

Kalle Jantunen

3D-KAMERAN JA YHTEISTYÖROBOTIN KÄYTTÖ
KONENÄKÖSOVELLUKSESSA

Automaatiotekniikan koulutusohjelma
2017

3D-KAMERAN JA YHTEISTYÖROBOTIN KÄYTTÖ KONENÄKÖSOVELLUKSESSA

Jantunen, Kalle
Satakunnan ammattikorkeakoulu
Automaatiotekniikan koulutusohjelma
Kesäkuu 2017
Ohjaaja: Leino, Mirka
Sivumäärä: 90
Liitteitä: 3

Asiasanat: konenäkö, robotiikka, 3D-kuvaus, yhteistyörobotti

Tässä opinnäytetyössä kehitettiin kevyiden kappaleiden tunnistus- ja poimintamenetelmä, kun kappaleet sijaitsevat laatikossa erilaisissa satunnaisissa asennoissa. Opinnäytetyön tavoitteena oli tutkia miten 3D-kuvausta voidaan käyttää yhteistyörobotin ohjaamisessa ja kappaleiden asentojen tunnistamisessa, kehittämällä sitä hyödyntävä konenäkösovellus.

Konenäkösovelluksen kehittäminen koostui 1) laitteistojen, ohjelmistojen ja kappaleiden suunnittelusta ja valinnasta, 2) järjestelmän asentamisesta ja toiminnan suunnittelusta sekä 3) toiminnan toteuttamisesta robotin ja kameran ohjelmoinnilla. Laitteistoksi valittiin rakenteelliseen valaisuun perustuva, 3D-laserprofiilia hyödyntävä 2380 Gocator-älykamera ja kaksisormisella Robotiq-tarttujalla varustettu, kevyen kanto kuorman UR5-yhteistyörobotti. Projektin yhteydessä laadittiin pikakäyttöohjeet ope- tuskäyttöä varten suunnitellulle 3D-älykamera-liukuhihnasovellukselle.

Gocatorin ohjelmointiin käytettiin Gocatorin graafista käyttöliittymää ja sen sisältämiä mittaustyökaluja, joiden avulla tunnistettiin ja mitattiin kappaleen poiminnan kannalta olennaisia piirteitä. Mittausarvot lähetettiin Ethernetin välityksellä UR5:lle.

UR5-yhteistyörobotin ohjelmointi toteutettiin sen graafisella PolyScope-käyttöliittymällä. Ohjelmointi koostui pääasiassa pisteiden opettamisesta, liikekäskyjen valinnoista, sekä Gocatorin väliseen tiedonsiirtoon liittyvistä asetuksista.

Valmis konenäkösovellus vastasi sille asetettuja tavoitteita. Järjestelmä esiteltiin Lounaisrannikon automaatiomessuilla. Teollisuusolosuhteita ja erilaista poimintakappa- letta varten tarvittaisiin uusien mittaustyökalujen valitsemista, järjestelmän yhteyksien sovittamista muihin mahdollisiin laitteisiin, sekä kameran ja robotin uudelleen kalib- rointia.

THE USE OF A 3D CAMERA AND A COLLABORATIVE ROBOT IN A MACHINE VISION SYSTEM

Jantunen, Kalle

Satakunnan ammattikorkeakoulu, Satakunta University of Applied Sciences

Degree Programme in automation technology

June 2017

Supervisor: Leino, Mirka

Number of pages: 90

Appendices: 3

Keywords: machine vision, robotics, 3D imaging, collaborative robot

In this thesis, a method for recognizing and picking of light objects was developed, when the objects are located in a box in different, random positions. The purpose of the thesis was to research how 3D imaging can be used to control a collaborative robot and positions of objects, by developing an appropriate machine vision system.

