

Kimi Kivinummi

# Konenäköjärjestelmän käyttäminen robotin ohjauksessa kappaleen poiminnassa

Opinnäytetyö

Kevät 2020

SeAMK Tekniikka

Automaatiotekniikka



SEINÄJOEN AMMATTIKORKEAKOULU  
SEINÄJOKI UNIVERSITY OF APPLIED SCIENCES

SEINÄJOEN AMMATTIKORKEAKOULU

## Opinnäytetyön tiivistelmä

Koulutusyksikkö: Tekniikan yksikkö

Tutkinto-ohjelma: Automaatiotekniikka

Suuntautumisvaihtoehto: Koneautomaatio

Tekijä: Kimi Kivinummi

Työn nimi: Konenäköjärjestelmän käyttäminen robotin ohjauksessa kappaleen poiminnassa

Ohjaaja: Juha Hirvonen

Vuosi: 2020

Sivumäärä: 70

Liitteiden lukumäärä: 1

---

Tämän opinnäytetyön tarkoituksena oli suunnitella ja toteuttaa konenäköjärjestelmä, jonka avulla voidaan paikoittaa liukuhinnalta kappaleita ja poimia ne automaattisesti liikkeestä robotilla. Opinnäytetyön kohdeyrityksenä toimi Robia Oy ja työn käytännön osuus toteutettiin Seinäjoen ammattikorkeakoulun robotiikan laboratoriossa.

Konenäköjärjestelmän kehittäminen koostui laitteistojen ja ohjelmistojen valinnasta, järjestelmän asentamisesta ja toiminnan suunnittelusta, sekä toiminnan toteuttamisesta kameran ja robotin ohjelmoinnilla. Konenäköjärjestelmän tarkistusrutiini ohjelmoitiin Cognexin In-Sight Explorer -ohjelmistolla ja robotin ohjelma ohjelmoitiin UR10-robotin graafisella PolyScope-käyttöliittymällä. Opinnäytetyön teoriaosuus sisältää tietoa konenäöstä ja teollisuusroboteista.

Työn tuloksena saatiin toteutettua pääsääntöisesti tavoitteiden mukainen järjestelmä, jossa robotti poimii konenäön avulla liukuhinnalta erilaisia kappaleita luotettavalla toistettavuudella. Työ luovutettiin Robia Oy:lle ja valmiiden suunnitelmien pohjalta Robia Oy:lla on mahdollisuus toteuttaa suunniteltu konenäköjärjestelmä.

Avainsanat: robotti, automaatio, konenäkö, kappaleenkäsittely, Cognex, In-Sight Explorer, Universal Robot

SEINÄJOKI UNIVERSITY OF APPLIED SCIENCES

**Thesis abstract**

Faculty: School of Technology

Degree programme: Automation Technology

Specialisation: Machine Automation

Author: Kimi Kivinummi

Title of thesis: Using a Machine Vision System to Control a Robot in a Part Handling Application

Supervisor: Juha Hirvonen

Year: 2020      Number of pages: 70      Number of appendices: 1

---

The objective of the thesis was to design and build a machine vision system for an application that locates pieces on the conveyor belt and automatically picks them up with a robot. The target company of the thesis was Robia Oy, and the practical part of the work was completed in the robotics laboratory of Seinäjoki University of Applied Sciences.

The development of the machine vision system consisted of the design and selection of the hardware, software and objects, the installation of the system and planning of operation, and the execution of the operation by programming the camera and the robot. The checking routine of the machine vision system was programmed with Cognex In-Sight Explorer software and the program of the robot was programmed with the UR10 robot's graphical PolyScope interface. The theoretical part of the thesis concentrated on machine vision and industrial robots.

As the result of the thesis, a system was implemented that generally met the set objectives, in which the robot uses machine vision to pick up different pieces from the conveyor belt with reliable repeatability. The work was handed over to Robia Oy and based on these designs Robia Oy has a change to implement the machine vision system.

Keywords: robot, automation, machine vision, object handling, Cognex, In-Sight Explorer, Universal Robot

# SISÄLTÖ

Opinnäytetyön tiivistelmä.....	1
Thesis abstract.....	2
SISÄLTÖ .....	3
Kuva-, kuvio- ja taulukkoluettelo.....	5
Käytetyt termit ja lyhenteet .....	8
<b>1 JOHDANTO.....</b>	<b>10</b>
1.1 Työn tausta .....	10
1.2 Työn tavoite .....	10
1.3 Työn rakenne .....	11
1.4 Yritysesittely.....	11
<b>2 KONENÄKÖ .....</b>	<b>12</b>
2.1 Konenäköjärjestelmän toimintaperiaate.....	13
2.2 Kuva-analyysiprosessin vaiheet .....	14
2.3 Kamera ja kenno.....	15
2.3.1 CCD- ja CMOS-kenno.....	16
2.3.2 Kameran valinta .....	16
2.3.3 Optiikka.....	17
2.4 Valaistus .....	18
2.4.1 Valonlähteet .....	18
2.4.2 Valaistusgeometriat.....	23
2.4.3 Valaistustekniikat .....	27
2.5 Kalibrointi .....	28
<b>3 TEOLLISUUSROBOTIT .....</b>	<b>30</b>
3.1 Teollisuusrobotin määritelmä ja rakenne .....	30
3.2 Robottityypit .....	31
3.2.1 Kiertyvänivelinen robotti .....	32
3.2.2 Suorakulmainen robotti .....	32
3.2.3 SCARA-robotti .....	33
3.2.4 Rinnakkaisrakenteinen robotti .....	34
3.2.5 Mobiilirobotit.....	35

3.2.6	Yhteistyörobotit .....	35
3.3	Robotin koordinaatistot .....	36
3.4	Robotin ohjelmointi.....	37
3.4.1	Offline-ohjelmointi .....	38
4	<b>LAITTEIDEN VALINTA JA ESITTELY .....</b>	<b>39</b>
4.1	Universal UR10 -käsivarsirobotti .....	40
4.1.1	Ohjausjärjestelmä ja tiedonsiirto.....	41
4.1.2	Käyttöliittymä ja ohjelmointi .....	42
4.2	Robotin tarttuja.....	43
4.3	Cognex Micro 1403 -älykamera .....	44
4.4	Kameran optiikka .....	45
4.5	Konenäköohjelmisto In-Sight Explorer .....	46
4.6	Latab-rengasvalo .....	47
4.7	Latab-valo-ohjain .....	47
4.8	Liukuhihna .....	48
5	<b>ROBOTTISOLUN SUUNNITTELU JA KÄYTÄNNÖN TOTEUTUS</b>	<b>49</b>
5.1	Konenäköjärjestelmän tavoite .....	49
5.2	Kohdekappaleet .....	49
5.3	Kameran kuvausetäisyyden määrittäminen .....	51
5.4	Valaistus .....	52
5.5	Kameran kalibrointi .....	52
5.6	Kappaleen paikoitus liukuhihnalta .....	55
5.7	Tiedonsiirto kameran ja robotin välillä .....	55
5.8	Konenäköohjelman tarkistusrutiini.....	55
5.9	Robotin liikkeiden ohjelmointi .....	56
5.10	Robotin ohjelma ja järjestelmän toimintasekvenssi .....	56
6	<b>TULOKSET .....</b>	<b>58</b>
6.1	Kappaleen paikoitus ja poiminta.....	58
6.2	Tekstimerkkijonon tunnistus .....	59
7	<b>POHDINTA JA YHTEENVETO .....</b>	<b>61</b>
	<b>LÄHTEET .....</b>	<b>63</b>
	<b>LIITTEET .....</b>	<b>66</b>

## Kuva-, kuvio- ja taulukkoluetelo

Kuva 1. Delta-robotti .....	35
Kuva 2. Kuva toteutetusta järjestelmästä, jossa vasemmalla puolella on UR10-yhteistyörobotti ja oikealla puolella on liukuhihna, jonka päälle on asennettu In-Sight Micro 1403 -älykamera, sekä Latabin rengasvalaisin.....	39
Kuva 3. UR10-yhteistyörobotti. ....	41
Kuva 4. UR10-robotin käsiohjain.....	42
Kuva 5. Robotin tarttuja 2F-85. ....	44
Kuva 6. Kamera ja rengasvalaisin.....	45
Kuva 7. In-Sight Explorerin Easybuilder-ohjelmointitila. ....	46
Kuva 8. In-Sight Explorerin Spreadsheet-ohjelmointitila.....	47
Kuva 9. Latabin valo-ohjain.....	48
Kuva 10. Kohdekappaleet.....	50
Kuva 11. MiniFactoryn 3D-tulostin (vasemmalla) ja RepetierHost-ohjelman tulostuksen esikatselusivu (oikealla). ....	51
Kuva 12. Kalibrointilevy.....	53
Kuva 13. In-Sight Explorerin kalibrointityökalun arvio kalibroinnin onnistumisesta. ....	54
Kuva 14. Kuvasarja kappaleen poiminnasta. ....	58
Kuva 15. Onnistunut tekstimerkkijonon tunnistus .....	59
Kuva 16. Tekstimerkkijonon tunnistus pienemmällä kuvausetäisyydellä. ....	60

Kuvio 1. Tyypillinen konenäköjärjestelmä .....	12
Kuvio 2. Konenäköjärjestelmän rakenne .....	13
Kuvio 3. Yksinkertaisen kuva-analyysiprosessin vaiheet.....	15
Kuvio 4. Neulatyyny- (keskellä) ja tynnyrivääristymä (oikealla). .....	18
Kuvio 5. Halogeenilampun vahvuuksien ja heikkouksien vertailua .....	19
Kuvio 6. Fluoresoivien valonlähteiden vahvuuksien ja heikkouksien vertailua.....	20
Kuvio 7. Led-valaisimien vahvuuksien ja heikkouksien vertailua .....	21
Kuvio 8. Laser-valaistuksen vahvuuksien ja heikkouksien vertailua .....	22
Kuvio 9. Xenon-lampun vahvuuksien ja heikkouksien vertailua .....	23
Kuvio 10. Kohtisuora valaisu rengasvalaisimella.....	24
Kuvio 11. Reiällinen kappale taustavalaisimella. ....	24
Kuvio 12. Rengasmaisen sivuvalo .....	25
Kuvio 13. Diffuusikupolivalaisin.....	26
Kuvio 14. Aksiaalisen diffuusivalaisin.....	27
Kuvio 15. Laserviivalla toteutettu rakenteellinen valaisu .....	28
Kuvio 16. Teollisuusrobotin peruskomponentit.....	30
Kuvio 17. Yleisimmät robottityypit ja niiden rakenne-esimerkkejä .....	31
Kuvio 18. Kiertyvänivelinen teollisuusrobotti .....	32
Kuvio 19. Suorakulmainen teollisuusrobotti .....	33
Kuvio 20. Scara-teollisuusrobotti.....	34
Kuvio 21. Teollisuusrobotin koordinaatioita .....	37

Taulukko 1. Optimaalisen kuvausetäisyyden laskevan laskurin tulos .....	52
---	----



## Käytetyt termit ja lyhenteet

<b>ASCII</b>	(American Standard Code for Information Interchange) on 7-bittinen eli 128 merkkipaikan laajuinen tietokoneiden merkistö, joka sisältää ensisijaisesti amerikanenglannissa tarvittavat kirjaimet, numerot, väli- ja erikoismerkkejä sekä eräitä ohjauskoodeja.
<b>CCD</b>	Charge-Coupled Device.
<b>CMOS</b>	Complementary Metal Oxide Semiconductor.
<b>Fotodiodi</b>	Valolle herkkä diodi, joka muuntaa valon sähköiseksi signaaliksi.
<b>I/O</b>	Input/Output.
<b>Koordinaatisto</b>	Geometrinen järjestelmä alueen kuvaamiseen ja sen mitasuhteiden, sijaintien tai muiden sellaisten ilmoittamiseen.
<b>Laser</b>	Light amplification by stimulated emission of radiation.
<b>LED</b>	Light-Emitting Diode.
<b>Manipulaattori</b>	Yksinkertaista ja toistuvaa rutiinia suorittava laite, esimerkiksi siirtää kappaletta tai laitetta. Mikäli rutiinit ovat monimutkaisia ja niitä voidaan muuttaa, puhutaan robotista.
<b>Polttoväli</b>	Optisen systeemin polttoväli on mitta sille, kuinka paljon se kokoaa tai hajottaa valoa. Systeemin optinen taittokyky on sitä suurempi mitä pienempi on sen polttoväli.
<b>Resoluutio</b>	Eli kuvatarkkuus on termi, jolla kuvankäsittelyssä ja tietotekniikassa kuvataan muun muassa bittikarttagrafiikana esitetyn kuvan erotuskykyä tai pikselien määrää.
<b>SCARA</b>	Selective Compliant Assembly Robot Arm.

<b>TCP</b>	Työkalun keskipiste (Tool Center Point) määrittää työkalun tarkan työpisteen.
<b>TCP/IP</b>	(Transmission Control Protocol / Internet Protocol) on useaan Internet-liikennöinnissä käytettävän tietoliikenne-protokollan yhdistelmä eli pino.
<b>Valotusaika</b>	Tarkoittaa valokuvauksessa aikaa, jonka kameran suljin on auki valotuksen aikana.

# 1 JOHDANTO

## 1.1 Työn tausta

Opinnäytetyön kohdeyrityksenä toimi Robia Oy. Robia Oy on Seinäjoella sijaitseva sähkö- ja automaatioalan yritys, joka toimii teollisuusrobottien maahantuojana sekä näiden ympärille rakennettujen tuotantoautomaatiojärjestelmien kokonaistoimittajana.

Kohdeyrityksellä oli suunnitteilla projekti, jossa yhdessä osassa isompaa automaatiojärjestelmää tarvittaisiin mahdollisesti konenäköä kappaleen paikoittamiseen. Tämän vuoksi syntyi tarve tutkia konenäköjärjestelmän käyttämistä yhdessä robotin kanssa ja etukäteen testata olisiko järjestelmä mahdollista toteuttaa. Haluttu järjestelmä koostuisi robotista, liukuhihnasta ja erillisestä konenäkökamerasta, jota ei olisi liitettyä robotin käsivarteeseen. Koululta löytyi kaikki tarvittavat laitteet halutun järjestelmän testaamiseen, joten tästä saatiin ajatus kokeilla järjestelmän toteuttamista ja toimintaa koulun tiloissa. Järjestelmän testaaminen etukäteen koululla antaisi hyvän kuvan järjestelmän toiminnasta ja sen mahdollisista haasteista.

## 1.2 Työn tavoite

Työn tavoitteena oli suunnitella ja toteuttaa konenäköjärjestelmä, jonka avulla voidaan paikoittaa liukuhihnalta poimittavia kappaleita. Kappaleiden poiminta sekä kappaleen kuvaus tapahtuisi liikkeestä ja kappaleet poimittaisiin robotilla. Tavoitteena olisi myös tehdä kuva-analyysi kohdeyrityksen suunnitteilla olevan projektin kohdekappaleesta. Kuva-analyysissä selvitettäisiin, voitaisiinko kohdekappaleen piirteet tunnistaa riittävän hyvin, jotta se voitaisiin poimia automaattisesti. Kappaleesta tulisi myös tunnistaa tekstimerkkijono.

### **1.3 Työn rakenne**

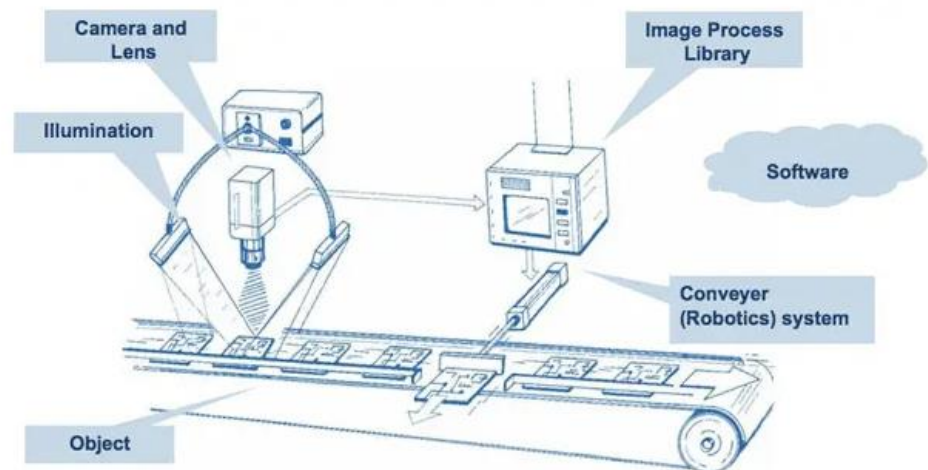
Työ alkaa johdannosta, jossa käsitellään työn tausta, tavoitteet ja rakenne sekä esitellään työn kohdeyritys. Tämän jälkeen siirrytään teoriaosuuteen, jossa kerrotaan yleisesti konenäöstä ja teollisuusroboteista. Teoriaosuuden jälkeen esitellään järjestelmän laitteisto ja käytännön toteutus, jossa perehdytään järjestelmän suunnitteluun ja sen toteutuksen eri vaiheisiin. Seuraavaksi käydään läpi työn tulokset sekä yhteenveto ja pohdinta. Tulokset-osiossa käydään läpi, mitä työssä saavutettiin, ja päästiinkö haluttuihin tavoitteeseen. Yhteenvedossa ja pohdinnassa taas käydään lyhyesti läpi työn eteneminen, tulokset, haasteet ja tulevaisuuden kehityskohteet.

### **1.4 Yritysesittely**

Työn kohdeyrityksenä toimi Robia Oy, joka on Seinäjoella sijaitseva sähkö- ja automaatioalan yritys, joka työllistää kuusi ihmistä. Robia Oy on robotisoitujen automaatiojärjestelmien kokonaistoimittaja, joka toimii Hyundai-teollisuusrobottien ja Kas-sow-yhteistyörobottien maahantuoja. Robia Oy:lla on toimipiste Seinäjoella ja yrityksellä on asiakkaita ympäri Suomea. (Robia Oy 2020.)

## 2 KONENÄKÖ

Automated Imaging Association (AIA) määrittelee konenäön koostuvan sovellutuksista, joissa ohjelmiston ja laitteiston yhdistelmällä saadaan otettua kuva ja prosessoitua siitä toimintakäsky ohjattavalle laitteelle. Kuvio 1 havainnollistaa tyypillisen konenäköjärjestelmän rakennetta. Konenäköjärjestelmä koostuu useasta eri komponentista, kuten kamerasta, objektiivista, valaistuksesta, kuvankäsittelystä ja tiedonsiirrosta. Kameralla ja objektiivilla muodostetaan kuva halutusta kohteesta ja valaistuksella saadaan korostettua kohdekappaleen halutut piirteet. Perinteisessä konenäköjärjestelmässä käytetään tietokonetta kuvan käsittelyyn ja analysointiin, mutta tämä voidaan myös suorittaa älykameralla, joka sisältää kaiken tarvittavan elektroniikan kuvan käsittelyyn ja sen analysointiin, kuten prosessorin ja muistit jne. (Cognex 2016.)

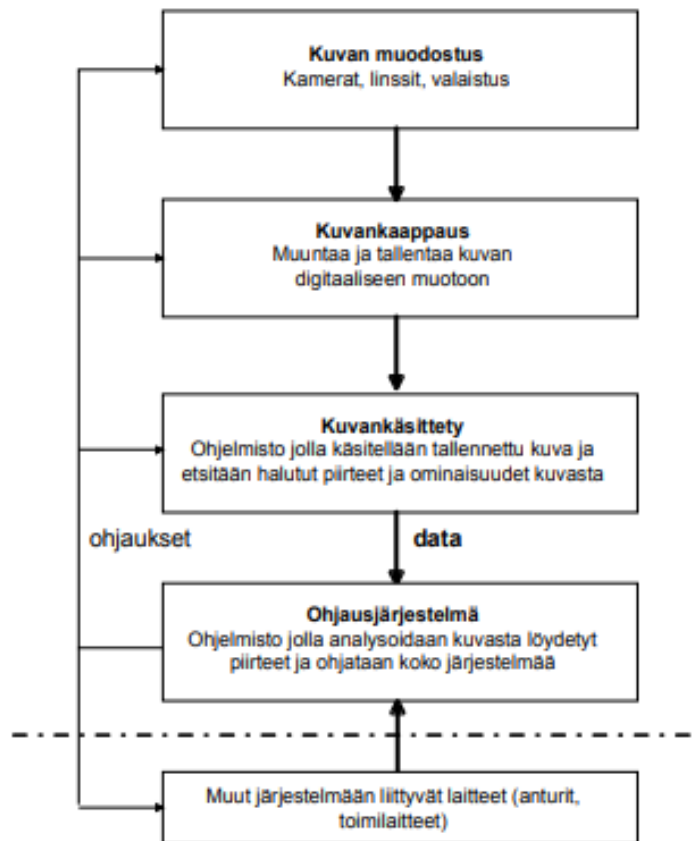


Kuvio 1. Tyypillinen konenäköjärjestelmä (Digi-Key 2012).

