.4.2021 https://www.researchgate.net/figure/AND-Logic-Gate-and-OR-Logic-Gate_fig3_233237474

Shake M. Ten popular industrial robot applications. Luettu 18.5.2021 <https://www.jabil.com/blog/ten-popular-industrial-robot-applications.html>

Terabee. Time of flight principle. Luettu 27.4.2021 <https://www.terabee.com/time-of-flight-principle/>

VisionSystemsDesign. Choosing a 3D vision system for automated robotics applications. Luettu 27.4.2021 <https://www.vision-systems.com/cameras-accessories/article/16737310/choosing-a-3d-vision-system-for-automated-robotics-applications>

VisionSystemsDesign. Understanding laser-based 3D triangulation methods. Luettu 27.4.2021 <https://www.vision-systems.com/cameras-accessories/article/16738248/understanding-laserbased-3d-triangulation-methods>

Yida. 2020. What is a Time of Flight Sensor and How does a ToF Sensor work? Luettu 18.5.2021 <https://www.seeedstudio.com/blog/2020/01/08/what-is-a-time-of-flight-sensor-and-how-does-a-tof-sensor-work/>

Østergaard E. H. 2016. Why do humans develop robots? Luettu 18.5.2021 <https://www.universal-robots.com/blog/why-do-humans-develop-robots/>