The development of the machine vision system consisted of 1) design and selection of the hardware, softwares and objects, 2) system's installation and planning of operation and 3) operation's execution by programming the robot and the camera. As a hardware, a Gocator smart camera based on structural lighting, utilizing a 3D-laser profile, was chosen as well as a UR5 collaborative robot of light bearing load equipped with a two-finger Robotiq gripper. During the project, a quick start manual for a conveyor belt application using the 3D smart camera, was made.

Programming of the Gocator was done by using Gocator's graphic user interface and software measurement tools, which made the recognitions and measurements of the object's features important for picking. The measurement data was transferred by using an Ethernet connection to the UR5.

For the UR5's programming, the PolyScope graphic interface was used. UR5's Programming mainly consisted of teaching the position points, choosing the motion commands as well as the settings of the data transfer for the Gocator.

Finished machine vision system met the expectations which were set for it. The system was presented at the Robocoast automation fair. To use the system in an industrial environment and to change a different picking object, choosing of new measurement tools, adjustments of connections for the other possible devices as well as calibrations of the camera and robot, have to be done for the system in order to make it function as required.

SISÄLLYS

1	JOHDANTO.....	7
2	KONENÄKÖ	8
2.1	Konenäön määritelmä	8
2.2	Konenäön historia	8
2.3	Teoria	9
2.3.1	Valon ominaisuuksista	9
2.3.2	Optiikka.....	11
2.3.3	Fotometria	12
2.3.4	Ihmisen näköjärjestelmä ja näkeminen	15
2.4	Konenäköjärjestelmän komponentit ja toiminta	17
2.5	Valaistus	19
2.5.1	Valonlähteet	20
2.5.2	Valaisutekniikoita.....	24
2.6	Kamerat ja kennot.....	28
2.6.1	CCD-kenno.....	31
2.6.2	CMOS-kenno	31
2.6.3	Älykamera	31
2.7	Kuvankäsittely ja graafinen esittäminen.....	32
2.7.1	Kuvankaappaus ja 3D-mallin muodostus.....	33
2.7.2	Kuvan analysointi ja algoritmit	34
2.8	3D-Kuvaus	34
2.8.1	Stereokuvaus	35
2.8.2	Rakenteellinen valaisu.....	36
2.8.3	Kinect	38
2.8.4	Time of Flight.....	39
2.8.5	Shape from X ja Shape from focus	40
2.9	Kalibrointi.....	41
3	ROBOTIIKKA	44
3.1	Robottiikan ja robotin määritelmät	44
3.2	Teollisuusrobottien historia	44
3.3	Standardit ja turvallisuus	45
3.4	Rakenteet ja osat	45
3.4.1	Ohjausjärjestelmä.....	46
3.4.2	Mekaaniset osat ja rakenteet	46
3.4.3	Asema-anturit	47
3.4.4	Kaapelit ja työkalut	47

3.5	Robottityypit	48
3.5.1	Suorakulmaiset robotit	49
3.5.2	Sylinterirobotit	50
3.5.3	Napakoordinaatistorobotit	50
3.5.4	Scara-robotit	51
3.5.5	Kiertyväniveliset robotit	52
3.5.6	Rinnakkaisrakenteiset robotit	53
3.5.7	Erikoisrobotit	53
3.5.8	Yhteistyörobotti	54
3.6	Toiminta ja kinematiikka	55
3.6.1	Koordinaatistot	55
3.6.2	Työkalun siirtäminen	56
3.7	Ohjelmointi	57
4	GOCATOR 2380 3D-PROFIILIKAMERA	58
4.1	Esittely ja käyttötarkoitus	58
4.1.1	Komponentit ja yleisnäkymä	58
4.1.2	Toiminta	61
4.2	Käyttöliittymä	63
4.3	Tiedonsiirto	64
4.4	Tekniset tiedot	66
5	UNIVERSAL ROBOT UR5 KÄSIVARSIROBOTTI	67
5.1	Yleisesitys	67
5.2	Käyttöliittymä ja ohjelmointi	68
5.3	Tiedonsiirto ja tekniset tiedot	71
6	KAPPALEEN KÄSITTELYJÄRJESTELMÄN SUUNNITTELU JA TOTEUTUS	73
6.1	Laitteiston suunnittelu ja valinta	73
6.1.1	Tehtävät ja edellytykset	73
6.1.2	Kappaleiden ja olosuhteiden esittely	74
6.1.3	Kamera ja valaistus	75
6.1.4	Ohjelmisto	75
6.1.5	Ohjausjärjestelmä	76
6.2	Järjestelmän toiminnan suunnittelu ja toteutus	77
7	YHTEENVETO	85
	LÄHTEET	86
	LIITTEET	