Konenäköjärjestelmiä käytetään laajasti eri teollisuuden aloilla. Tyypillisesti konenäköjärjestelmiä käytetään erilaisissa prosessien automatisoinnissa, jotka olisivat ihmisoperaattorille liian aikaa vieviä, vaikeita, vaarallisia tai suoranaisesti mahdottomia. Tällaisia prosesseja ovat esimerkiksi laaduntarkkailu, liikkeenohjaus, lajittelu ja erilaiset mittaukset. (Halinen 2007, 1-2.)

## 2.1 Konenäköjärjestelmän toimintaperiaate

Konenäköjärjestelmän toiminnan voi jakaa karkeasti neljään eri osaan (kuvio 2). Konenäköjärjestelmän toiminta koostuu kuvanmuodostuksesta, kuvankaappauksesta, kuvankäsittelystä ja ohjausjärjestelmästä. Kuvan muodostuksessa kamera ottaa kuvan halutusta kohteesta, kuvankaappauksessa kuva muunnetaan muotoon, jossa sitä voidaan käsitellä. Kuvankäsittelyssä kuvasta erotellaan tarvittava ja tarpeeton tieto toisistaan, ja ohjausjärjestelmä tekee ohjaus- tai korjaustoimenpiteitä konenäköjärjestelmän muihin osa-alueisiin kuvankäsittelystä saadun informaation perusteella. Järjestelmään usein myös liittyy automaatiojärjestelmän muita laitteita ja antureita, mutta ne eivät ole varsinaisesti osa konenäköjärjestelmää. (Halinen 2007, 1-12.)



Kuvio 2. Konenäköjärjestelmän rakenne (Halinen 2007, 3).

## 2.2 Kuva-analyysiprosessin vaiheet

Tyypillisesti konenäköjärjestelmän yksinkertainen kuva-analyysiprosessi koostuu kuudesta eri vaiheesta: esikäsittely, segmentointi, sisällön kuvauksen muodostus, sovitus ja tunnistus. Tässä yhteydessä on oletettu, että analysoitava näkymä on kaksiulotteinen, jolloin syvyystietoa ei tarvita (Pietikäinen & Silven, [viitattu 16.01.2020], 4.)

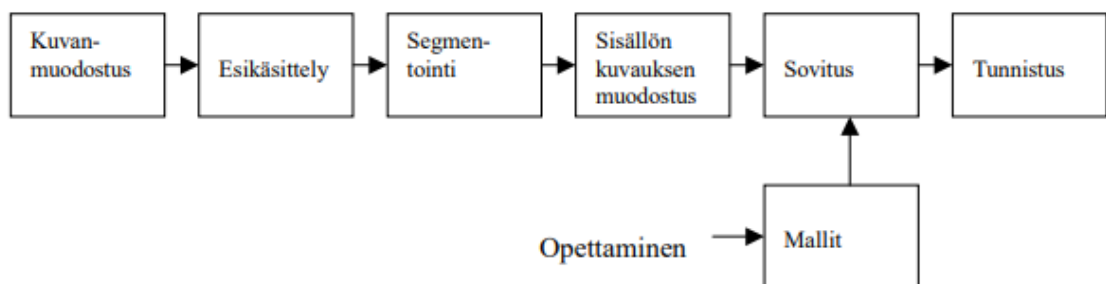
Kuva-analyysiprosessi alkaa kuvan muodostamisesta, jossa kuva taltioidaan kuvälähteellä kuten kameralla. Kuvausjärjestelyjen ja valaistuksen hyvin suunnittelemisen helpottaa kuvan analysointia. (Pietikäinen & Silven, [viitattu 16.01.2020], 5.)

Kuvan muodostamisen jälkeen kuva esikäsitellään, jolloin kuvaa muokataan digitaalisen kuvankäsittelyn menetelmillä analyysin kannalta helpommin käsiteltäväm-  
pään muotoon. Kuvasta esimerkiksi korostetaan halutut piirteet esille, suodatetaan sävyvaihteluja kuten kohinaa pois ja normalisoidaan kuva, jotteivat ympäristötekijät kuten valaistus vaikuta liikaa lopputulokseen. (Pietikäinen & Silven, [viitattu 16.01.2020], 5.)

Esikäsitteilyn jälkeen kuva segmentoidaan, jossa tavoitteena on erottaa kohteet ja kohteiden osat toisistaan ja taustastaan. Segmentointi on kuva-analyysin tärkeimpiä vaiheita ja huono segmentointi saattaa vaikeuttaa analyysiä huomattavasti tai tehdä siitä jopa mahdottoman. Segmentoinnissa käytetään yleisesti kahta eri periaatetta, jotka ovat aluepohjaiset menetelmät ja reunailmaisuus. Aluepohjaisissa menetelmissä kuva jaetaan väreiltään, harmaasävyiltään tai muulla vastaavalla ominaisuudella homogeenisiin alueisiin, kun taas reunanilmaisussa kuvasta ilmaistaan jyrkkiä sävynmuutoskohtia eli alueiden reunoja. (Pietikäinen & Silven, [viitattu 16.01.2020], 5-6.)

Tämän jälkeen lasketaan segmentoitujen alueiden, reunojen tai muiden vastaavien piirteiden ominaisuuksia, joiden perusteella halutut kohteet voidaan erottaa muista kohteista. Tällaisia ominaisuuksia ovat mm. väri, muoto ja alueiden pintarakennetta kuvaava tekstuuri. Tunnistettava kohde saattaa muodostua monesta eri segmentistä, jolloin kohteen rakenteen kuvauksessa tarvitaan alueiden ominaisuuksien lisäksi myös alueiden keskinäisiä relaatioita. Rakenteen kuvaukseen käytetään usein

semanttisia verkkoja, joiden solmut kuvaavat alueita ja linkit niiden keskinäisiä riippuvuuksia. Tarvittavien kuvauksien muodostamisen jälkeen niitä verrataan systeemille etukäteen opetettujen prototyyppikohteiden malleihin ja pyritään tunnistamaan haluttuja piirteitä sekä tunnistamaan poikkeamia. Tätä ei kuitenkaan välttämättä aina tarvitse tehdä, vaan yksinkertaisimmat kohteet pystytään tunnistamaan pelkästään segmentoinnin tai joidenkin siitä laskettujen parametrien perusteella. (Pietikäinen & Silven, [viitattu 16.01.2020], 6.) Nämä eri vaiheet havainnollistetaan kuviossa 3.



Kuvio 3. Yksinkertaisen kuva-analyysiprosessin vaiheet (Pietikäinen & Silven, [viitattu 16.01.2020], 5).

### 2.3 Kamera ja kenno

Kameran on konenäköjärjestelmän tärkein komponentti ja sen tehtävänä on yksinkertaisuudessaan kuvata tarkasteltava kohde. Kohteesta heijastuva valo ohjautuu kameran optiikan kautta valoherkälle kennolle, joka varautuu sähköisesti siihen tulevan valon kirkkauden perusteella. Kenno on kameran tärkein osa, ja sitä käytetään kuvan muodostamiseen. Kenno voi olla joko viiva- tai matriisikenno, riippuen kennon valoherkkien tunnistimien järjestelytavasta. Matriisikennolla kaksiulotteinen kuva voidaan muodostaa yhdellä valotuksella, mutta viivakenno tarvitsee kaksiulotteisen kuvan muodostamiseen kameran ja kohteen välisen suhteellisen liikkeen. Nykyaikaisissa digitaalikameroissa käytetään tyypillisesti kahta eri kennotekniikkaa, jotka ovat CCD (Complementary Metal-Oxide-Semiconductor-sensor) ja CMOS (Charged-Coupled Device). (Hornberg 2006, 363-364, 373-374.)



### 2.3.1 CCD- ja CMOS-kenno

CMOS- ja CCD-kennot eroavat toimisistaan niiden tavassa käsitellä pikselivarauksia. Valotusaikana CCD-kennon fotodiodien pikselivaraukset siirtyvät puolijohde-kondensaattoreita hyödyntävien siirtoporttien kautta sarjaliikennelukijarekistereihin, jonka jälkeen varaukset muunnetaan jännitteiksi, ja jännitteet vahvistetaan sekä konvertoidaan joko analogisiksi tai digitaalisiksi signaaleiksi. Näistä signaaleista muodostuu digitaalinen kuva. CMOS-kennojen pikselin muunnos varauksesta jännitteeksi sekä jännitteen vahvistus ja muuntaminen tapahtuvat taas samalla kenno-piirillä. Tämän takia CMOS-kennoilla päästään yleensä pienempään virrankulutukseen verrattuna CCD-kennoihin. CMOS-kennojen teknisiä vahvuuksia ovat korkea kuvataajuus (frame rate) ja korkea integraatioaste. CCD-kennojen teknisiä vahvuuksia ovat vähäinen kohina kuvissa sekä hyvä valoherkkyys. (Hornberg 2006, 364-367, 373-379.)

### 2.3.2 Kameran valinta

Konenäköjärjestelmän kameran valintaan vaikuttaa suuresti, se mihin sovellukseen sitä tarvitaan. Esimerkiksi paikoitussovelluksessa kameralta vaaditaan eri ominaisuuksia kuin pinnan laaduntarkkailussa.

Yksi tärkeä tekijä kameran valinnassa on resoluutiovaatimus. Resoluution pitää olla riittävä, että analysoinnin kohteet saadaan riittävän hyvin näkyviin, mutta taas liiallinen resoluutio voi olla haitaksi, koska suurempi resoluutio johtaa pidempään analysointi-aikaan. Resoluution valinnassa pitää ottaa myös huomioon kuva-alan suuruus, koska mitä suurempi kuva-ala tarvitaan, sitä suuremman resoluution kamera tarvitsee. Joskus voi olla kustannustehokkaampaa hankkia kaksi pieniresoluutiosta kameraa verrattuna yhteen erittäin suurella resoluutiolla varustettuun kameraan. (Leino, Valo & Kortelainen 2014a, 21-22.)

Toinen tärkeä tekijä on se, halutaanko älykamerajärjestelmä vai perinteinen konenäköjärjestelmä. Älykamerajärjestelmä ja perinteinen konenäköjärjestelmä eroavat toisistaan siten, että älykamerajärjestelmä ei vaadi erillistä tietokonetta toimiak-

seen, vaan älykamerajärjestelmä analysoi otetun kuvan itse ja välittää vain analysointituloksen eteenpäin. Se siis sisältää kaiken tarvittavat elektroniikan kuvan analysointiin kuten prosessorin ja muistit jne. Älykamerajärjestelmä myös sisältää lukuisia algoritmeja ja käyttöliittymän sovellusten nopeaan rakentamiseen. Perinteinen konenäköjärjestelmä taas lähettää otetun kuvan tietokoneelle analysoitavaksi. Älykameroiden hinnat ovat kalliimpia verrattuna perinteisiin kameroihin, ja älykamerajärjestelmät ovat usein laskentatehollisesti jäljessä kameran ja tietokoneen yhteistoiminnasta. (Leino ym. 2014a, 21-22.)

Kameran valinnassa pitää myös valita tarvitseeko värikameran vai riittääkö pelkkä harmaasävykamera. Värikamera kannattaa hankkia vain silloin, jos värien tunnistuksella on merkitystä. Usein värikameroissa värien muodostamiseen käytetty tekniikka aiheuttaa tarkkuuden tippumisen n. kolmasosaan vastaavan harmaasävykameran tarkkuudesta. (Leino ym. 2014a, 21-22.)

### 2.3.3 Optiikka

Konenäössä optiikan merkitys kuvanlaatuun on merkittävä. Optiikan tehtävä on heijastaa ja kohdistaa valo kameran kennolle ja skaalata kuva-alue kameran kennolle sopivaksi. Optiset linssit kuitenkin usein vääristävät kuvaa jollain tapaa. Vääristymiä pystytään kuitenkin korjaamaan kalibroimalla tai käyttämällä erikoislinsskejä. (Halinen 2007, 6-7.)

Nämä optiset vääristymät voidaan jakaa kahteen eri osaan: monokromaattisiin ja kromaattisiin vääristymiin. Kromaattiset vääristymiä johtuvat valon eri aallonpituuksien taittumisesta kameran linssissä. Tämä ilmenee esimerkiksi värien siroamisena. Monokromaattiset vääristymät taas tapahtuvat, vaikka valonsäteet olisivat samaa aallonpituutta. Monokromaattisia heijastuksia ovat esimerkiksi geometriset vääristymät kuten neulatyyny- ja tynnyrivääristymä, joita havainnollistaa kuvio 4. Nämä vääristymät aiheutuvat siitä, että valonsäteet taittuvat linssin reunoilla eri tavalla kuin linssin keskellä. (Hornberg 2006, 316-321.)



Kuvio 4. Neulatyyny- (keskellä) ja tynnyrivääristymä (oikealla) (Vision Doctor 2020b).

Optiikan valinnassa tärkeitä tekijöitä ovat mm. optiikan riittävyys valitun resoluution kanssa, haluttu kuvakoko (field of view), kuvausetäisyys ja kameran kennon koko. On myös tärkeää varmistaa, että optikka sopii kiinnitykseltään valittuun kameraan. (Leino ym. 2014a, 22-23.)

Oikean optiikan valinnan apuna voi käyttää laskureita, jotka löytyvät esim. Vision Doctorin verkkosivulta: <https://www.vision-doctor.com/en/service-overview.html>

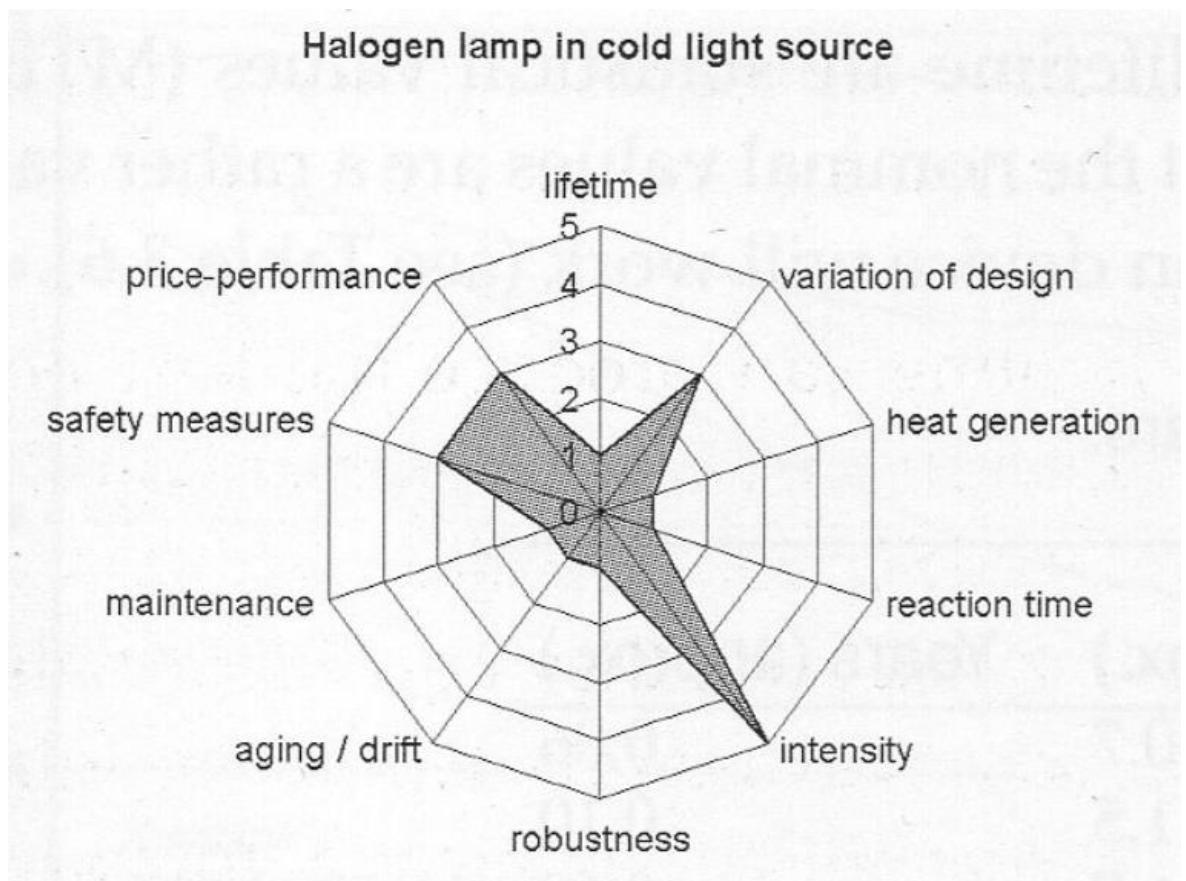
## 2.4 Valaistus

Valaistus on yksi konenäköjärjestelmän tärkeimmistä komponenteista, jota ilman yksinkertainenkin järjestelmä voi olla hankala toteuttaa. Oikean valaistuksen valitsemisessa pitää huomioida työn vaatimukset ja olosuhteet, kuten käyttöympäristön valaistus ja varjot sekä niiden muutokset. Hyvin suunniteltu valaistus minimoi kuvankäsittelytarpeen ja varmistaa, että oikeat piirteet saadaan korostettua kuvattavasta kohteesta. (Leino, Kortelainen & Valo 2014b, 25.)

### 2.4.1 Valonlähteet

Konenäössä valaisuun voidaan käyttää useita eri valonlähteitä. Kaikilla valonlähteillä on omat hyvät ja huonot puolensa, joten kokeilemalla saa parhaan tuloksen. Seuraavassa osiossa esitellään tyypillisimpiä valonlähteitä, sekä niiden hyviä ja huonoja puolia konenäköjärjestelmissä.

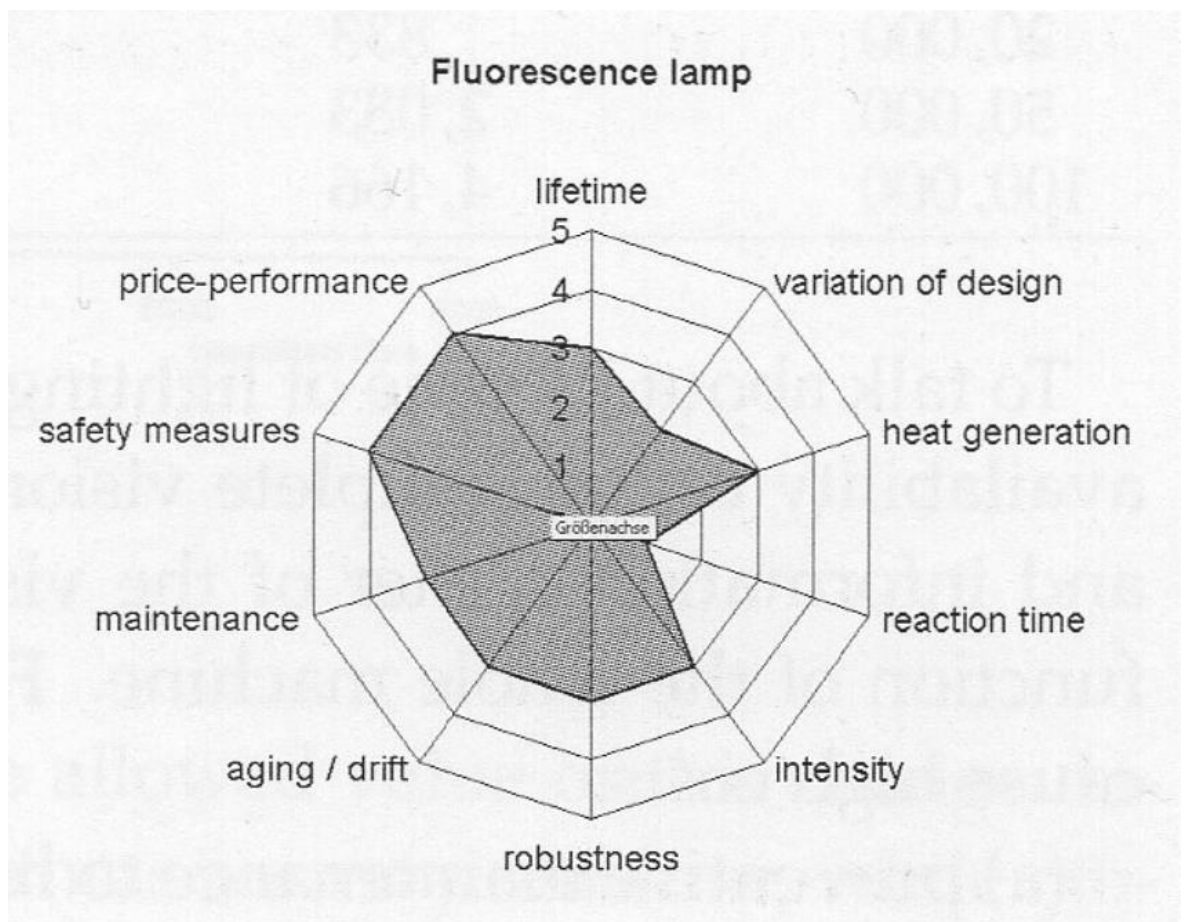
**Halogeenilamput.** Halogeenilamppujen toimintaperiaate muistuttaa tavallista hehkulamppua. Lampussa olevan ohuen volframilangan läpi johdetaan sähkövirta, jonka seurauksena lanka kuumenee ja alkaa hehkua tuottaen valoa. Volframilankaa ympäröi inertti kaasu, johon on lisätty jotain halogeenia (esim. bromi tai jodi). Halogeenilampun etuja ovat mm. suuri valovoima, matala käyttöjännite ja korkea lämmönsieto. Halogeenilampun huonoja puolia ovat esim. sen lyhyt käyttöikä (n. 300–2000 tuntia), herkkyys jännitteen muutoksille ja värähtelyille, sekä huono valotehokkuus. Konenäköjärjestelmissä halogeenivalaisimia valaisimia käytetään lähinnä suurien alueiden valaisemiseen. (Hornberg 2006, 89-91; Vision Doctor 2020a.) Kuvio 5 havainnollistaa halogeenilampun vahvuuksia ja heikkouksia.



Kuvio 5. Halogeenilampun vahvuuksien ja heikkouksien vertailua (Hornberg 2006, 101).

**Fluoresoivat valonlähteet.** Fluoresoivat valonlähteet, kuten loisteputket, ovat ns. kaasupurkauslamppuja. Loisteputket sisältävät elohopeaa, joka kaasuuntuu sähköpurkauksen vaikutuksesta ja tuottaa UV-säteilyä. Ulkopuolelle nähtävä valo syntyy

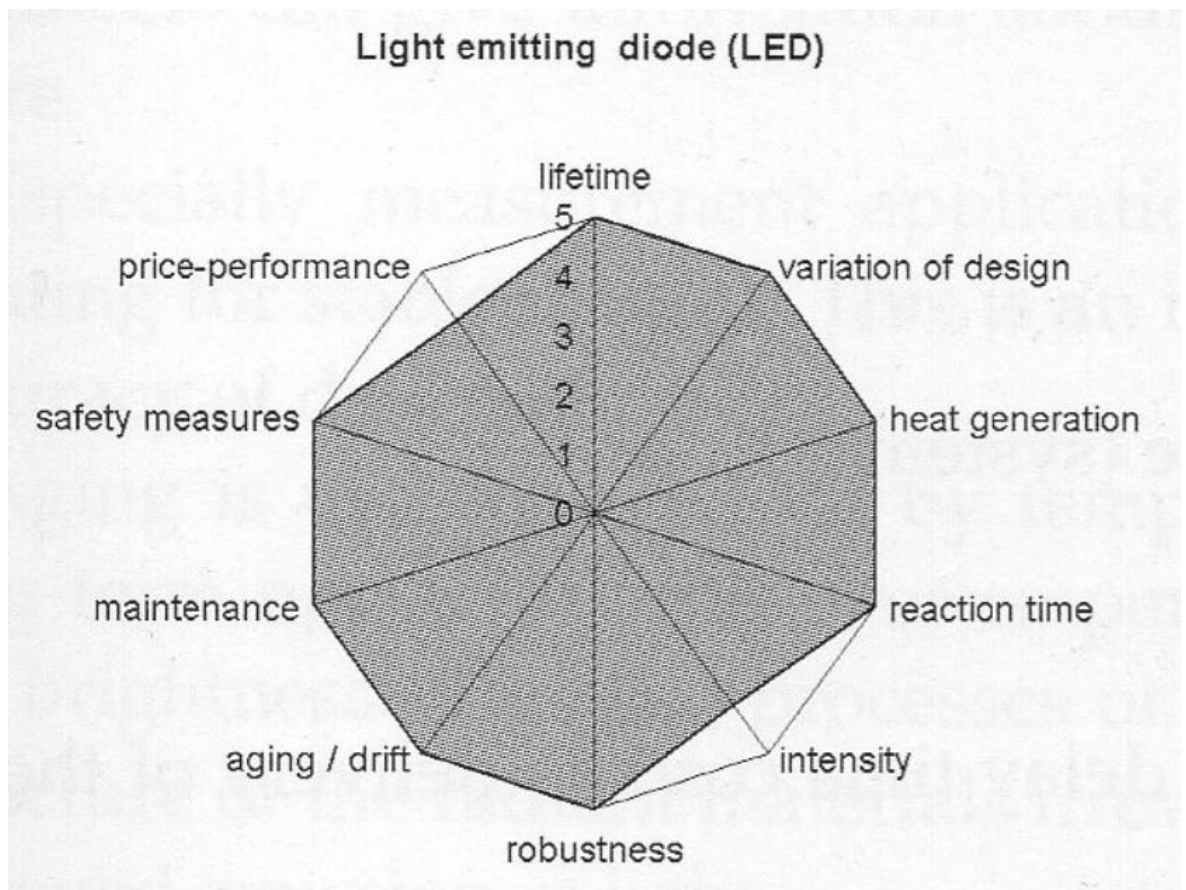
siitä, kun UV-säteily osuu putken sisäpinnalle pinnoitettuun fosforisuolaan. Perinteiset loisteputket toimivat 50 Hz:n taajuudella, mutta konenäössä käytetään huomattavasti korkeampia taajuuksia välkkymisen estämiseksi. Fluoresoivien valonlähteiden etuja ovat mm. niiden halpa hinta ja niiden kyky tuottaa paljon erilaisia värilämpötiloja (3000–6000 K). Huonoja puolia ovat esim. niiden soveltuvuus vain staattiseen valaisuun, ne vaativat erillisen liitäntälaitteen välähdystaajuuden kasvattamiseksi ja niiden kohtalaisen lyhyt käyttöikä (2500–12000 h). Konenäköjärjestelmissä fluoresoivia valonlähteitä käytetään tyypillisesti suurien alueiden valaisemiseen ja matala kustannuksellisiin sovelluksiin. (Hornberg 2006, 93-95; Vision Doctor 2020a.) Kuvio 6 havainnollistaa fluoresoivien valonlähteiden vahvuuksia ja heikkouksia.



Kuvio 6. Fluoresoivien valonlähteiden vahvuuksien ja heikkouksien vertailua (Hornberg 2006, 101).

**LED.** Ledit (Light-Emitting Diode) ovat puolijohdekomponentteja, jotka säteilevät valoa, kun niiden läpi johdetaan virtaa. Led-valaisimet ovat yleistyneet voimakkaasti

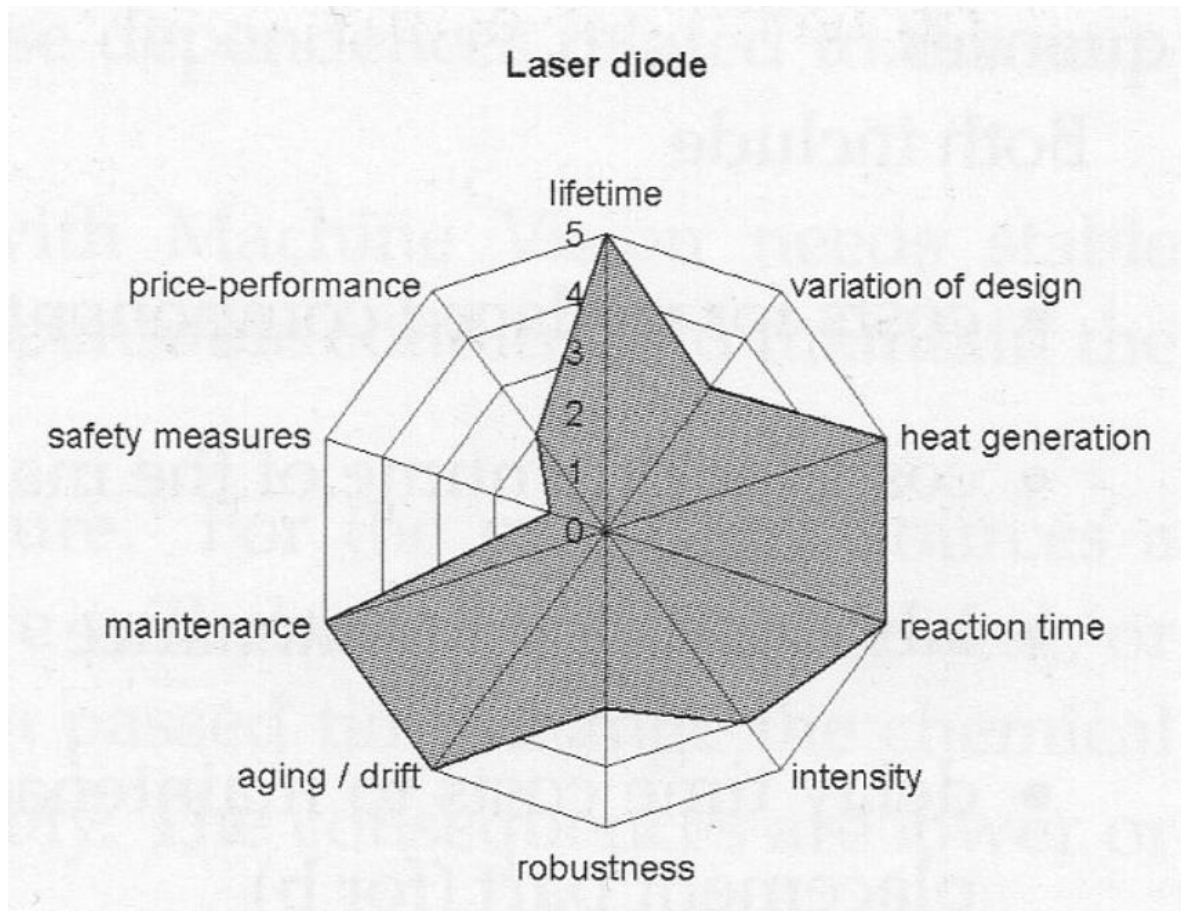
viime vuosina ja monesti ne ovat paras vaihtoehto konenäköjärjestelmiin. Ledin etuja ovat mm. niiden pitkä käyttöikä (jopa 30000–100000 h), matala virrankulutus ja lämmöntuotto. Niitä voidaan moduloida eli sytyttää ja sammuttaa suurella taajuudella, sekä niillä pystyy emittoimaan kaikkia värejä ja infrapunavaloa. (Hornberg 2006, 95-98; Vision Doctor 2020a.) Kuvio 7 havainnollistaa Led-valaisimien vahvuuksia ja heikkouksia.



Kuvio 7. Led-valaisimien vahvuuksien ja heikkouksien vertailua (Hornberg 2006, 101).

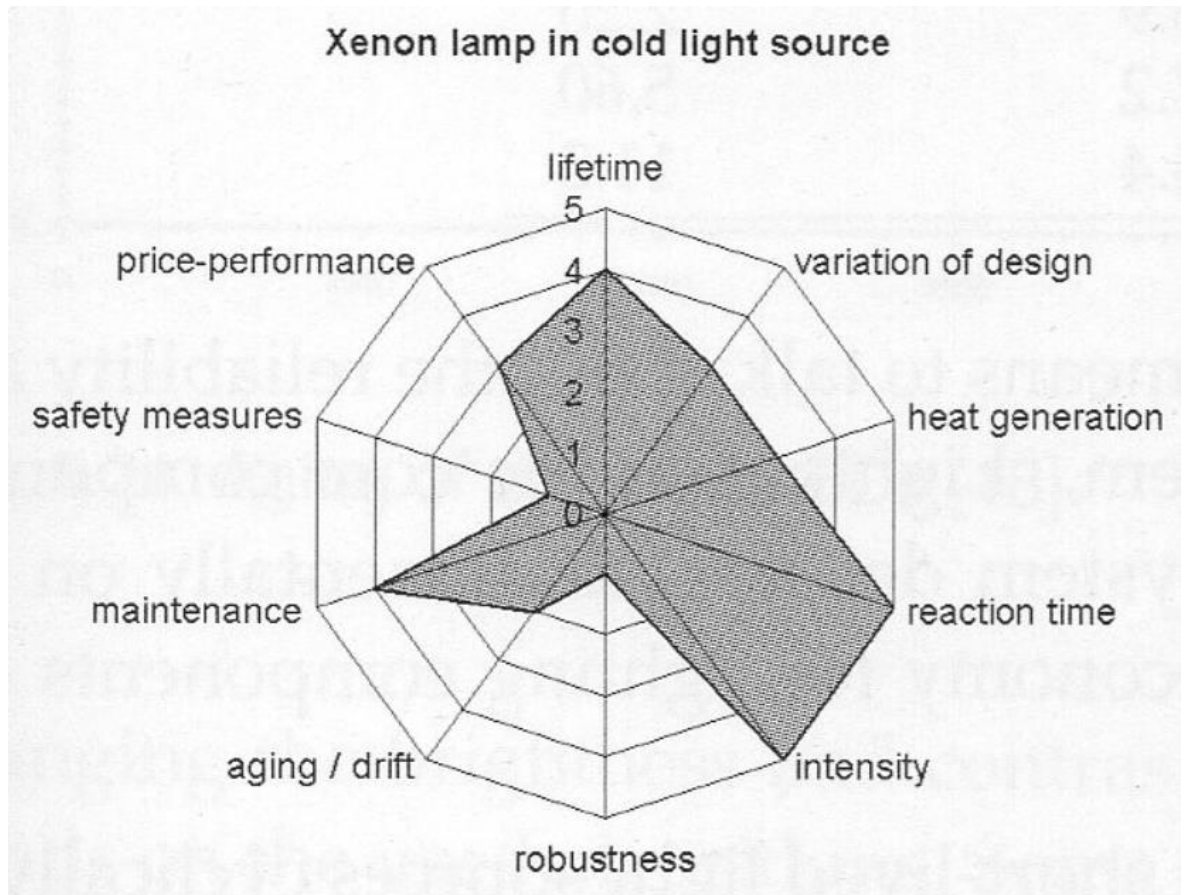
**Laser.** Laser (Light amplification by stimulated emission of radiation) on laite, joka tuottaa koherenttia eli saman vaiheista yhteen suuntaan etenevää valoa. Tällöin aallot vahvistavat toisiaan, eli valon intensiteetti on korkea. Yleisimmät lasertyypit ovat puolijohde- ja kaasulaserit. Konenäössä käytetään lähinnä vain puolijohdelasereita. Laservalot soveltuvat hyvin etäisyyksien mittaamiseen, minkä vuoksi niitä

käytetään usein 3D-kuvautekniikoiden valonlähteinä. (Hornberg 2006, 99-100; Vision Doctor 2020a.) Kuvio 8 havainnollistaa Laser-valaistuksen vahvuuksia ja heikkouksia.



Kuvio 8. Laser-valaistuksen vahvuuksien ja heikkouksien vertailua (Hornberg 2006, 101).

**Xenon-lamput.** Xenon-lamput ovat ns. kaasupurkauslamppuja, jossa xenon-kaasun hehkuttaminen tuottaa valoa. Xenon-valaisimen etuja ovat suuri valovoima, korkea väriämpötila, nopea välähdystaajuus ja pitkä käyttöikä. Sen huonoja puolia taas ovat esim. sen kallis hinta ja korkea käyttöjännite. Xenon-valaisimet soveltuvat hyvin konenäkösovelluksiin, joissa tarvitaan lyhyttä prosessointiaikaa tai värien tunnistamista. (Hornberg 2006, 92-93.) Kuvio 9 havainnollistaa xenon-lampun vahvuuksia ja heikkouksia.



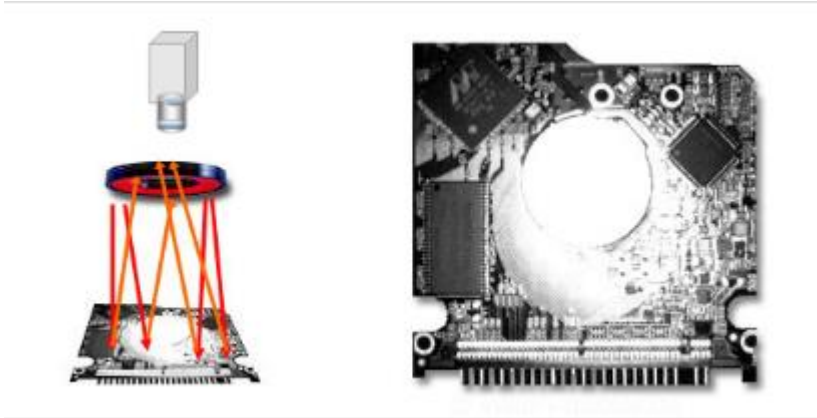
Kuvio 9. Xenon-lampun vahvuuksien ja heikkouksien vertailua (Hornberg 2006, 101).

#### 2.4.2 Valaistusgeometriat

Valaistuksen suunnittelussa on tärkeää ottaa huomioon valonlähteiden lisäksi käytettävä valaistusgeometria. Valaistusgeometrialla tarkoitetaan valaistuksen suuntaa suhteessa kohteeseen ja kameraan. (Leino ym. 2014b, 25.) Seuraavassa osiossa esitellään tyypillisimpiä valaistusgeometrioita, sekä niiden hyviä ja huonoja puolia konenäköjärjestelmissä.

**Kohtisuora valaisu.** Kohtisuorassa valaisussa valo pyritään osoittamaan kohteeseen samasta suunnasta, josta kamera sitä kuvaa. Kohtisuora valaistus voidaan toteuttaa erimuotoisilla valaisimilla. Kuvio 10 havainnollistaa kohtisuoraa valaistusta rengasvalaisimella. Menetelmällä saadaan aikaan hyvä yleisvalaistus, mutta se saattaa aiheuttaa peilimäisiä heijastuksia kiiltäville pinnoille. (Leino ym. 2014b, 26.)





Kuvio 10. Kohtisuora valaisu rengasvalaisimella (Vision Doctor 2020a).

**Taustavalaisu.** Taustavalaisussa valonlähde on kappaleen alla, jolloin kappale näkyy ns. mustana "varjokuvana" eli siluettina (kuvio 11). Taustavalaisulla saadaan mm. eroteltua kohteen ääriviivat ja reiät, minkä vuoksi sitä käytetään tyypillisesti silloin, kun halutaan tutkia kappaleiden mittoja ja pinta-aloja. (Leino ym. 2014b, 27.)