1 JOHDANTO

3D-kuvaus eli kohteiden kolmiulotteisten mallien luominen konenäön avulla, on tietojen ja robotitekniiikan kehittymisen myötä saanut yhä enenevässä määrin lisää jalansijaa erilaisissa konenäköjärjestelmissä. Yksi käytetyimmistä 3D-kuvaustekniikoista on laserprofiili-skannaus, jonka avulla pystytään mittaamaan mikrometrien tarkkuudella pinnankorkeuden vaihtelueroja, muodostamaan kappalemalleja tietokoneohjelmia ja 3D-tulostusta varten, sekä tunnistamaan esineitä erilaisista kolmiulotteisista asennuksista.

Kappaleen poiminnan suorittamiseksi robotin tulee selvittää kappaleesta sopivan tartuntakohdan sijainti ja asento. Yleensä tähän tehtävään on riittänyt 2D-toiminen konenäkösovellus. Tarve 3D-kuvauksen käytölle on syntynyt, kun kappale ei ole sijainnut tasaisella alustalla tai sen koko on vaihdellut mittasuhteiden pysyessä vakioina. Todellisella 3D-kuvauksella ei vain oleteta etäisyyksiä ja kokoja, vaan ne myös mitataan. Tämä lisää kappaleen poimintavarmuutta ja -nopeutta sekä vähentää poimintaolosuhteiden parantamiseen liittyviä manuaalisia säätöjä.

Tässä opinnäytetyössä kehitettiin 3D-konenäköön perustuva kappaleen poimintajärjestelmä opetuskäyttöä varten. Projekti alkoi laitteistojen ja poimittavan kappaleen suunnittelulla ja valinnalla. Tämän jälkeen perehdyttiin laitteiden ominaisuuksiin ja toimintoihin, joiden pohjalta tehtiin järjestelmän suunnittelu ja toteutus. Järjestelmä esiteltiin vuoden 2016 Lounaisrannikon automaatiomessuilla, Ulvilassa.

2 KONENÄKÖ

2.1 Konenäön määritelmä

Konenäkö on oppi metodeista ja tekniikoista, joiden avulla muodostetaan keinotekoiseen näköön perustuvia, käytännöllisiin sovelluksiin sopivia järjestelmiä (Davies 2005, 13). Konenäkö sisältää tietotekniikan, mekaniikan, optiikan ja teollisuusautomaation osaamista (Teledyne dalsa 2014). Sovellusalueita on mm. teollisuudessa, lääketieteessä, valvonnassa ja turvallisuudessa sekä viihde-elektronikassa (Leino 2015, 20).

2.2 Konenäön historia

Konenäön historia sisältää valaisemiseen, kameratekniikkaan ja tietokoneisiin liittyvien innovaatioiden ja keksintöjen historiaa. Eräitä nykyaikaisen konenäkötekniikan kehitykseen vaikuttaneita keksintöjä olivat 1920-luvulla keksitty LED-valonlähde, vuoden 1969 CCD-kameratekniikka, 1980-luvulla kehitetyt PC ja älykamera. Kuitenkin jo 1950-luvulla tehtiin ensimmäinen kaksikulotteisten kuvien kuviontunnistusmenetelmä. Vuonna 1960 luotiin 3D-konenäölle sen teoreettinen pohja, kun Larry Roberts esitti väitöskirjassaan, että kohteen kolmiulotteinen malli olisi mahdollista luoda vertaamalla kahta samasta kohteesta otettua kaksikulotteista kuvaa toisiinsa. (Vision-systems 2013; Epic Systems Inc. n.d.)