Kuvio 11. Reiällinen kappale taustavalaisimella (Leino ym. 2014b, 27).

**Sivuväläisy.** Sivuväläistuksessa valo suunnataan tarkasteltavaan kohteeseen pienestä kulmasta suhteessa tasoon, jolla kohde on (kuvio 12). Kun valo kohdistetaan

sivulta päin kohteen pinnalle, se heijastuu tasaisilta pinnoilta vastakkaiseen suuntaan. Taas jos pinnalla on epätasaisuuksia, valo heijastuu niistä suoraan ylöspäin. Kameran kuvatessa ylhäältä päin pinnanmuodot korostuvat kameralle vaaleana. Sivuvuvalaisu sopii hyvin pinnanlaadun ja pintavikojen tarkasteluun. (Leino ym. 2014b, 26.)



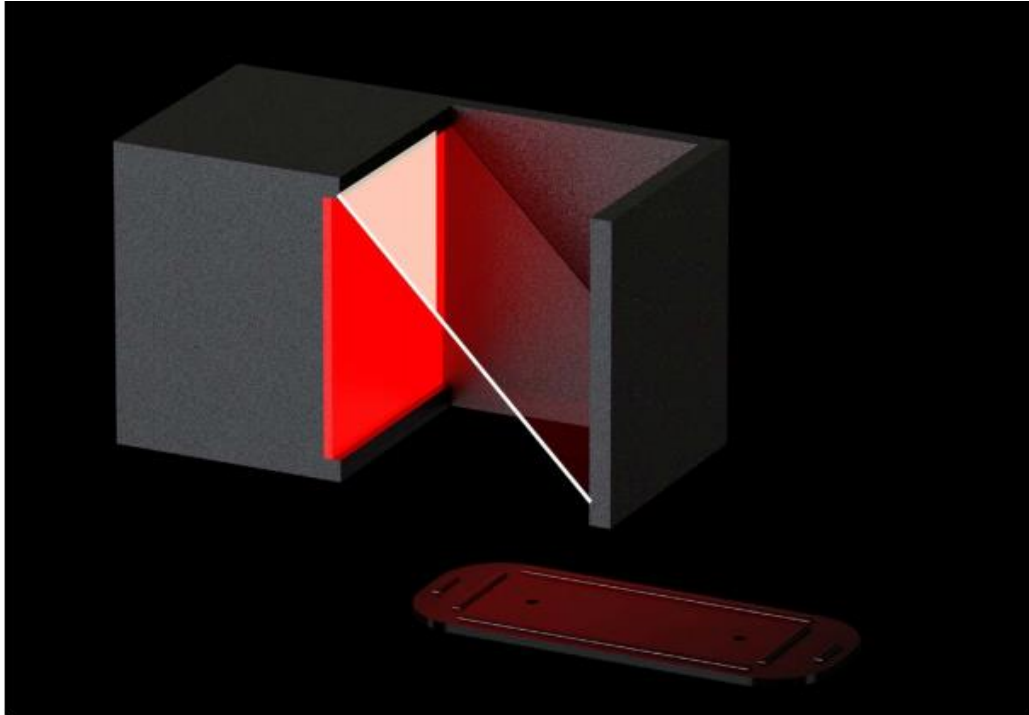
Kuvio 12. Rengasmainen sivuvalo (Leino ym. 2014b, 26).

**Diffuusivalaisu kupolivalonlähteellä.** Diffuusivalaisussa kupolivalonlähteellä valo heijastetaan kupolin kautta kuvattavaan kohteeseen, tämän avulla kuvasta saadaan poistettua varjoja ja heijastuksia (kuvio 13). Käyttämällä erilaisia kupolin muotoja voidaan vaikuttaa siihen, mitä kohtia kappaleesta halutaan korostaa. Kupolivalaistus sopii hyvin heijastavien ja koverien kohteiden valaistukseen. (Leino ym. 2014b, 28.)



Kuvio 13. Diffuusikupolivalaisin (Leino ym. 2014b, 28).

**Aksiaalinen diffuusivalaisu.** Aksiaalisessa diffuusivalaistuksessa valo suunnataan kappaleeseen kameran suunnasta puoliläpäisevän peilin avulla (kuvio 14). Menetelmä soveltuu hyvin tasaisille pinnoille, kun taas kaarevilla pinnoilla menetelmä voi olla ongelmallinen. Aksiaalinen diffuusivalaisu sopii esim. kaiverrusten tarkasteluun. (Leino ym. 2014b, 29.)



Kuvio 14. Aksiaalisen diffuusivalaisin (Leino ym. 2014b, 29).

### 2.4.3 Valaistustekniikat

**Strobovalaisu.** Strobo- eli salamavalaisua käytetään tyypillisesti silloin, kun kuvataan liikkuvaa kohdetta ja liike tuottaisi häiriöitä kuvan tarkkuuteen. Strobovalaisussa saadaan kohteeseen lyhyeksi ajaksi voimakas valaisu. Valotuksen tapahtuessa lyhyessä ajassa kohde ei ehdi juurikaan liikkua ja kuvasta tulee tarkempi. (Leino ym. 2014b, 31.)

**Rakenteellinen valaisu.** Rakenteellisessa valaisussa käytetään kohteen valaamiseen valaisukuviota, joka usein tuotetaan laservalolla (kuvio 15). Valaisukuviona voidaan esim. käyttää viivaa (kuvio 15), viivaverkkoa tai pistematriisia. Rakenteellisessa valaisussa tarkkaillaan valaisukuvion muutoksia kohteen pinnalla, jolloin saadaan analysoitua kohteen pinnan muotoja. (Leino ym. 2014b, 30.)



Kuvio 15. Laserviivalla toteutettu rakenteellinen valaisu (Leino ym. 2014b, 30).

## 2.5 Kalibrointi

Kalibroimisella tarkoitetaan kameran kuvan muodostuneiden mittavirheiden määrittämistä ja pikseliarvojen kytkemistä reaali maailman arvoihin, kuten metreihin tai tuumiin. Konenäköjärjestelmän kalibrointi koostuu kahdesta eri vaiheesta, jotka ovat sisäinen ja ulkoinen suuntaus. Sisäisessä suuntauksessa määritetään kameran sisäiset parametrit ja ulkoisessa suuntauksessa määritetään sisäisesti määritetyn kamerakoordinaatiston ja maailmankoordinaatiston välisen koordinaatiomuunnoksen. Ulkoista suuntausta ei aina välttämättä tarvitse tehdä, vaan sitä tarvitsee vain silloin, jos konenäköjärjestelmään pitää yhdistää eri koordinaatistoa käyttävä ohjauslaite kuten robotti. (Microscan, [viitattu 16.03.2020]; Adaptive Vision [viitattu 16.03.2020]).

Sisäinen suuntauksen tarkoituksena on poistaa kuvan vääristymät, sekä polttovälin ja optisen keskipisteen määrittäminen. Nämä vääristymät voidaan korjata kameran sisäisten parametrien säätämällä eli kuvankäsittelyllä. Optisia vääristymiä tarkoitetaan linssi- ja suodatinvääristymiä, joita ovat esim. aiemmin mainitut neulatyyny- ja tynnyrivävääristymä. Sisäisen suuntauksen apuna voidaan käyttää kappaletta, jonka reaali maailman arvojen avulla voidaan suorittaa muunnos tai vaikka käyttä-

mällä kalibrintilevyä. Ulkoisessa suuntauksessa taas pitää määrittää kamerakoordinaatiston ja ohjauslaitteen maailmankoordinaatiston välille kiertymä- ja/tai siirtymäparametreja, joiden avulla pystytään esim. paikoittamaan kappale toisesta koordinaatistosta. Monessa konenäköjärjestelmässä pitää ennen ulkoista suuntausta tehdä kamerakoordinaatistojen väliset suhteelliset suuntaukset. (Microscan, [viitattu 16.03.2020]; Adaptive Vision [viitattu 16.03.2020].)

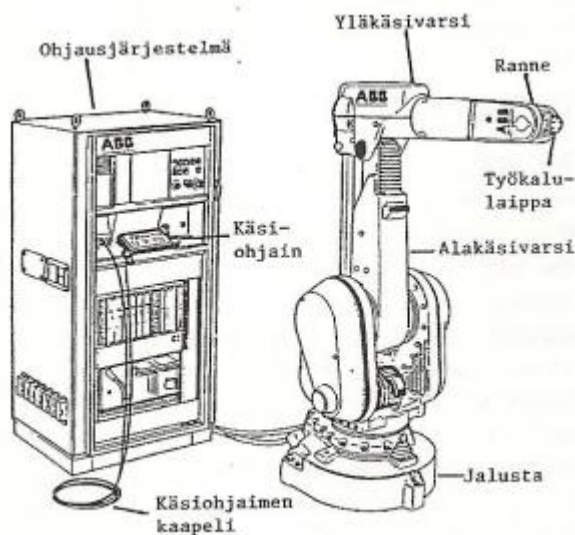
Monissa konenäköjärjestelmissä on valmiita ratkaisuja ulkoisen suuntauksen suorittamiseen. Siihen voi esimerkiksi käyttää ns. point-to-point-menetelmää, jossa koordinaatiomuunnos suoritetaan etsimällä eri koordinaatistojen yhteisiä pisteitä. Näiden pisteiden välisistä eroista voidaan laskea koordinaatistojen väliset kiertymät ja paikkavektorit. (Microscan, [viitattu 16.03.2020]; Adaptive Vision [viitattu 16.03.2020].)

### 3 TEOLLISUUSROBOTIT

#### 3.1 Teollisuusrobotin määritelmä ja rakenne

Kansainvälisen robottiyhdistyksen määritelmän mukaan teollisuusrobotti on teollisuuden sovelluksissa käytettävä monipuolinen vähintään kolmenivelinen mekaaninen laite, joka on uudelleen ohjelmitavissa, ja on suunniteltu liikuttamaan osia, kappaleita, työkaluja tai erikoislaitteita ohjelmitavien liikkein monenlaisten tehtävien suorittamiseksi (Kuivanen 1999, 13).

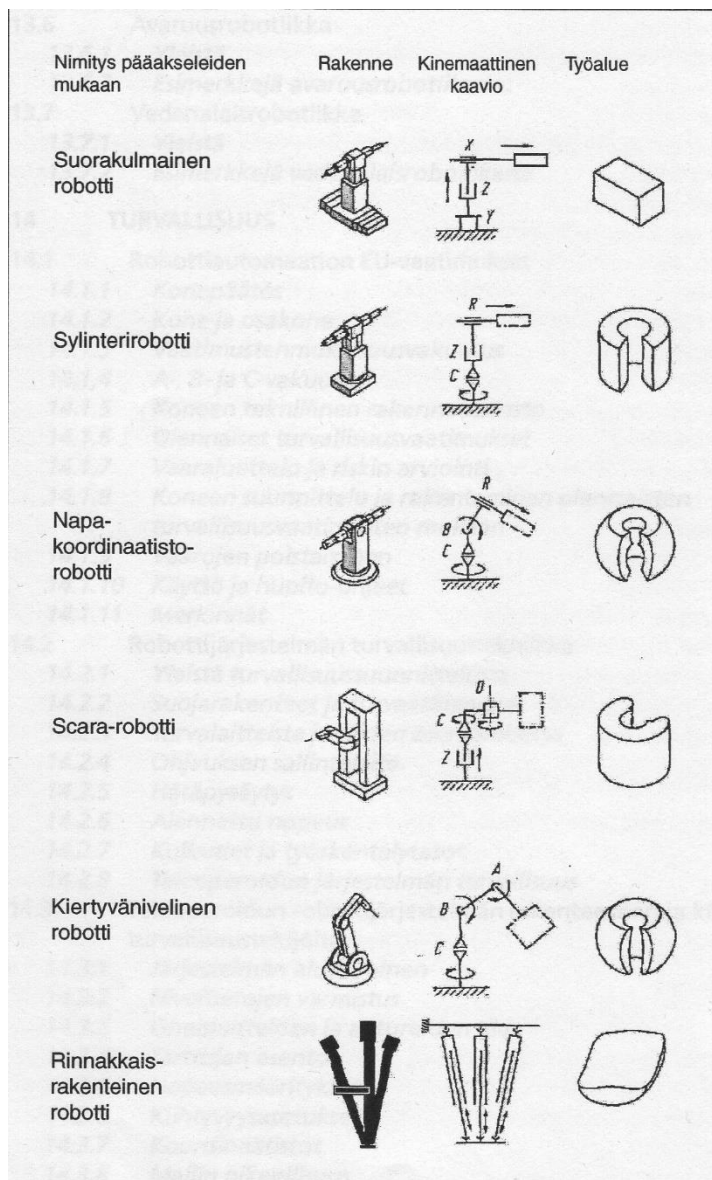
Teollisuusrobotti muodostuu tyypillisesti kahdesta pääkomponentista, jotka ovat mekaaninen manipulaattori ja ohjausjärjestelmä (kuvio 16). Mekaaninen manipulaattori koostuu toistensa suhteen liikkuvista tukivarsista eli nivelistä. Nivelten liike tuotetaan servotoimilaitteilla etukäteen ohjelmitujen liikeratojen mukaan erillisten tai ulkoisten tuntoelimien perusteella. Ohjausjärjestelmä koostuu tyypillisesti robotin keskusyksiköstä, massamuistista, käsiohjaimesta, ulkoisista liitännöistä, servotoimilaitteista ja tehonlähteestä. (Kuivanen 1999, 13-14, 34.)



Kuvio 16. Teollisuusrobotin peruskomponentit (Kuivanen, 1999, 13).

### 3.2 Robottityypit

Standardi ISO 8373 määrittelee yleisimmät robottityypit, niiden mekaanisen rakenteen ja työalueiden mukaan. (Kuivanen 1999, 12). Seuraavassa osiossa esitellään nykYTEOLLISUUDEN tyypillisimpiä robottityyppejä. Näiden robottityyppien rakenteita, kinemaattiset kaaviot ja työalueet on havainnollistettu kuviossa 17. Napakoordinaatorobotit ja sylinterirobotit ovat nykyisin sen verran harvinaisia teollisuudessa, että niitä ei sen vuoksi esitellä tässä.

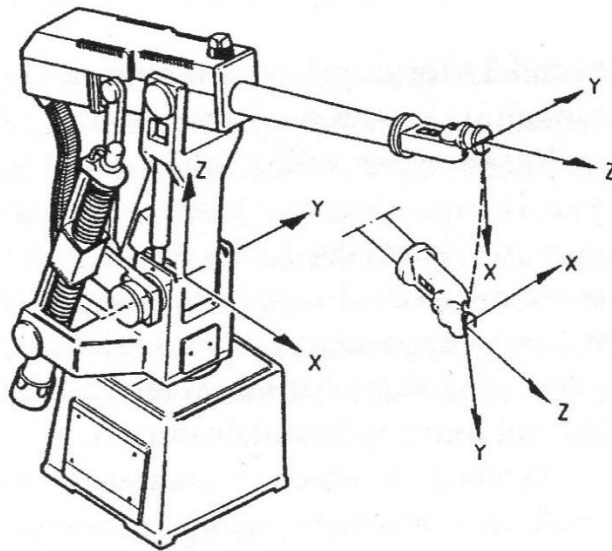


Kuvio 17. Yleisimmät robottityypit ja niiden rakenne-esimerkkejä (Kuivanen 1999,12).



### 3.2.1 Kiertyvänivelinen robotti

Kiertyväniveliset robotit (articulated robots) ovat tyypillisimpiä robotteja teollisuudessa ja niitä valmistaa suurin osa robotteja valmistavista yrityksistä. Tyypillisesti kiertyvänivelisillä roboilla on 6 kiertyvää niveltä (kuvio 18), mutta nivelien määrä voi kuitenkin vaihdella 2–10 nivelen välillä. Kiertyvänivelisten robottien etuja ovat mm. niiden monikäyttöisyys ja laaja työalue. Kiertyvänivelisiä robotteja käytetään monissa teollisuuden sovelluksissa, kuten hitsauksessa, kappaleenkäsittely- ja kokoonpanotehtävissä. (Technavio 2018.)

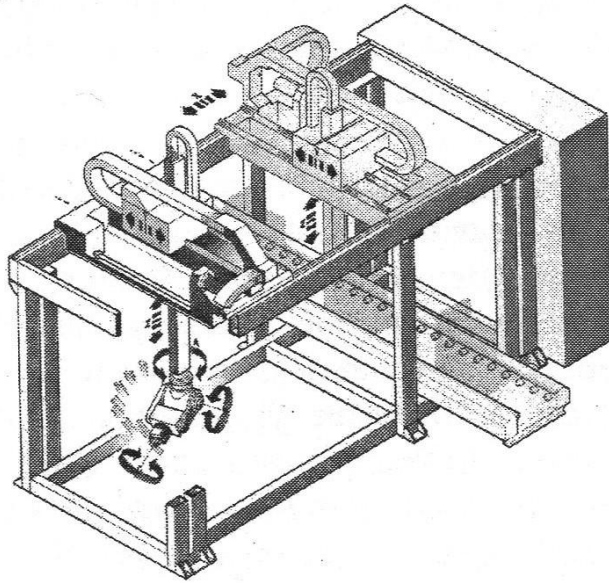


Kuvio 18. Kiertyvänivelinen teollisuusrobotti (Kuivanen 1999,17).

### 3.2.2 Suorakulmainen robotti

Suorakulmaiset robotit (rectangular robot) koostuvat vähintään kolmesta lineaarisesta vapausasteesta (kuvio 19). Suorakulmaisia robotteja kutsutaan myös karteesisiksi roboteiksi (cartesian robots). Suorakulmaisten robottien etuja ovat niiden yksinkertainen ja modulaarisen rakenteen mahdollistama laaja työalue. Ne pystyvät

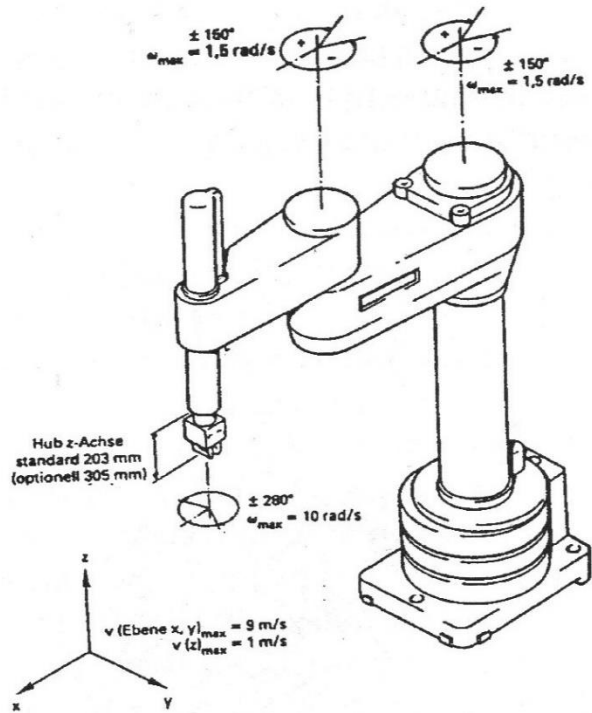
käsittämään suuriakin massoja, ja niitä on helppo ohjelmoida. Suorakulmaisia robotteja käytetään tyypillisesti kappaleenkäsittely- ja kokoonpanotehtävissä. (Technavio 2018.)



Kuvio 19. Suorakulmainen teollisuusrobotti (Kuivanen 1999,17).

### 3.2.3 SCARA-robotti

Scara-robotit (Selective compliance assembly robot arm) koostuvat kolmesta kiertävästä ja yhdestä lineaarisesta pysty akselista (kuvio 20). Scara-robottien etuja ovat mm. niiden nopeus ja korkea toistotarkkuus. Scara-robotteja käytetään tyypillisesti pieniä kuormia käsittelevissä sovelluksissa, kuten pakkaus- ja kokoonpanotehtävissä. (Technavio 2018.)



Kuvio 20. Scara-teollisuusrobotti (Kuivanen 1999,17).

### 3.2.4 Rinnakkaisrakenteinen robotti

Rinnakkaisrakenteiset robotit (parallel link robots) koostuvat useista rinnakkaisista nivelistä, jotka on kytketty yhteiseen jalustaan (kuva 1). Rinnakkaisrakenteisten robottien etuja ovat niiden todella korkea nopeus sekä toistotarkkuus. Rinnakkaisrakenteisiä robotteja käytetään tyypillisesti pakkaus- ja kokoonpanotehtävissä. (Technavio 2018.)



Kuva 1. Delta-robotti (Omron 2020).

### 3.2.5 Mobiilirobotit

Mobiilirobotit ovat teollisuuskäyttöön tarkoitettuja itsenäisesti liikkuvia koneita, jotka käyttävät tekoälyä ja antureita, kuten laserskannereita, ympäristön tunnistamiseen ja siellä navigoimiseen. Mobiilirobotteja käytetään tyypillisesti sisälogistiikan automatisointiin esim. sairaalaympäristöissä. (Pakkanen & Koukkari 2016, 41.)

### 3.2.6 Yhteistyörobotit

Yhteistyörobotti, eli cobotti, on nimensä mukaisesti robotti, joka on suunniteltu turvallisesti työskentelemään ihmisen kanssa samassa tilassa. Turvallinen työskentely mahdollistetaan esim. tehojen rajoittamisella kontaktin tapahtuessa ja ympäristön tarkkailulla. Yhteistyörobotissa on usein pehmustettu sekä pyöristetty ulkopinta kontaktin vahingon minimoimiseksi, ja ne ovat hyvin kevytrakenteisia. (IFR 2018, 2-3.)

Kansainvälisten standardien ISO 10218 osan 1 ja 2 mukaan teollisuusroboteissa voi olla neljä erityyppistä yhteistyöominaisuutta: robotin käsin liikuttelu, nopeuden ja välimatkan monitorointi, turvaluokiteltu valvottu pysäytys sekä tehon ja voiman rajoittaminen (Bélanger-Barrette 2015).

**Robotin käsin liikuttelu.** Robotin käsiohjauksella voidaan opettaa robotille liikera-toja. Robotti on moottoroitu, mutta robotin liikkeet ovat hallinnassa. Tämän tyyppistä yhteistyötä pystyy käyttämään vain tietyissä tarkoituksissa. Jos robotin kanssa ha-lutaan toimia samassa työskentelytilassa muussa kuin robotin käsiohjaustilassa, se tarvitsee muita turvatoimintoja. (Bélanger-Barrette 2015.)

**Nopeuden ja välimatkan monitorointi.** Robotti tunnistaa ihmisen työskentelyalu-eella ja hidastaa liikkeitään ihmisen sijainnin mukaan. Robotti lopettaa työskentelyn henkilön tullessa liian lähelle, ja henkilön poistuessa robotti jatkaa työskentelyään. (Bélanger-Barrette 2015.)

**Turvaluokiteltu valvottu pysäytys.** Robotti lopettaa työskentelyn ihmisen tullessa työskentelytilaan. Yleensä hyödynnetään yhtä tai useampaa sensoria ihmisen ha-vainnoimiseen. Tällöin ihminen pystyy työskentelemään robotin työskentelytilassa, ilman että robotin ohjelma keskeytyy. (Bélanger-Barrette 2015.)

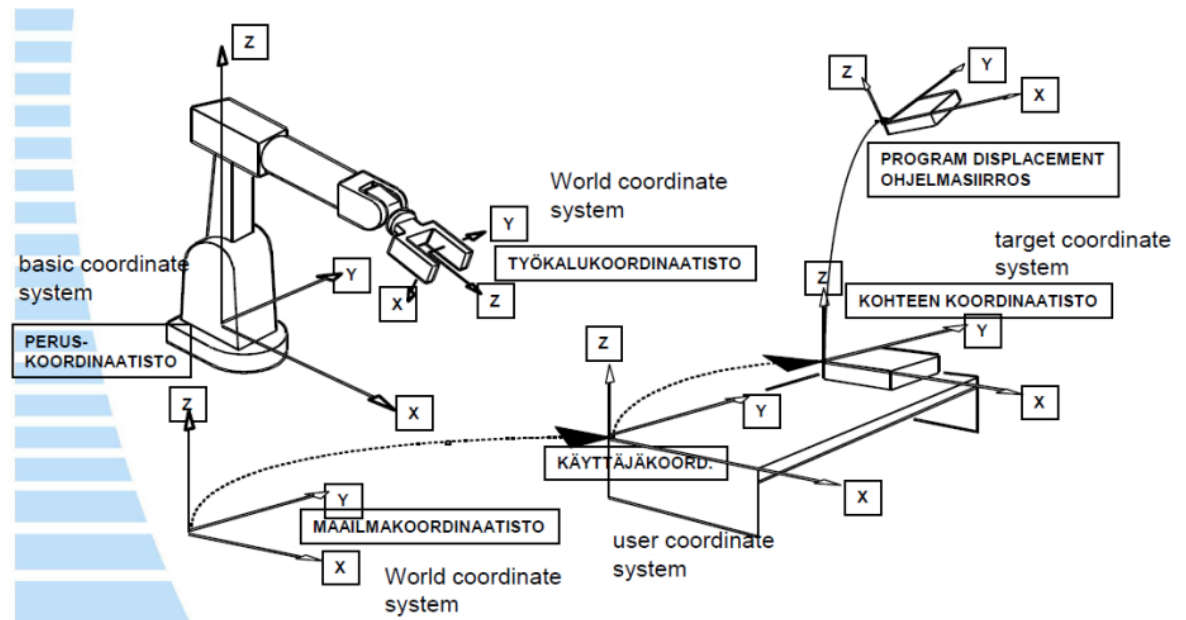
**Tehon ja voiman rajoittaminen.** Robotti rajoittaa tehojaan ollessaan tekemisissä ihmisen kanssa, ja robotti pysähtyy sen tunnistessa poikkeavia voimia liikeradois-saan. Robotti rajoittaa myös voimaa törmäyksen yhteydessä, jottei ihmiselle eikä robotille koituisi vahinkoa. (Bélanger-Barrette 2015.)

### 3.3 Robotin koordinaatistot

Teollisuusrobottien yleisesti käytettäviä koordinaatistoja ovat: maailmankoordinaa-tisto, peruskoordinaatisto ja työkalukoordinaatisto. Peruskoordinaatisto on robotin kantaan sidottu koordinaatisto, jossa tavallisesti Z-akseli yhtyy ensimmäiseen va-pausasteen akseliin, X-akseli osoittaa ensimmäisen nivelen työalueen keskikohtaa ja XY-taso yhtyy lattiaan. Työkalukoordinaatisto on taas robotin työkalulaippaan si-dottu koordinaatisto ja maailmakoordinaatisto on robotin ympäristöön, esimerkiksi

robotin oheislaitteeseen, sidottu robotin ulkopuolinen koordinaatisto. (Kuivanen 1999, 20-21.)

Teollisuusroboteissa myös usein käytetään käyttäjä- ja kohdekoordinaatistoa. Käyttäjäkoordinaatisto voidaan sijoittaa käyttäjän haluamaan paikkaan ja kohdekoordinaatisto voidaan sitoa esim. työstettävään kappaleeseen. (Pakkanen & Koukkari 2016, 17-18.) Teollisuusrobotin eri koordinaatistoja havainnollistaa kuvio 21.



Kuvio 21. Teollisuusrobotin koordinaatistoja (Pakkanen & Koukkari 2016, 18).

### 3.4 Robotin ohjelmointi

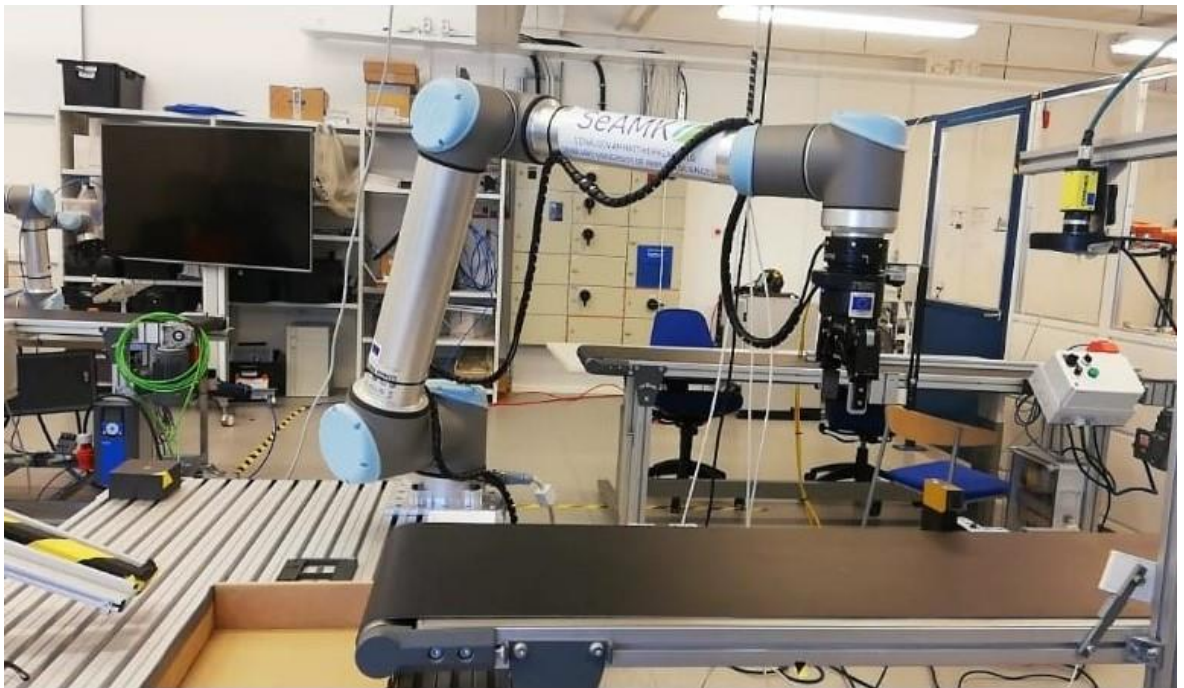
Robottien ohjelmointiin käytetään yleensä robotin käsiohjainta, jolla voidaan luoda robotin ohjelma kokonaisuudessaan. Perinteisesti robotteja ohjelmoidaan viemällä käsiohjaimen avulla robotin työkalu haluttuun pisteeseen ja tallentamalla aseman robotin muistiin. Asemien välillä liikkumiseen tai kohdeaseman valitsemiseen taas käytetään erilaisia hyppykäskyjä ja aliohjelmiä. (Kuivanen 1999, 79.)

### **3.4.1 Offline-ohjelmointi**

Offline-ohjelmoinnissa robotille luodaan simulointimallin perusteella robottivalmistaja-kohtaiset robottikoodit robottiohjaimeen. Tällöin robotin ohjelma voidaan tehdä etukäteen ilman robottia. Offline-ohjelmoinnin etuja ovat seisokkiaikojen minimointi, ohjelmoinnin turvallisuus, ohjelman validointi ja testaus, laadun parantuminen ja vakiintuminen, sekä robottisolun ohjauksen suunnittelu ja testaus etukäteen. (Pakkanen & Koukkari, 29.)

## 4 LAITTEIDEN VALINTA JA ESITTELY

Työssä käytettävä järjestelmä (kuva 2) rakennettiin Seinäjoen ammattikorkeakoulun robotiikan laboratorioon. Robottisolu koostuu Universal Robotin UR10-käsivarsirobotista, Cognexin In-Sight Micro -sarjan 1403-mallin älykamerasta, Latabin rengasvalosta, optisesta anturista, liukuhihnasta ja sitä ohjaavasta Omronin SYSDRIVE 3G3JV -taajuusmuuntajasta. Liukuhihnaan on liitetty Omronin inkrementaalipulssi-anturi E6C2-CWZ5B, joka on yhdistettynä robotin I/O-rajapintaan. Robotin ohjelma ohjelmoidaan UR10-robotin PolyScope-käyttöliittymällä ja konenäköjärjestelmän tarkistusrutiini ohjelmoidaan Cognexin In-Sight Explorer -ohjelmalla.



Kuva 2. Kuva toteutetusta järjestelmästä, jossa vasemmalla puolella on UR10-yhteistyörobotti ja oikealla puolella on liukuhihna, jonka päälle on asennettu In-Sight Micro 1403 -älykamera, sekä Latabin rengasvalaisin.



#### 4.1 Universal UR10 -käsivarsirobotti

Työhön valittiin Universal Robotsin UR10-yhteistyörobotti (kuva 3), koska kohdeyrittäjä toimi työn alkaessa Universal Robotsin robottien maahantuojana. Yhteistyörobotin valintaan vaikutti myös se, että se mahdollisti turvallisen työskentelyn ympäristössä, jossa liikkui paljon ihmisiä.

Universal robot UR10 on ihmisen ja robotin väliseen yhteistyöhön suunniteltu kuuden vapausasteen käsivarsirobotti. UR10 koostuu robottikäsivarresta, ohjausjärjestelmästä (CB3.1) ja käsiohjaimesta eli pendantista. Käsivarsi koostuu alumiiniputkista, nivelistä, toimilaitteista ja niiden voimansiirtomekanismeista. Robotin enimmäisulottuvuus on 1300 millimetriä ja enimmäiskantokyky on 10 kilogrammaa, joka mainitaan robotin nimen numeroliitteessä UR10. Robotin jokainen nivel pystyy kääntymään 360 astetta jopa kulmanopeudella 180 astetta sekunnissa. Kahdella isoimmalla nivelellä (kanta- ja olkapäänivel) enimmäiskulmanopeus on 120 astetta sekunnissa. (Universal Robots 2019, 11, 85.)

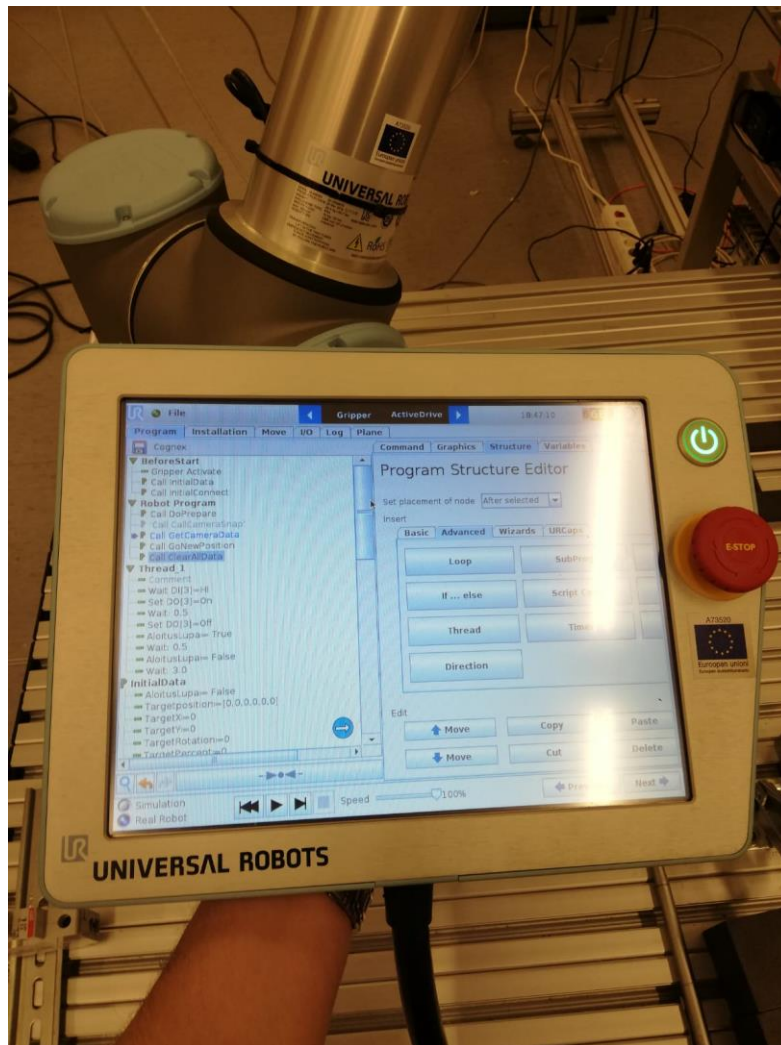
Robotin toimilaitteina käytetään kestopagneettisia AC-tahtimoottoreita ja niiden ohjaamiseen kaksoisenkooderijärjestelmään perustuvia servonohjaimia (Universal robot, [viitattu 19.03.2020]).



Kuva 3. UR10-yhteistyörobotti.

#### 4.1.1 Ohjausjärjestelmä ja tiedonsiirto

Ohjausjärjestelmä sisältää erilliset liitännät käsivarren toimilaitteiden ohjaamiseen sekä virransyötön ja käsiohjaimen (kuva 4) kytkennät. Käsiohjain on varustettu graafisella PolyScope-käyttöliittymällä ja kosketusnäytöllä, jolla voidaan ohjelmoida robottia. Ohjausjärjestelmä sisältää myös I/O-rajapinnan ja Ethernet-liitännän. UR10 tukee myös monia tiedonsiirron kenttäväyläprotokollia kuten Profinetiä, Modbusia ja Ethernet/IP:tä. (Universal Robots 2019, 11, 85.)



Kuva 4. UR10-robotin käsiohjain.

#### 4.1.2 Käyttöliittymä ja ohjelmointi

UR10-robotin PolyScope-käyttöliittymän hallintaan käytetään ohjauskeskukseen liitettyä kosketusnäytöllistä käsiohjainta. Ohjelmoinnin helpottamiseksi käyttäjä voi myös liittää USB-liitännällä erillisiä osoitinlaitteita, kuten hiiren ja näppäimistön. PolyScopen ohjelman lukujärjestelmä on jaettu kolmeen eri tasoon: graafisen käyttöliittymän tasoon (GUI level eli Graphic User Interface level), script-tasoon (URScript level) ja alemman tason ohjaintensäätötasoon (C-API Level, joka pohjautuu C++-kieleen). (Universal Robots, [viitattu 19.03.2020].)

Robotin liikkeiden ohjelmointi on melko yksinkertaista. Robotti liikutetaan haluttuun sijaintiin, joka tallennetaan liikekäslynä robottiohjelmaan. Robottia voi liikuttaa lineaarisesti robotin maailman-, työkalukoordinaatiston ja käyttäjän opettaman kohdekoordinaatiston mukaan tai kääntämällä robotin yksittäisiä niveliä yksitellen. Robottia voi myös liikuttaa käsiohjauksella, kunhan vapaa-ajopainike robotin käsiohjaimessa on painettuna pohjaan.

Opetettujen liikekäskyjen lisäksi liikettä voidaan toteuttaa myös ohjelmallisesti siirtokäskeyillä. Siirtokäskyt ovat hyödyllisiä mm. lavauksessa ja konenäköön liittyvässä paikoituksessa. Jos kameralla esim. tunnistetaan työkappaleen siirtyneen tietyn määrän tiettyyn suuntaan, niin siirtokäskeyillä opetettuja liikekäskyjä voidaan siirtää vastaava määrä, jolloin ohjelman liikekäskyt siirtyvät samassa suhteessa. (Universal Robots 2019, 143-144, 171.)

## **4.2 Robotin tarttuja**

Työssä käytetään kappaleen poimintaa Robotiqin kaksisormista 2F-85-tarttujaa (kuva 5). Tarttujan tartuntaetäisyys voidaan ohjelmallisesti säätää 0 – 85 mm:n välille, ja hyötykuorma on 5 kg. Lisäksi tarttujan puristusvoima on mahdollista säätää 20 – 235 N:n välille. (Robotiq 2020.)