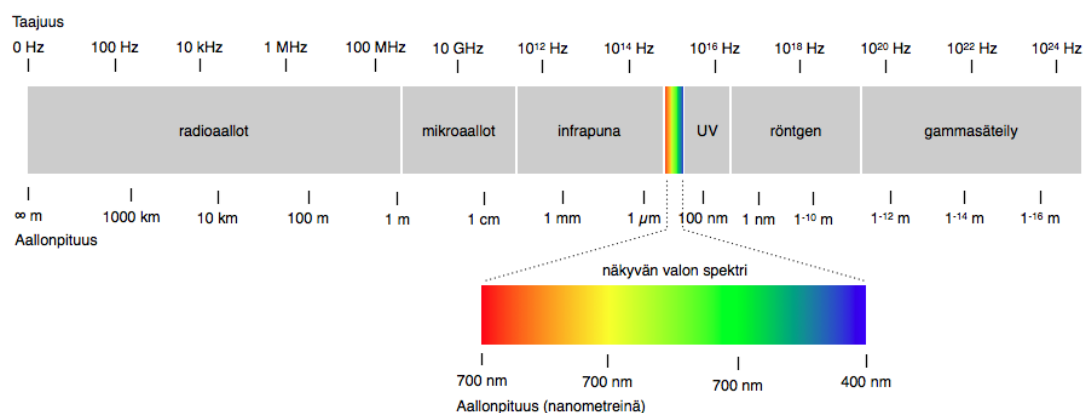
Ensimmäiset teolliset konenäkösovellukset tulivat käyttöön 1970-luvulla. Sovellukset keskittyivät tyypillisesti etsimään, useiden nykyaikaistenkin konenäköjärjestelmien tapaan, tuotteista löytyviä poikkeamia ja vikoja. (Birchfield 2016, 4.) Suomessa saha-teollisuus hyödynsi ensimmäisenä konenäköä tukkien halkaisijoiden mittaamiseksi ja niiden esilajittelemiseksi sahausta varten (Soini, n.d.). 1980-luvun aikana tehtiin ensimmäiseen tutkimus- ja kehitystyötä. Ala ei vielä ollut teollisuudessa yleisesti käytössä. Konenäön teollinen käyttäminen alkoi vasta 1990-luvulla menestyä niin paljon, että sitä alettiin pitää omana teollisuudenhaaranaan. 2000-luvun puolesta välistä nykypäivään, tietokoneiden prosessointitehojen kasvaminen ja kamerateknologian kehittyminen

nen ovat mahdollistaneet useisiin eri tehtäviin erikoistuneiden, kustannuksiltaan suhteellisen matalien konenäköjärjestelmien ja 3D-kuvausteknologian valmistamisen ja käyttämisen. (Birchfield 2016, 4-5; Epic Systems Inc. n.d.)

2.3 Teoria

2.3.1 Valon ominaisuuksista

Valolla tarkoitetaan yleensä ihmissilmälle näkyvää, sähkömagneettisen säteilyn n. 380-780 nm aallonpituusvälillä sijaitsevaa valon spektriä eli värikirjoa. Kuvassa 1 nähdään näkyvän valon spektrin osuus koko sähkömagneettisesta spektristä. Valo, kuten kaikki muukin sähkömagneettinen säteily, on luonteeltansa dualistista eli se voi samanaikaisesti koostua sekä aalto- että hiukkasliikkeestä. Tätä valon ominaisuutta kutsutaan aalto-hiukkasdualismiksi. (Hornberg 2006, 82-83.)



Kuva 1. Sähkömagneettisen säteilyn ja näkyvän valon spektrit (Aaltomuoto, n. d.).