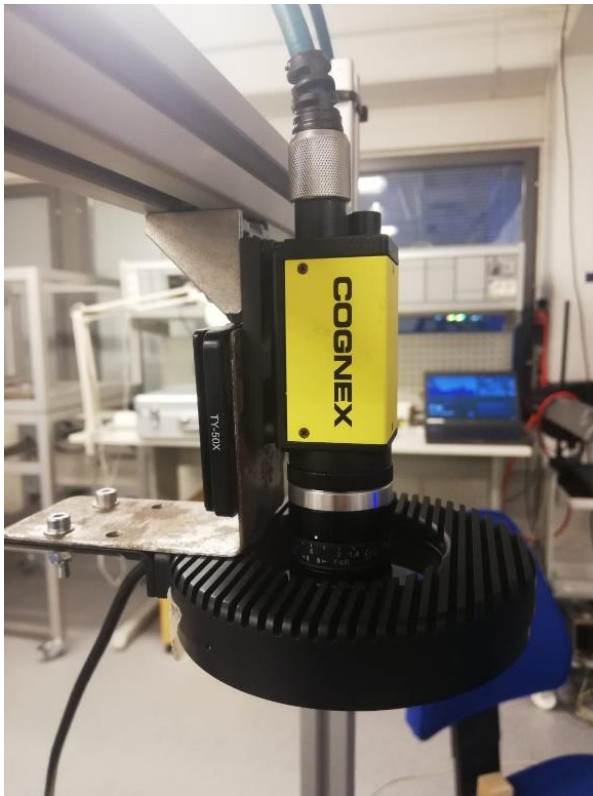
Kuva 5. Robotin tarttuja 2F-85.

### 4.3 Cognex Micro 1403 -älykamera

Työhön valittiin Cognex In-Sight Micro -sarjan 1403-mallin älykamera (kuva 6), koska koulun tarjoamista kameroista tämä kamera sopi parhaiten ominaisuuksiltaan haluttuun sovellukseen.

Cognexin Micro-sarjan älykamerat ovat pienikokoisia konenäkölaitteita, joita voidaan käyttää monissa eri tehtävissä, kuten tarkastelu-, mittaus- ja tunnistussovelluksissa sekä robotin ohjauksessa. Kameran kuvakennon resoluutio on 1600 x

1200, ja kamerassa käytetään 1/1.8 tuuman CCD-kennoa. Kameran ohjelmointiin käytetään In-Sight Explorer -ohjelmaa. (Cognex 2020a.)



Kuva 6. Kamera ja rengasvalaisin.

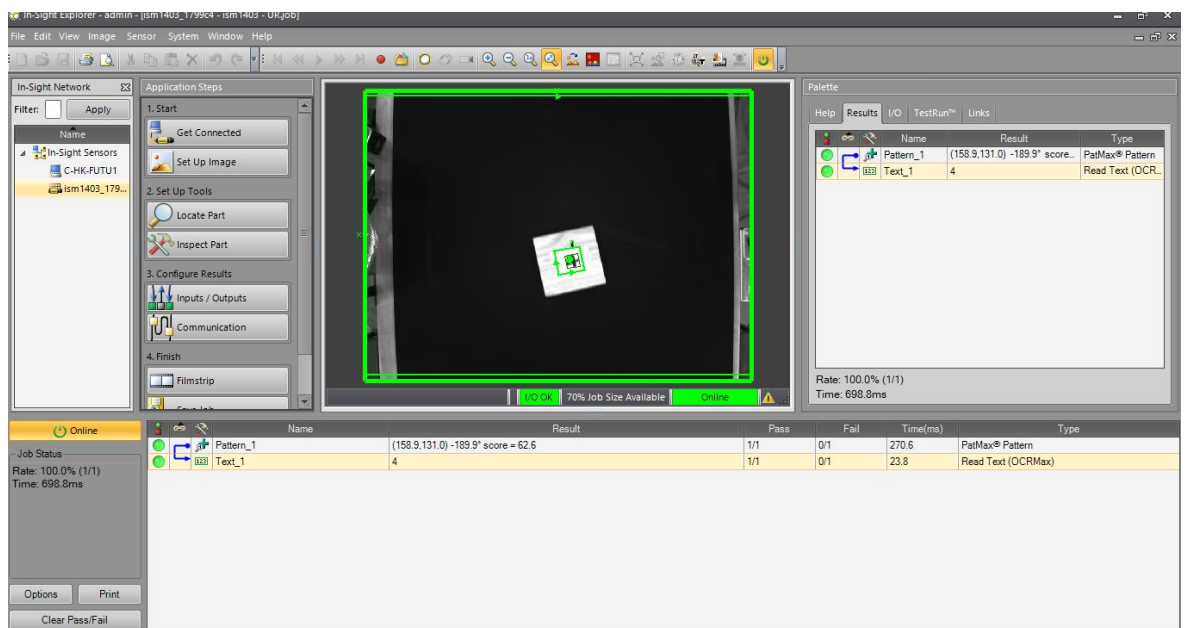
Kamera on yhdistettynä CIO-Micro I/O-moduuliin, jonka välityksellä käydään osa kameran ja robotin välisestä kommunikoimisesta, sekä ohjataan kameran kuvanottoa ja valaisimen valotusta. I/O-moduulissa on kahdeksan digitaalista tuloa ja -lähettä sekä erillinen liipaisutulo, jolla aktivoidaan kameran kuvanotto.

#### 4.4 Kameran optiikka

Kamerassa käytetään Fujifilmin valmistamaa objektiivia HF9HA-1B. Objektin polttoväli oli 9 mm, ja objektin tarkennus sekä kuvausaukon koko on manuaalisesti säädettävissä. Kuvausaukon suuruus ja valotusaika yhdessä määrittävät, kuinka paljon valoa pääsee kameran kennolle.

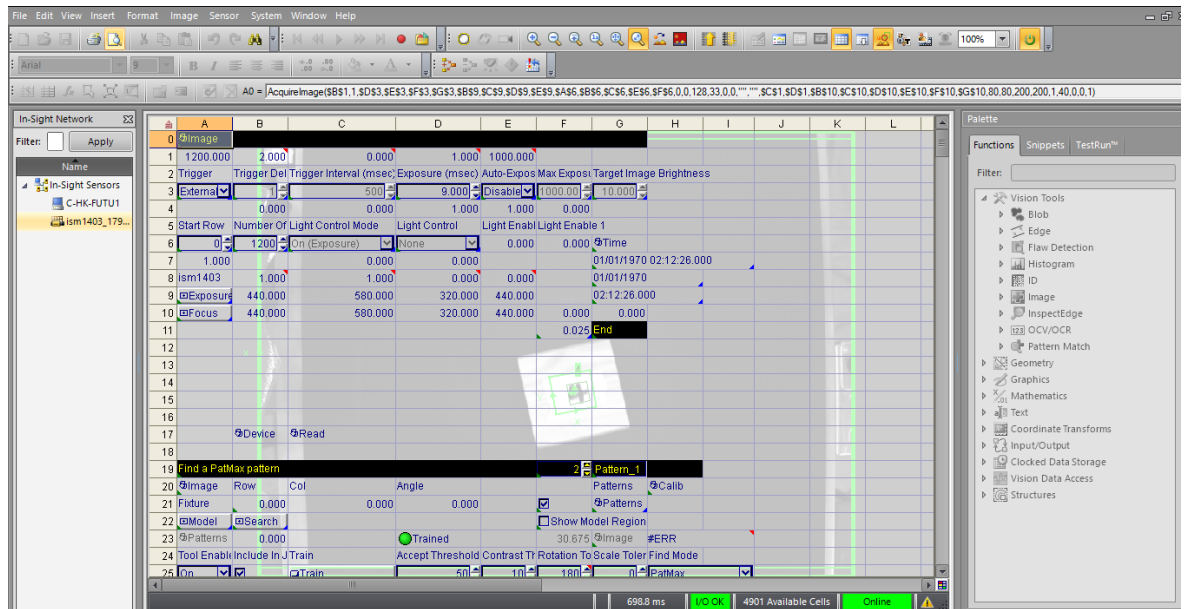
## 4.5 Konenäköohjelmisto In-Sight Explorer

Cognexin kameroiden ohjelmointiin käytetään In-Sight Explorer -ohjelmistoa. Tarkistusrutiinin voi luoda käyttämällä EasyBuilder- (kuva 7) tai Spreadsheet-ohjelmointitilaa (kuva 8). EasyBuilder-ohjelmointitila on käyttäjäystävällisempi, ja sen avulla konenäkösovelluksen toteuttaminen on helppoa ja nopeaa, kun taas Spreadsheet-ohjelmointitilassa ohjelmoijalla on yksityiskohtaisempi ja tarkempi kontrolli ohjelman toiminnasta. Spreadsheet-ohjelmointitila muistuttaa Microsoft Excel -taulukkolaskentaohjelmasta tuttua taulukkonäkymää. In-Sight Explorerilla on myös mahdollista tehdä konenäköohjelma offline-tilassa ilman kameraa esitallennettujen kuvien avulla. (Cognex 2020b.)



Kuva 7. In-Sight Explorerin Easybuilder-ohjelmointitila.





Kuva 8. In-Sight Explorerin Spreadsheet-ohjelmointitila.

## 4.6 Latab-rengasvalo

Työssä käytettiin kuvattavan kappaleen valaisuun Latabin SAX3 1102 -LED -rengasvalaisinta (kuva 6), joka kiinnitettiin kamera objektin ympärille. Rengasvalaisin ei ollut ehkä optimaalisin valaisin järjestelmään, mutta se oli ainoa tähän käyttötarkoitukseen soveltuva valaisin, mitä koululla oli tarjota, eikä työssä ollut budjettia laitehankintoihin.

## 4.7 Latab-valo-ohjain

Rengasvalaisimen valotuksen ohjaukseen käytettiin Latabin valo-ohjainta (Kuva 9). Ohjaimesta pystyi mm. säätämään valaistuksen intensiteettiä. Valo-ohjain oli liitetty konenäköjärjestelmän I/O-moduuliin lähtöön, jonka kautta valo-ohjain sai valotuksen liipaisupulssin.





Kuva 9. Latabin valo-ohjain.

#### 4.8 Liukuhihna

Järjestelmässä kappale poimitaan liukuhihnalta, jota ohjataan taajuusmuuntajalla. Liukuhihnaan on asennettuna Omronin inkrementaalipulssianturi E6C2-CWZ5B, joka on yhdistettynä robotin I/O-rajapintaan. Pulssianturin pulssiarvoa käytetään kappaleen paikoitukseen liukuhihnalta. Kappaleen paikoittamisen perehdytään tarkemmin työn seuraavassa osiossa.

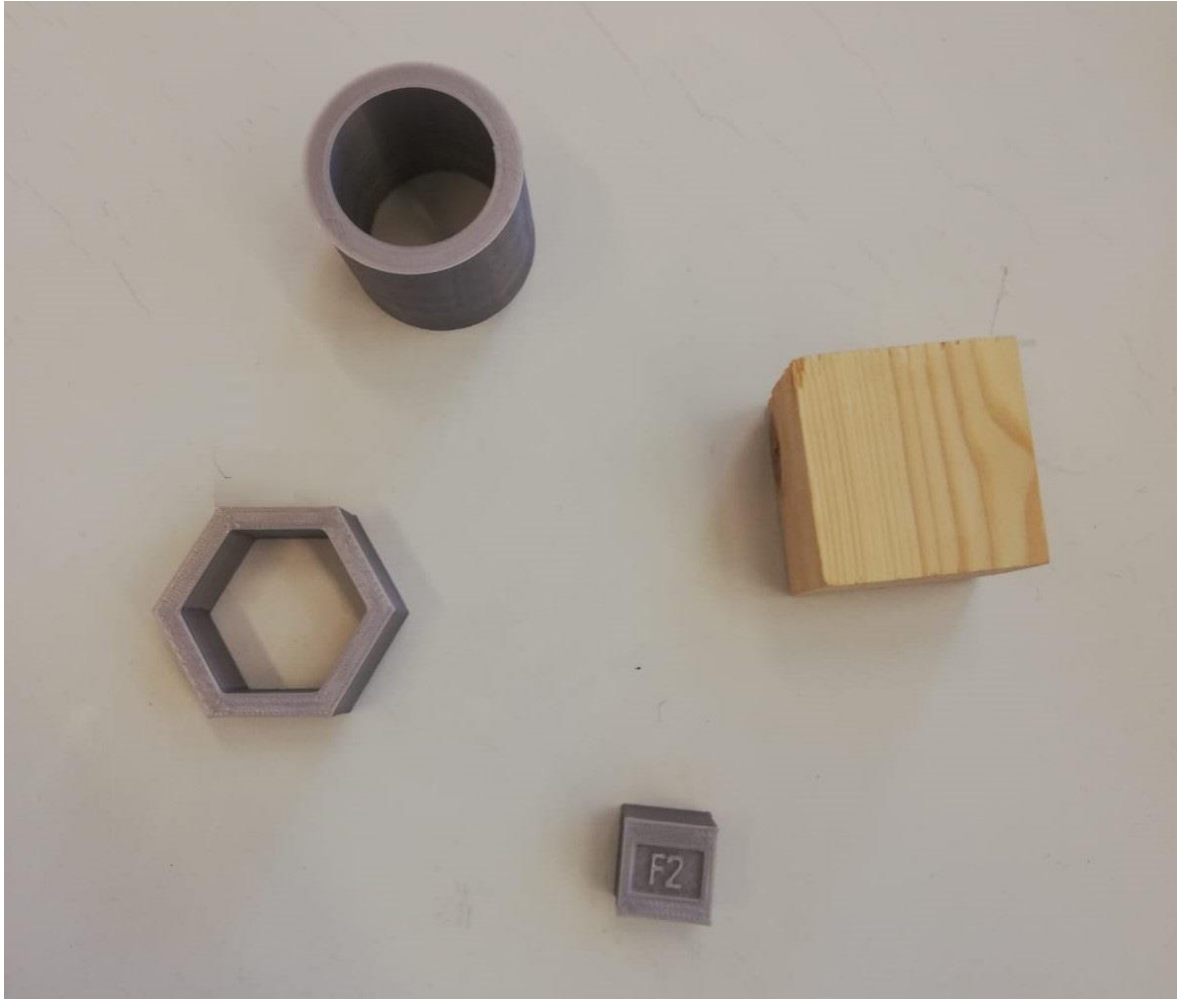
## 5 ROBOTTISOLUN SUUNNITTELU JA KÄYTÄNNÖN TOTEUTUS

### 5.1 Konenäköjärjestelmän tavoite

Konenäköjärjestelmän tavoitteena on tunnistaa kappaleesta tarvittavat piirteet, jotta kappale voidaan automaattisesti poimia robotilla. Kappaleen kuvaus sekä poiminta tulisi tapahtua liukuhihnalta suoraan liikkeestä. Konenäköohjelman tarkistusrutiinissa ensisijaisena tavoitteena on tunnistaa kohdekappaleen XY-koordinaatit sekä kiertymäkulma. Toissijaisena tavoitteena on tunnistaa yhdestä kappaleesta tekstimerkkijono, jonka perusteella järjestelmä päättää poimitaanko kappale vai ei. Paikoituksessa tulee lisäksi ottaa huomioon kameran kalibrointi, tunnistusalgoritmien vaikutus mittausten tarkkuuteen ja liukuhinnan liikkuminen.

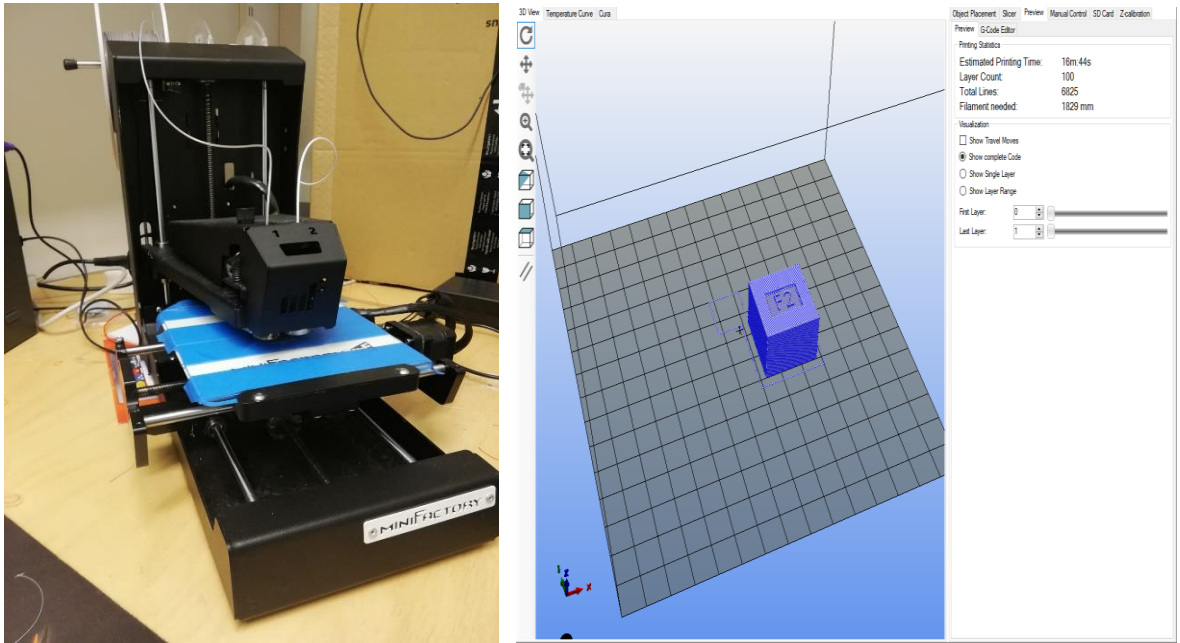
### 5.2 Kohdekappaleet

Työn alkuperäisenä kohdekappale oli yrityksen suunnitteilla olevaan projektiin liittyvä kappale, mutta projektin peruuntuessa alkuperäistä kappaletta ei voitu käyttää järjestelmän toiminnan havainnollistamiseen. Tämän vuoksi järjestelmän toiminnan havainnollistamiseksi täytyi valita muita kohdekappaleita. Kohdekappaleiden valintaa rajoitti robotin tarttujan 85 mm:n maksimitartuntaetäisyys. Siispä kohdekappaleiksi valittiin neljä erikokoista ja -muotoista kappaletta, jotka näkyvät kuvassa 10. Kappaleista täytyy tunnistaa XY-koordinaatit ja niiden kiertymäkulma, jotta ne voitaisiin onnistuneesti poimia robotilla. Alkuperäisestä kohdekappaleesta tunnistettiin myös tekstimerkkijono, jonka vuoksi yhteen kohdekappaleeseen (kuva 10 pienempi kuutio) lisättiin tekstimerkkijono, joka vastasi alkuperäisen kohdekappaleen tekstimerkkijonoa.



Kuva 10. Kohdekappaleet.

Puista kuutiota lukuun ottamatta kaikki kohdekappaleet tulostettiin MiniFactoryn 3D-tulostimella (kuva 11). Kappaleet mallinnettiin Solid Edge -mallinnusohjelmalla, joka on Siemensin tuottama ohjelma, jolla pystyy mallintamaan 2D- ja 3D-mallinnuksia. Mallinnuksen jälkeen CAD-tiedostot muutettiin STL-tiedostoiksi, jonka jälkeen ne voitiin ladata 3D-tulostimen ohjaukseen tarkoitettuun RepetierHost-ohjelmaan (kuva 11). Ohjelmassa STL-tiedostot muutetaan G-koodiksi, jota 3D-tulostin käyttää tulostuksen liikeratojen muodostamiseen.



Kuva 11. MiniFactoryn 3D-tulostin (vasemmalla) ja RepetierHost-ohjelman tulostuksen esikatselusivu (oikealla).

### 5.3 Kameran kuvausetäisyyden määrittäminen

Kamera päätettiin sijoittaa liukuhinnan päälle. Jotta saataisiin mahdollisimman tarkka kuva, pitää kameralle määrittää optimaalinen kuvausetäisyys. Parhaan kuvausetäisyyden saa selvitettyä kokeilemalla, mutta sen voi myös arvioida esim. Cognexin verkkosivulta löytyvällä laskurilla, joka laskee kuvausetäisyyden objektin polttovälin ja halutun kuva-alan pituuden ja leveyden avulla. Kameran kuva-alan pitäisi olla vähintään yhtä leveä kuin kuljetin, jotta kohdekappaleet voitaisiin paikoittaa liukuhinnalta jokaisesta kohdasta. Kuva-alueen leveydeksi mitattiin 250 mm ja kuva-alan pituuden ohjelma määrittää automaattisesti. Työssä käytettävän objektin polttoväli oli taas 9 mm. Näillä arvoilla saatiin laskettua optimaaliseksi kuvausetäisyydeksi n. 426 mm (taulukko 1).

## RESULTS

	Focal Length (mm)	Working Distance (mm)	Width (mm)	Height (mm)
<b>Closest Higher</b>	10.0	473.485	333.333	250.000
<b>Exact</b>	9.0	426.136	333.333	250.000
<b>Closest Lower</b>	8.0	378.788	333.333	250.000

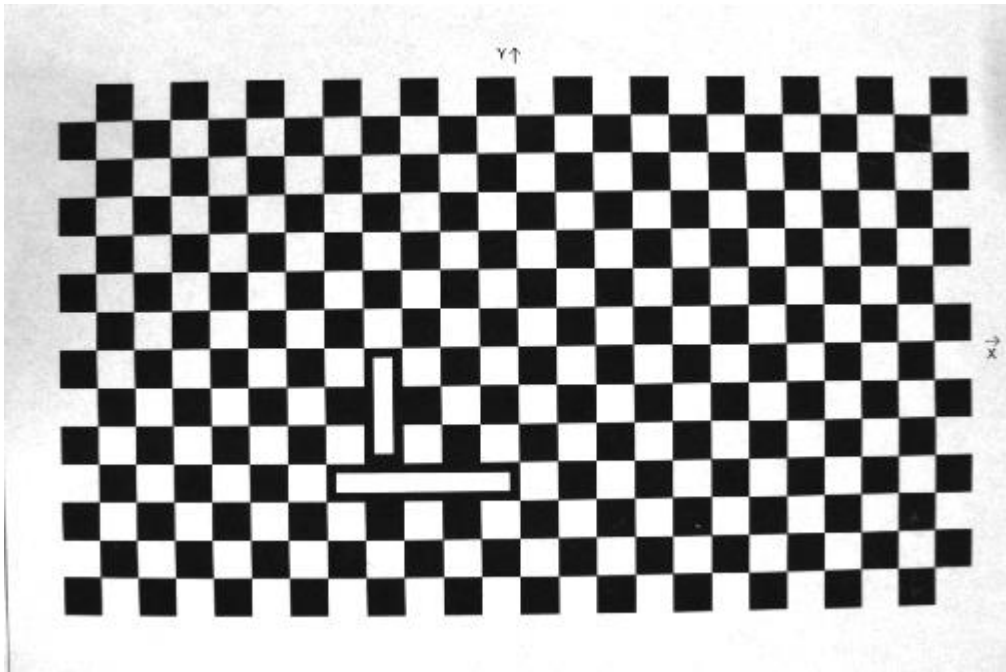
Taulukko 1. Optimaalisen kuvausetäisyyden laskevan laskurin tulos (Cognex 2020c).

### 5.4 Valaistus

Projekti tehtiin SeAMKin robotiikan laboratoriossa, jossa huoneen valaistukseen käytettiin loisteputkivalaistusta. Auringonvalon pääsy laboratorioon oli enimmiltä osin estetty ikkunan sädekaihtimilla, jolloin valaistusympäristön muutokset olivat vähäisiä. Pelkästään huoneen valaistus ei kuitenkaan riittänyt tarvittavan kohdekapaleiden haluttujen piirteiden tunnistamiseen. Siispä kohteen valaistukseen päätettiin käyttää Latabin rengasvalaisinta, joka sijoitettiin kameran objektin ympärille. Rengasvalaisimella saatiin luotua kohtisuoran valaistuksen valaistusgeometria, joka loi riittävän yleisvalaistuksen tarvittavien piirteiden korostamiseksi. Koska kappaleen kuvaus haluttiin suorittaa liikkeestä, valotukseen käytettiin strobo- eli salama-valaisua.

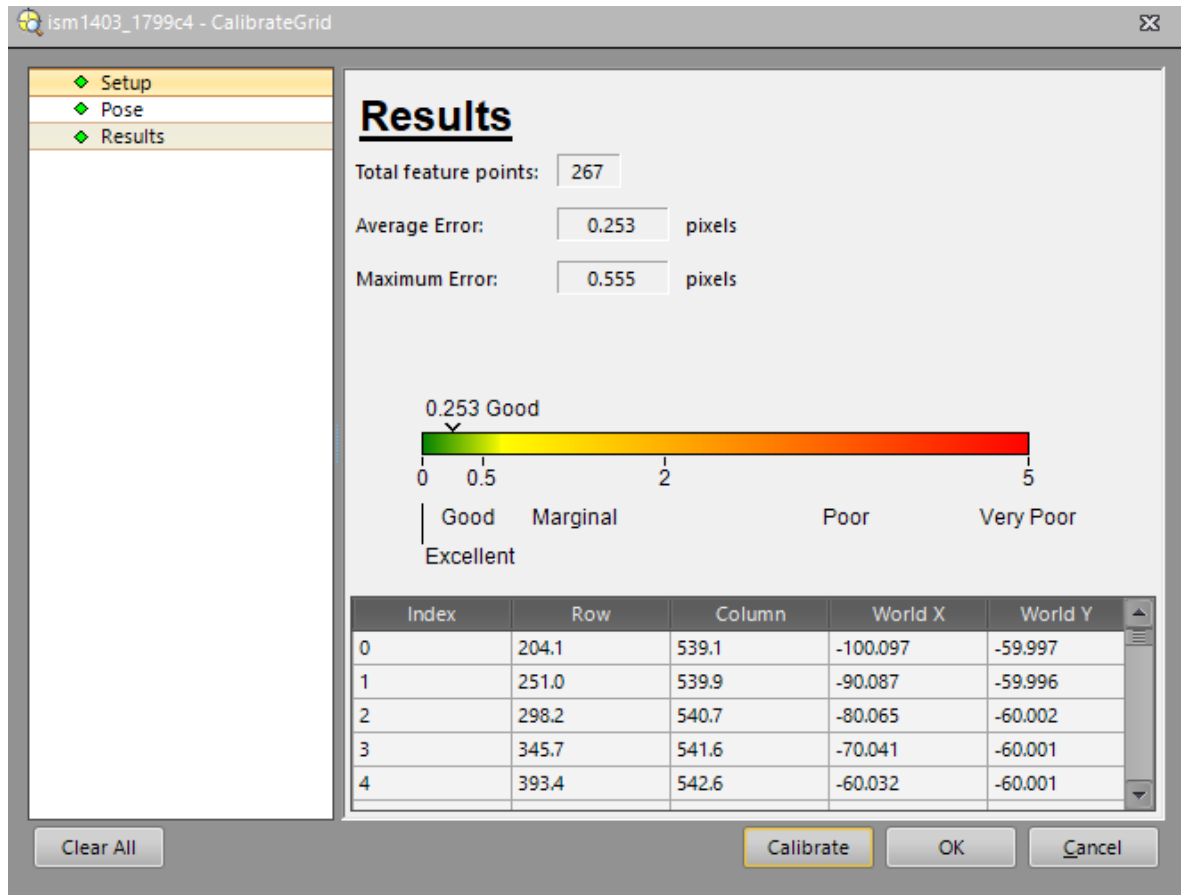
### 5.5 Kameran kalibrointi

Kameran sisäinen suuntaus tehtiin skannaamalla kalibrointilevy (kuva 12) In-Sight Explorerin ohjelman kalibrointityökalulla. Kalibrointityökalulle pitää määrittää kalibrointilevyn neliöiden koko tai pisteiden etäisyys, minkä perusteella se osaa tehdä pikseliarvojen muunnoksen reaali maailman arvoiksi.



Kuva 12. Kalibrintilevy.

Kalibrintilevyn keskellä olevan kuvion perusteella määritty kamerakoordinaatiston origo, joka määrittää koordinaatin X- ja Y-akselien suunnat. Lopuksi kalibrointityökalu kertoo tuloksen, miten tarkasti kalibrointi saatiin suoritettua (kuva 13). Kalibroinnin onnistumiseen vaikuttaa monta tekijää, kuten valittu optiikka ja valaistuksen muutokset. Siispä kalibroinnin tulosta saadaan parannettua säätämällä valotusaikaa sekä objektiivin aukon kokoa ja tarkennusta.



Kuva 13. In-Sight Explorerin kalibrointityökalun arvio kalibroinnin onnistumisesta.

Aluksi suunnitelmana ulkoisen suuntauksen suorittamiseksi oli käyttää In-Sight Explorerin Robot calibration -kalibrointityökalulla, joka on nimenomaan tehty robotin ja kameran väliseen ulkoiseen suuntaukseen. Kamera oli kuitenkin sen verran vanha, ettei se tukenut kyseistä kalibrointityökalua. Siispä ulkoinen suuntaus toteutettiin yksinkertaisesti määrittämällä yhteinen koordinaatiopiste UR10:n maailmankoordinaatistolle ja sisäisesti suunnatun kamerakoordinaatiston välille. Pisteeksi valittiin kalibroinnin yhteydessä määritetty kamerakoordinaatiston origo, joka toimii kamerakoordinaatiston nollapisteinä. Piste saatiin määritettyä robotille viemällä robotin työkalun TCP-piste origon kohdalle ja tallentamalla se robotin paikkamuuttujaksi. Tallennetun paikkamuuttujan X- ja Y-akseli määritettiin saman suuntaiseksi kamerakoordinaatiston X- ja Y-akselin kanssa. Tätä paikkamuuttujaa robotti käyttää liikekäskeyden koordinaatistopisteinä.

## 5.6 Kappaleen paikoitus liukuhihnalta

Jotta kappale voitaisiin poimia liikkeestä liukuhihnalta eri paikasta kuin kamera on sen kuvannut, pitää liukuhihnan liikkeitä saada mitattua. Tämän takia kuljettimeen asennettiin Omronin E6C2-CWZ5B-inkrementaalipulssianturi, joka yhdistettiin robotin I/O-rajapinnan digitaaliseen tuloon. Inkrementaalianturin antaa pulssiarvon liukuhihnan liikkeiden mukaisesti, tämän avulla voidaan määrittää, kuinka monta pulssia anturi antaa suhteessa haluttuun välimatkaan, esim. pulssia per metri. Tämän saa laskettua anturin erottelukyvyn ja pulssianturin kontaktilevyn säteen avulla kaavalla:

$$\text{Pulssia per metri} = \frac{500}{2 \cdot \pi \cdot 0.025} \approx 3183 \text{ p/m} \quad (1)$$

Kaavassa luku 500 on anturin erottelukyky ja luku 0,0025 on pulssianturin kontaktilevyn säde metreinä. Vastaukseksi saatiin 3183, joka kertoo, kuinka monta pulssia anturi antaa liukuhihnan liikuessa metrin pituisen matkan.

## 5.7 Tiedonsiirto kameran ja robotin välillä

Konenäkökameran ja rengasvalaisimen aktivointiin käytetään kameran I/O-moduulia, jonka liipaisemiseen käytetään robotin ohjaamaan digitaalista tuloa. Konenäköjärjestelmän tulokset välitetään robotille Ethernet-väylällä. Kameraan ohjelmoitiin TCP/IP-protokollaa käyttävä palvelin (server), johon robotti toimi asiakkaana (client). Kun kameran tarkistusrutiini tunnistaa halutun kappaleen, muodostaa kamera ASCII-muotoisen merkkijonon saaduista arvoista ja lähettää sen robotille.

## 5.8 Konenäköohjelman tarkistusrutiini

Kappaleen tarkistusrutiini ohjelmoitiin In-Sight Explorer -ohjelmalla ja sen ohjelmointiin käytettiin enimmäkseen Easybuilder-ohjelmointitilaa. Robotti poimii kappaleita liukuhihnalta, jossa kohdekappaleiden sijainti on satunnainen. Tämän vuoksi kohdekappaleista täytyi saada selville niiden XY-koordinaatit sekä kiertymäkulma.



Nämä tiedot saadaan mitattua In-Sight Explorer -ohjelman pattern-paikannustyökälulla, jonka algoritmi vertailee kuvattavaa kappaletta etukäteen opetettuun malliin. Yhdestä tunnistettavasta kappaleesta pitää myös tunnistaa tekstimerkkijono. Tekstimerkkijono tunnistamiseen käytetään Read Text -tunnistustyökälua. Read Text -tunnistustyökälu käyttää tunnistamiseen algoritmia, joka vertaa kuvattavaa tekstimerkkijonoa etukäteen opetettuun tekstimerkkimalliin.

## **5.9 Robotin liikkeiden ohjelmointi**

Robotin poimintasekvenssin liikkeet ohjelmoitiin siirtokäskyllä, jonka suunnan ja matkan määritti kameran paikoituksen parametrit, kameran koordinaatiston nolla-kohtaan määritetty paikkamuuttuja, sekä pulssianturin pulssiarvo.

## **5.10 Robotin ohjelma ja järjestelmän toimintasekvenssi**

UR10-robotin PolyScope-ohjelma (Katso liite 1) koostuu kolmesta osasta: alustuksesta (Before Start), säieohjelmasta (Thread) ja pääohjelmasta (Robot Program). Alustuksessa määritellään muuttujat sekä muodostetaan yhteys kameraan. Yhteyden muodostamiseksi kamera ja robotti lähettävät toisilleen ns. kättelysignaalit. Jos yhteyden muodostaminen onnistuu, ohjelma siirtyy pääohjelmaan. Säieohjelma toimii pääohjelman kanssa samanaikaisesti, ja sen tehtävä on ohjata kameran kuvanottoa ja valaisimen valotusta. Säieohjelma käynnistyy kameran alle sijoitetun optisen anturin aktivoituessa, jonka jälkeen robotin digitaalinen lähtö lähettää kameran I/O-moduulille liipaisupulssin, joka aktivoi kameran kuvanoton sekä valaisimen valotuksen.

Pääohjelmassa suoritetaan kappaleen poimintasekvenssi. Pääohjelma alkaa siitä, että robotti siirtyy poiminnan aloittamispisteeseen ja jää odottamaan kameralta poimintalupaa sekä tarvittavia parametreja. Jos kameran tarkistusrutiini on onnistunut, kamera lähettää parametrit sisältävän merkkijonon robotille. Näistä parametreista robotti saa tiedon, montako prosenttia kappale vastasi alkuperäisesti opetettua kappaletta, kappaleen XY-koordinaatit, kiertymäkulman, ja jos kappaleesta tunnistetaan tekstimerkkijonoa, niin tieto siitä, oliko halutut merkit onnistuttu tunnistamaan.

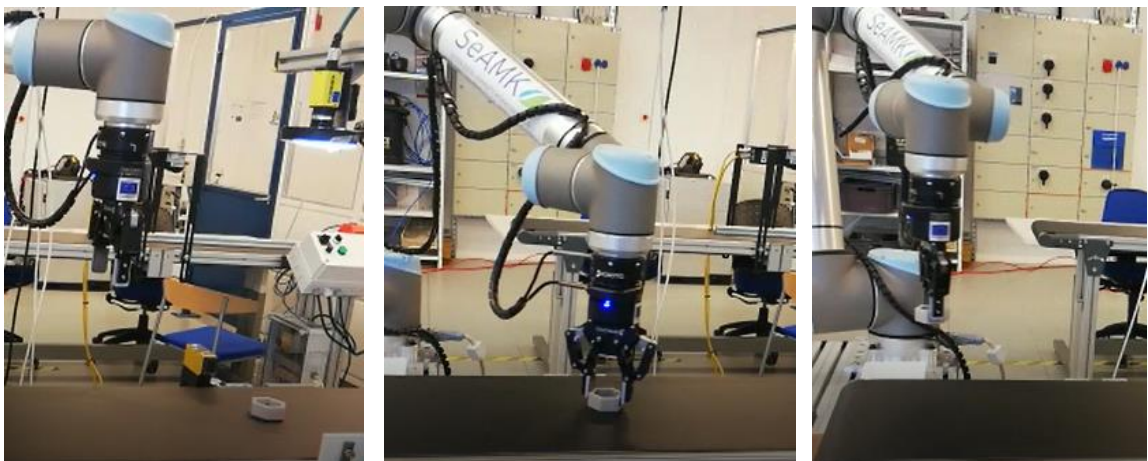
Tämän jälkeen suoritetaan kappaleen poiminta. Kun robotti saa kameran parametrit ja kappale vastaa vähintään 80 % alkuperäistä kappaletta, siirtyy se välittömästi poimimaan kappaleen. Kappale poimitaan kuljettimelta suoraan liikkeestä, jonka jälkeen kappale siirretään tiputuspisteeseen. Jos kappaleesta tunnistettiin tekstimerkkijonoa, ja tekstimerkkijono oli väärä tai sitä ei löytynyt ollenkaan, robotti ei siirry poimimaan kappaletta.

## 6 TULOKSET

Työn tulokseksi saatiin tehtyä pääsääntöisesti tavoitteiden mukainen järjestelmä, jolla saatiin poimittua kaikki testikappaleet luotettavalla toistettavuudella. Tulos ei riippunut kappaleen sijainnista tai sen kiertymäkulmasta. Yhden testikappaleen tekstimerkkijonon tunnistamisessa kuitenkin ilmeni hieman ongelmia. Tätä järjestelmässä tulisi tulevaisuudessa kehittää, jos sitä haluttaisiin käyttää teollisessa ympäristössä.

### 6.1 Kappaleen paikoitus ja poiminta

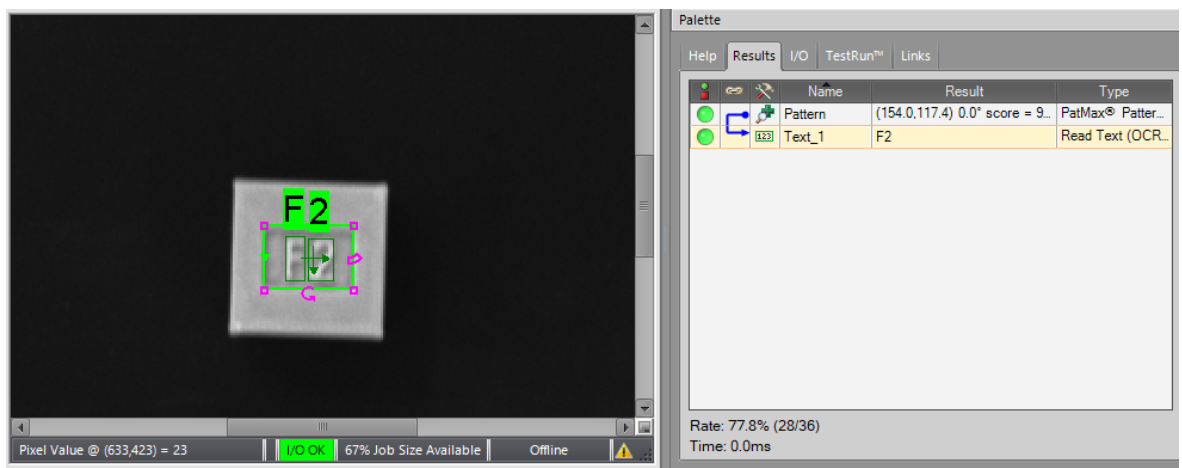
Järjestelmän toimintaa testattiin neljällä erilaisella kohdekappaleella. Kappaleiden paikoittaminen onnistui hyvin ja kappaleet saatiin paikoitettua luotettavalla toistettavuudella. Kappaleiden poimintaa testattiin eri kiertymäkulmista ja kuva-alan eri sijainneista n. 30 kertaa per kappale. Näiden testien aikana ei tullut yhtään virheellistä poimintaa tai tarkistusrutiinia. Kuva 14 havainnollistaa robotin liikkeitä kappaleen poiminnassa.



Kuva 14. Kuvasarja kappaleen poiminnasta.