Valon aaltoliike koostuu kahdesta komponentista: sähkö- ja magneettikentän poikittaisista, sinimuotoisista värähtelyistä, jotka normaalisti värähtelevät ilman vaihe-eroa kohtisuorasti toisiinsa nähden. Valolla on myös samat poikittaisten aaltoliikkeiden ominaisuudet kuin mekaanisilla aalloillakin, kuten taittuminen ja heijastuminen eri rajapinnoilta. Näiden lisäksi voi vielä tapahtua valon interferenssiä eli sähkömagneettis-

ten aaltojen yhteisvaikutusta, polarisoitumista eli säteilyn sähkökentän värähtelyä yhteen suuntaan ja eri sirontalajeja kuten diffraktiota eli valon taipumista. (Hornberg 2006, 83; Inkinen, Manninen & Tuohi 2011, 367-400.)

Valon hiukkasmallin mukaan aineen molekyylit ja atomit pystyvät emittoimaan ja absorboimaan säteilyä ainoastaan tietyn suuruisina energiapaketteina; hiukkaskvantteina eli fotoneina. Hiukkasmallin avulla voidaan selittää muun muassa mustan kappaleen säteily suurilla taajuuksilla, atomien lähettämät viivaspektrit sekä valosähköinen ilmiö. Muun muassa fotodiodit eli ledit hyödyntävät valosähköistä ilmiötä. (Inkinen ym. 2011, 453-478.)

Klassisen fysiikan mukaan sähkömagneettista säteilyä syntyy sähkövarauksen ollessa kiihtyvässä liikkeessä. Poikkeuksena tähän sääntöön ovat atomia kiertävät elektronit, jotka kvanttimekaniikan mukaan emittoivat säteilyä ainoastaan siirtyessään korkean energian radoilta matalammille. Elektronin siirtymistä matalammalta energiaradalta korkeammalle kutsutaan atomin virittymiseksi. (Inkinen ym. 2011, 333-334, 470-473, 512-519.)

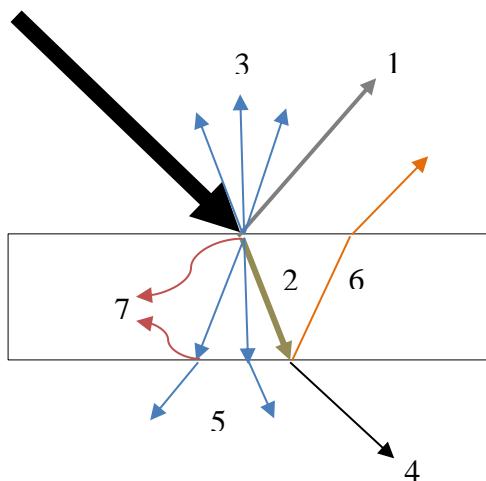
Luminesenssilla eli loistevaloilmioilla tarkoitetaan ilmiötä, jossa loisteaine emittoi valoa vielä energiatilansa virittymisen jälkeen. Virittyminen voidaan toteuttaa monella eri tavalla, kuten elektroni- ja ionisuihkulla, sähkökentällä tai valolla ja sitä korkeampitaajuisella sähkömagneettisella säteilyllä. Jälkimmäistä ilmiötä kutsutaan fotoluminesenssiksi ja sitä hyödynnetään esimerkiksi loistelampuissa. (Inkinen ym. 2011, 493-494.)

Sähkömagneettisen säteilyn intensiteetti tarkoittaa Poyntingin vektorin keskimääräistä arvoa eli aaltoenergian keskimääräistä siirtymistä kohtisuoraa pinta-alaa ja aikaa kohden (Inkinen ym. 2011, 340-342).