## 6.2 Tekstimerkkijonon tunnistus

Yhdestä kohdekappaleesta tunnistettiin myös kahden merkin tekstimerkkijono (kuva 15), jonka tunnistuksessa ilmeni hieman ongelmia. Tekstimerkkijonon tunnistusta testattiin n. 70 kertaa ja tekstimerkkijono saatiin tunnistettua pääsääntöisesti hyvin, mutta satunnaisesti ohjelma ei onnistunut tunnistamaan tekstimerkkijonoa oikein. Tekstimerkkijono saatiin tunnistettua luotettavasti sen ollessa suurin piirtein samassa kiertymäkulmassa, jossa teksti oli alkuperäisesti järjestelmälle opetettu. Tekstimerkkijonon tunnistuksista kiertymäkulmista 45–135 asteen välillä sekä 225–315 asteen välillä n. yksi viidestä tunnistuksesta epäonnistui. Valitettavasti ulkoisten tekijöiden vuoksi ei ollut mahdollista käyttää enempää aikaa järjestelmän testaamiseen, ja testien määrä jäi hieman suppeaksi.

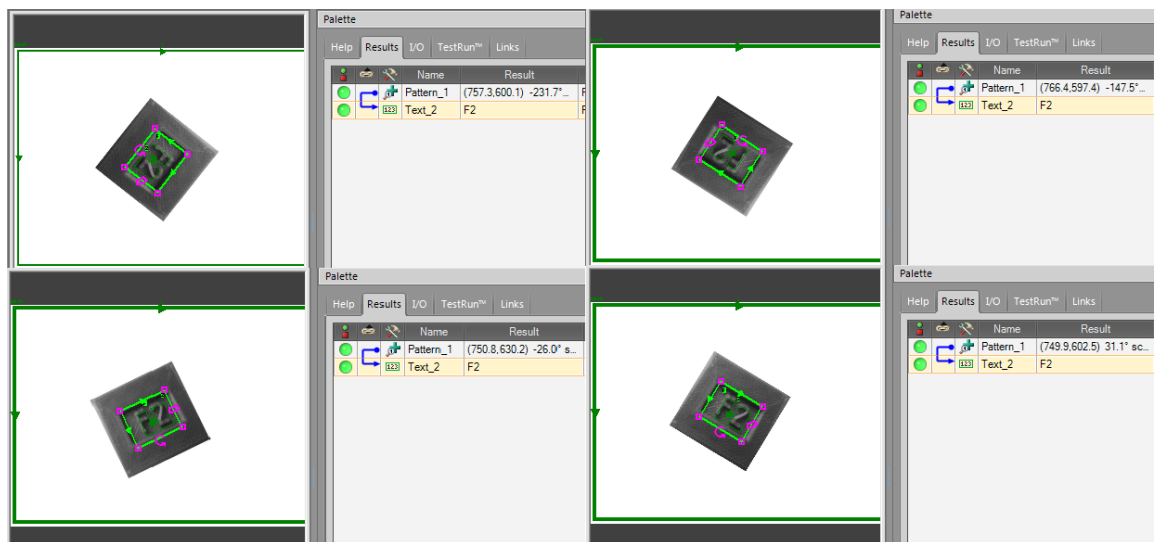


Kuva 15. Onnistunut tekstimerkkijonon tunnistus (kuvaa on zoomattu).

Tunnistusalgoritmin asetusarvojen hienosäädöllä olisi voinut mahdollisesti parantaa tunnistuksen luotettavuutta, mutta kuvausympäristössä oli myös monta tekijää, jotka mahdollisesti vaikuttivat tekstimerkkijonon epäonnistuneisiin tunnistuksiin: kuvausetäisyys ei ollut optimaalinen noin pienen tekstimerkkijonon tunnistamiseen huomioon ottaen käytetyn kamerasen resoluution. Tekstimerkkijonon materiaali ja väri oli sama tekstimerkkijonon taustan kanssa, mikä hieman hankaloitti ihanteellisen kontrastin luomista tekstimerkkijonon ja taustan välille. Käytetty valaistusgeometria ei myöskään ollut optimaalisin kappaleeseen kaiverrettuun kuvion tunnistamiseen. Siispä päätettiin kokeilla tekstimerkkijonon tunnistamista optimaalisemmassa ympäristössä, jossa kuvausetäisyyttä pienennettiin ja kuvista saatiin tarkempia. Tällöin

saatiin selvitettyä, johtuivatko epäonnistuneet tunnistukset kuvan analysointiohjelman algoritmista tai sen vääristä asetusarvoista vai konenäköjärjestelmän komponenttien riittämättömyydestä tuottaa tarpeeksi tarkka kuva järjestelmän testausympäristössä.

Tekstimerkkijonon tunnistusta testattiin n. 100 mm:n kuvausetäisyydellä 20 eri kiertymäkulmasta (kuva 16). Kuvauksessa käytettiin vähäisempää valaistusta ja kuvanoton aikana kappale ei ollut liikkeessä. Tällöin tekstimerkkijono saatiin tunnistettua oikein jokaisella kerralla. Tästä voidaan päätellä, että epäonnistuneet tunnistukset johtuivat siitä, ettei kyseisellä järjestelmällä saatu kuvattua kappaletta riittävän tarkasti, jotta kappaleen tekstimerkkijono olisi voitu tunnistaa luotettavasti eri kiertymäkulmista. Tulevaisuudessa ongelma voidaan ratkaista esim. hankkimalla korkeammalla resoluutiolla varustettu kamera tai korvaamalla kamera kahdella kameralla, jolloin kuvausetäisyyttä saataisiin laskettua pienemmäksi ja kuva-ala pysyisi samana. Myös kuvausgeometriaksi voitaisiin vaihtaa esim. aksiaalinen diffuusivalaistus, joka soveltuisi paremmin pinnan muutoksien tarkasteluun.



Kuva 16. Tekstimerkkijonon tunnistus pienemmällä kuvausetäisyydellä.

## 7 POHDINTA JA YHTEENVETO

Työn tavoitteena oli suunnitella ja toteuttaa konenäköjärjestelmä, jonka avulla voidaan paikoittaa liukuhihnalta kappaleita ja poimia ne automaattisesti liikkeestä robotilla. Työn tavoitteet ja edellytykset tulivat suurimmilta osin täytetyiksi ja tulokseksi saatiin toteutettua toimiva järjestelmä halutuilla ominaisuuksilla. Alun perin järjestelmällä oli tarkoitus tunnistaa ja poimia kappale, jota käytettäisiin yrityksen suunnitella olevassa projektissa. Projekti kuitenkin peruuntui, eikä alkuperäistä kohdekappaleita pystytty käyttämään projektissa. Järjestelmän toimivuuden testaamiseksi valittiin neljä erikokoista ja -muotoista kohdekappaleita. Jokaisen kohdekappaleen poimintaa testattiin n. 30 kertaa per kappale ja kohdekappaleet saatiin poimittua järjestelmällä luotettavalla toistettavuudella riippumatta kappaleiden sijainnista liukuhihnalla tai kappaleen kiertymäkulmasta.

Tavoitteena oli myös tunnistaa yhdestä kohdekappaleesta tekstimerkkijono eri kiertymäkulmista. Tämä onnistui kohtalaisella toistettavuudella. Tekstimerkkijono saatiin tunnistettua luotettavasti kappaleen ollessa suurin piirtein samassa kiertymäkulmassa, jossa kappale oli järjestelmällä opetettu, mutta kiertymäkulman vaihtuessa n. yli 45 astetta n. yksi viidestä tunnistuksesta epäonnistui. Epäonnistuneiden tunnistusten pääteltiin johtuneen järjestelmän riittämättömyydestä tuottaa tarpeeksi tarkka kuva konenäköohjelman algoritmille.

Jotta kuvasta saataisiin halutulla kuva-alalla tarpeeksi tarkka tekstimerkkijonon tunnistamiseksi, täytyisi esim. järjestelmän kamera korvata korkeamman resoluution omaavalla kameralla tai saman resoluution omaavalla kahdella kameralla, jolloin kuvausetäisyyttä voitaisiin pienentää, ja silti saada kuvattua sama kuva-ala.

Koska kahta tai korkeamman resoluution omaavaa kameraa ei ollut saatavilla, päätettiin tekstimerkkijonon tunnistamista testata pienemmällä kuvausetäisyydellä. Tällöin kappale saatiin tunnistettua luotettavalla toistettavuudella. Tämä ei kuitenkaan lopullisena ratkaisuna sopinut sovellukseen, koska pidempi kuvausetäisyys oli välttämätön, jotta saatiin koko kuva-ala kuvattua. Tällä saatiin kuitenkin varmistettua, että haluttu järjestelmä on tulevaisuudessa mahdollista toteuttaa muuttamalla järjestelmän laitteistoa tai ympäristöä, esim. kuljettimen leveyttä.

Työn aikana ilmeni myös muita haasteita. Koulu jouduttiin sulkemaan ulkoisten tekijöiden vuoksi, jolloin järjestelmän testaamiseen jouduttiin käyttämään vähemmän aikaa, mitä alkuperäisesti oli suunniteltu. Myös työssä käytetyn konenäköjärjestelmän kamera oli melko vanha ja siitä puuttui ominaisuuksia, joita uudemmissa kameroissa löytyisi. Tämän vuoksi työssä jouduttiin käyttämään eri ratkaisua kameran ja robotin välisen tiedonsiirtoon, robottisolun kalibroimiseen sekä robotin liikkeen ohjaukseen kuin alkuperäisesti työssä oli suunniteltu käytettäväksi. Tämä ei kuitenkaan vaikuttanut järjestelmän toimintaan, vaan teki sen rakentamisesta hieman haastavampaa. Myös tarttujan maksimitartuntaetäisyys rajoitti paljon sitä, minkälaisilla kohdekappaleilla järjestelmän toimintaa voitiin testata. Isompien kappaleiden poiminta olisi muuten ollut mahdollista.

Koska järjestelmää ei tehty oikeaan käyttöön ja aikataulu oli varsin kiireellinen, jäi myös toimintavarmuuden ja paikoitustarkkuuden tarkempi testaaminen, sekä lopullinen hienosäätö tekemättä. Jos järjestelmä tehtäisiin oikeaan käyttöön, pitäisi mm. kappaleiden virtausta suunnitella tarkemmin, esim. kuinka nopeasti järjestelmä pysyisi kappaleita poimimaan ja tarvittaisiinko esim. jonkinlaista puskuria säätelämään kuljettimelle tulevien kappaleiden määrää.

Valmis työ luovutettiin Robia Oy:lle ja valmiiden suunnitelmien pohjalta Robia Oy:lla on mahdollisuus toteuttaa konenäköjärjestelmä tai hyödyntää muuten työtä tulevaisuuden projekteissa.

## LÄHTEET

- Adaptive Vision. Ei päiväystä. Calibration of world coordinates. [Verkkosivu]. Adaptive Vision. [Viitattu 16.03.2020]. Saatavissa [https://docs.adaptive-vision.com/4.7/studio/machine\\_vision\\_guide/CalibrationWorldCoordinates.html](https://docs.adaptive-vision.com/4.7/studio/machine_vision_guide/CalibrationWorldCoordinates.html)
- Bélanger-Barrette, M. 2015. What Does Collaborative Robot Mean. [www-dokumentti]. [Viitattu 16.01.2020]. Saatavilla: <https://blog.robotiq.com/what-does-collaborative-robot-mean>
- Cognex. 2020a. In-Sight Micro Vision System Specifications. [www-dokumentti]. Cognex Corporation. [Viitattu 16.03.2020]. Saatavilla: [http://www.cognex.com/support/downloads/ns/1/11/35/Micro\\_all\\_ss.pdf](http://www.cognex.com/support/downloads/ns/1/11/35/Micro_all_ss.pdf)
- Cognex. 2020b. In-Sight Support. [Verkkosivu]. Cognex Corporation. [Viitattu 16.03.2020]. Saatavilla: [https://support.cognex.com/en-hu/downloads/in-sight/training/manuals\\_2019](https://support.cognex.com/en-hu/downloads/in-sight/training/manuals_2019)
- Cognex. 2020c. Lens advisor. [Verkkosivu]. Cognex Corporation. [Viitattu 16.03.2020]. Saatavilla: <https://www.cognex.com/resources/interactive-tools/lens-advisor>
- Cognex. 2016. Introduction to machine vision. [Verkkosivu]. Cognex Corporation. [Viitattu 16.03.2020] Saatavilla: <https://www.cognex.com/what-is/machine-vision/what-is-machine-vision>
- Digi-Key. 2012. Versatile LEDs Drive Machine Vision in Automated Manufacture. [Verkkosivu]. Digi-Key Electronics. [Viitattu 18.03.2020]. Saatavilla: <https://www.digikey.com/en/articles/techzone/2012/jan/versatile-leds-drive-machine-vision-in-automated-manufacture>
- Halinen, M. 2007. Konenäkö robotin ohjauksessa. [www-dokumentti] Aalto-yliopisto. [Viitattu 22.05.2014]. Saatavissa: [http://automation.tkk.fi/attach/AS-0-2230/lab3c\\_teoria.pdf](http://automation.tkk.fi/attach/AS-0-2230/lab3c_teoria.pdf)
- Hornberg, A. 2006. Handbook of machine vision. Weinheim: Wiley-VCH.
- IFR. 2018. Robottiikan taustaselvityksiä. [www-dokumentti]. International Federation of Robotics. [Viitattu 16.01.2020]. Saatavissa: [https://ifr.org/downloads/papers/IFR\\_Demystifying\\_Collaborative\\_Robots.pdf](https://ifr.org/downloads/papers/IFR_Demystifying_Collaborative_Robots.pdf)
- Kuivanen, R. 1999. Robottiikka. Suomen robottiikkayhdistys. Helsinki: Talentum.



- Kämäräinen, J. 1999. Kuvan muodostus ja esikäsittely. [Verkkosivu]. Lappeenranta University Of Technology [Viitattu 16.01.2020]. Saatavissa: <https://www.it.lut.fi/project/VQCCD/thesishtml/node11.html>
- Leino, M. Valo, P. & Kortelainen, J. 2014a. Konenäkötekniikan kustannustehokas hyödyntäminen. Teoksessa: M. Leino Teknologiatiedolla tuottavuutta: ammattikorkeakoulut kansainvälisen teknologiatiedon tulkkina pk-yrityksille. Pori: Satakunnan ammattikorkeakoulu, 21 - 24.
- Leino, M. Kortelainen, J. & Valo, P. 2014b. Valaistus – ratkaiseva osa konenäköjärjestelmää. Teoksessa: M. Leino Teknologiatiedolla tuottavuutta: ammattikorkeakoulut kansainvälisen teknologiatiedon tulkkina pk-yrityksille. Pori: Satakunnan ammattikorkeakoulu, 25 - 32.
- MicroScan. Ei Päiväystä. [Verkkosivu]. MicroScan. [Viitattu 16.03.2020]. Saatavissa: [http://files.microscan.com/helpfiles/visionscapetools\\_help\\_file/visionscape\\_tools\\_reference-04-1.html](http://files.microscan.com/helpfiles/visionscapetools_help_file/visionscape_tools_reference-04-1.html)
- National Instruments. 2019. A Practical guide to machine vision lighting. [Verkkosivu]. National Instruments. [Viitattu 16.01.2020]. Saatavilla: <https://www.ni.com/fi-fi/innovations/white-papers/12/a-practical-guide-to-machine-vision-lighting.html>
- Omron. 2020. X-Delta 3+1. [Verkkosivu]. Omron Electronics Oy. [Viitattu 16.03.2020]. Saatavilla: <https://industrial.omron.eu/en/products/x-delta-3+1>
- Pakkanen, J. & Koukkari, T. 2016. Robotiikka, Konenäkö, Mobiili robotiikka. [www-dokumentti]. Seinäjoen ammattikorkeakoulu. [Viitattu 16.01.2020]. Saatavissa: <https://storage.googleapis.com/seamk-production/2017/06/robotiikka.pdf>
- Pietikäinen, M. & Silven, O. Ei päiväystä. Konenäkö. [www-dokumentti]. Oulun yliopisto. [Viitattu 16.01.2020]. Saatavissa: <https://www oulu.fi/sites/default/files/content/konenako.pdf>
- Robia. 2020. [Verkkosivu]. Robia Oy. [Viitattu 18.03.2020]. Saatavissa: <https://www.robia.fi/>
- Robotiq. 2020. 2F-85 and 2F-140 Grippers. [Verkkosivu]. Robotiq [Viitattu 16.01.2020]. Saatavilla: [https://robotiq.com/products/2f85-140-adaptive-robot-gripper?ref=nav\\_product\\_new\\_button](https://robotiq.com/products/2f85-140-adaptive-robot-gripper?ref=nav_product_new_button)
- Technavion. 2018. 6 Major Types of Industrial Robots Used in the Global Manufacturing 2018. [Verkkosivu]. [Viitattu 05.03.2020]. Saatavissa: <https://blog.technavio.com/blog/major-types-of-industrial-robots>
- Universal Robots. 2019. User manual UR10/CB3 Versio 3.12.

Universal Robots. Ei päiväystä. FAQ [Verkkosivu]. Universal Robots. [Viitattu 19.03.2020]. Saatavissa: <https://www.universal-robots.com/how-tos-and-faqs/faq/>

Vision Doctor. 2020a. Lightning for industrial machine vision. [Verkkosivu]. Vision Doctor. [Viitattu 16.01.2020]. Saatavissa: <https://www.vision-doctor.com/en/lighting-for-machine-vision.html>

Vision Doctor. 2020b. Optic distortion. [Verkkosivu]. Vision Doctor. [Viitattu 20.04.2020]. Saatavissa: <https://www.vision-doctor.com/en/optical-errors/distortion.html>

## LIITTEET

### Liite 1. UR10 ohjelmakoodi

#### Program

BeforeStart

Gripper Activate

Call InitialData

Call InitialConnect

#### Robot Program

Call DoPrepare

Call GetCameraData

Call GoNewPosition\_Text

'Call GoNewPosition'

Call ClearAllData

#### Thread\_1

Wait sensor=HI

Set Camera\_trigger=On

Wait: 0.1

Set Camera\_trigger=Off

Permission:= True

Wait: 0.1

Permission:= False

Wait: 3.0

InitialData

Permission:= False

Targetposition:=[0,0,0,0,0,0,0]

TargetX:=0

TargetY:=0

TargetRotation:=0

TargetPercent:=0

Text:=0

RefObjectX:=0

RefObjectY:=0

MatchPercent:=80

InitialConnect

Connected:= False

Loop Connected<sup>?</sup> False

Connected:=socket\_open("172.17.128.107",3000,"socket0")

Wait: 0.1

DoPrepare

MoveJ

Prepare

Gripper Open (1)

GetCameraData

Wait Permission<sub>2</sub> True

Loop Targetposition[0]<sub>2</sub>0

Targetposition:=socket\_read\_ascii\_float(6,"socket0")

Wait: 0.1

TargetX:=Targetposition[2]

TargetY:=Targetposition[3]

TargetRotation:=d2r(Targetposition[4])

TargetText:=Targetposition[5]

TargetPercent:=Targetposition[6]

xOffset:=- (TargetX-RefObjectX)/1000

yOffset:=- (TargetY+RefObjectY)/1000

GoNewPosition

Offset:=p[xOffset,yOffset,0.0,0.0,0,TargetRotation]

PreMove:=pose\_trans(refpos1,Offset)

PickMove:=pose\_trans(pickPos1,Offset)

If TargetPercent >MatchPercent

Tracking Conveyor 1

MoveL

Wait: 0.0

MoveJ

```
    PreMove

    PreMove

    PickMove

    Gripper Close (1)

    PreMove

    MoveJ

    PlacePos

    Gripper Open (1)

    Prepare

Else

    Popup

GoNewPosition_Text

    Offset:=p[xOffset,yOffset,0.0,0,0,TargetRotation]

    PreMove:=pose_trans(refpos1,Offset)

    PickMove:=pose_trans(pickPos1,Offset)

If TargetPercent >MatchPercent

    If TargetText≠1

        Tracking Conveyor 1

        MoveL

        MoveJ

        PreMove
```

PreMove

PickMove

Gripper Close (1)

PreMove

MoveJ

PlacePos

Gripper Open (1)

Prepare

Else

Popup

Else

Popup

ClearAllData

Targetposition:=[0,0,0,0,0,0,0]

Wait: 0.01