2.3.2 Optiikka

Konenäön kannalta on tärkeää tietää kappaleiden optisista ominaisuuksista mahdollisimman paljon. Tällöin konenäköjärjestelmään voidaan valita yksilökohtaisesti sopivimmat kameratyypit ja valaisutekniikat, tieteellisin menetelmin ja perustein. Tämän lisäksi vaaditaan aina kuitenkin myös järjestelyn testaaminen, jonka avulla varmistetaan ja hienosäädetään menetelmien toimivuutta myös käytännössä. Aalto-optiikka tutkii valon luonnetta ja käyttäytymistä aineessa aaltoliikkeenä, kun taas sädeoptiikka käsittelee valon kulkua lineaarisina säteinä kuvan muodostumisen ymmärtämiseksi. Kuviossa 1 näytetään valon käyttäytymistä kappaleessa ja sen rajapinnoilla:

1. Spekulaarinen heijastuminen eli peiliheijastus
2. Taivutus
3. Diffusiivinen heijastuminen
4. Yhdensuuntaissiirtymä
5. Diffusiivinen yhdensuuntaissiirtymä
6. Takaosan heijastus
7. Absorptio



Kuvio 1. Valon käyttäytyminen kappaleen kanssa (Steger, Ulrich & Wiedemann 2008, 9).

Optisia linsejä käytetään konenäössä ohjaamaan valonsäteiden kulkua halutulla tavalla, kuten niiden keskittämiseksi (kuperat linssit) tai hajauttamiseksi (koverat linssit). Konenäön käytetyimpiä linsejä ovat muun muassa perspektiivi-, telesentriset ja

astigmaattiset linssit. Optiset suodattimet hyödyntävät enimmäkseen valon additiivstä luonnetta, poistaen sähkömagneettisesta säteilystä tietyn tyyppisiä valoaltoja. Väri-suodattimet suodattavat tiettyjä aallonpituuksia, polarisaatiosuodattimet poistavat suotimen läpäisyakseliin nähden poikittaisesti suuntautuneita aaltoja. (Hornberg 2006, 138-149).

Halutusta tarkastelutavasta riippuen, linssien ja suodattimien käytöstä aiheutuu kuitenkin myös erilaisia kuvien vääristymiä. Nämä optiset vääristymät voidaan jakaa monokromaattisiin eli ei-Gaussisen optiikan mukaisiin ja kromaattisiin vääristymiin. Monokromaattisia vääristymiä ovat astigmaattiset eli hajataitteiset vääristymät, erilaiset pinnan kaareutumiset, kuten pallo-, tynnyri- ja neulatyynyvääristymät, sekä koma-ilmiö. Kromaattiset vääristymät jaetaan pitkittäis- ja sivuttaispoikkeamiin. Kromaattiset poikkeamat johtuvat eri aallonpituuksien erilaisista taitekertoimista, mikä näkyy kuvassa yleensä värien siroamisena. (Hornberg 2006, 316-321.)

2.3.3 Fotometria

Fotometrialla tarkoitetaan optiikan osa-aluetta, joka määrittelee ihmisen näköaistin mukaisesti painotettuja, valaistustekniikkaan liittyviä suureita (Inkinen ym. 2011, 424-427, 436). Vaikka ihmisenäön ja konenäön välillä on eroavuuksia, voidaan fotometrian termejä ja suureita kuitenkin käyttää suurimmalta osin myös konenäössä. Taulukossa 1 määritellään yleisimmät fotometrian suureet ja yksiköt:

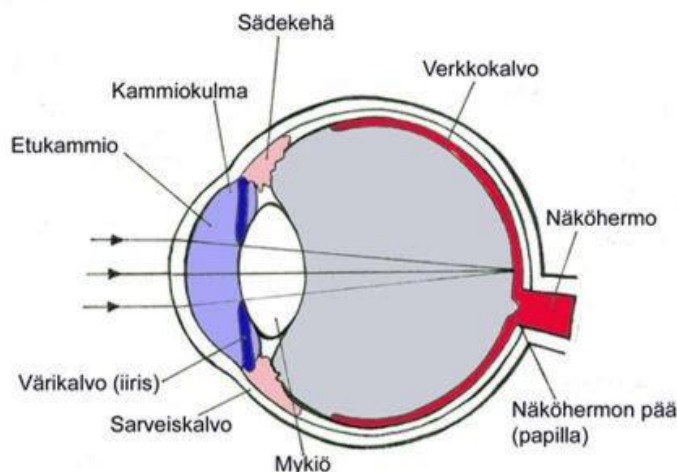
Taulukko 1. Yleisimmät fotometrian suuret (Inkinen ym. 2011, 436-448).

Suure	Yksikkö	Määritelmä	Selitys
Avaruuskulma ω	steradiaani (sr)	$\omega = \frac{A}{r^2}$	Ilmaisee 3-ulotteisen suunnan.
Valovoima I	kandela (cd)	$I = \frac{\Phi}{\omega}$	Ilmaisee valon voimakkuuden/määrän tiettyä avaruuskulmaa kohden.
Valovirta Φ	lumen (lm)	$\Phi = I\omega$	Ilmaisee valon määrän.
Valotehokkuus η	lm/W	$\eta = \frac{\Phi}{P}$	Ilmaisee valovirran suhdetta sen ottamaan sähkötehoon.
Luminanssi L	cd/m ²	$L = \frac{I}{A_p}$ $L = \frac{\rho E}{\pi} * \frac{1}{sr}$ jossa $\rho = \text{heijastusuhde}$	Ilmaisee pinnan valovoiman tiheyttä tarkastelusuuntaan.
Valaistusvoimakkuus E	luksi (lx)	$E = \frac{\Phi}{A}$ $E = \frac{I \cos\theta}{r^2} * sr$	Ilmaisee kuinka hyvin pinta on valaistu.
Kontrasti K	-	$K = \frac{ L_k + L_t }{L_t}$	Ilmaisee luminanssieron.

Jotta katseltava kohde pystyttäisiin nähtävästi havaitsemaan, tarvitaan kohteen ja sen taustan välillä riittävän suuri kontrasti eli luminanssiero. Luminanssilla tarkoitetaan tässä silmän spektriherkkyydellä painotettua, kohteen pinnalta heijastuvan valon voimakkuutta tarkastelusuuntaan kohtisuoraan vasten projisoitua pinta-alaa kohden. Koska silmä aistii eri värejä eri herkkyyksillä, kontrastin käsitteeseen sisältyy näin ollen myös värikontrasti. (Inkinen ym. 2011, 426, 439-446.) Konenäön tapauksessa kontrastilla tarkoitetaan vastaavasti pikselien harmaasävyeroja, jotka syntyvät pikseliin kohdistuvista valon intensiteettieroista ja kirkkaudella usein säteilyn intensiteettiä tai valovirtaa (Hornberg 2006, 86-87, 134-135).

2.3.4 Ihmisen näköjärjestelmä ja näkeminen

Konenäössä käytettävien menetelmien ymmärtämiseksi on hyödyllistä tietää ihmisen näköjärjestelmän toimintaa ja sen kohtaamia haasteita. Näkeminen koostuu kolmesta perusvaiheesta: kuvan muodostuksesta, hermosignaalien siirrosta ja tulkinnasta. Ylösalaisen optisen kuvan muodostus tapahtuu kuvassa 2, kun sarveiskalvon läpäisevät valon säteet kulkevat lukuisten valoa taittavien, heijastavien ja absorptiivien elinten läpi verkkokalvolle. Verkkokalvon sisällä sijaitsevat valoherkät sauva- ja tappisolut muuntavat ja “esikäsittelevät” fotonit lukuisiksi näköhermon kautta kulkeviksi, rinnakkaismuotoisiksi sähkökemiallisiksi hermosignaaleiksi. Signaalit kulkeutuvat talamuksen kautta silmän vastakkaisen aivokuoren takaraivolohkopuoliskon visuaalisia tietoja käsitteleville näkökeskuksille ja niihin yhteyksissä oleville aivojen osille, joissa näköaistimuksen eli kuvan tulkinnan arvellaan syntyvän. (Hornberg 2006, 1-31.)

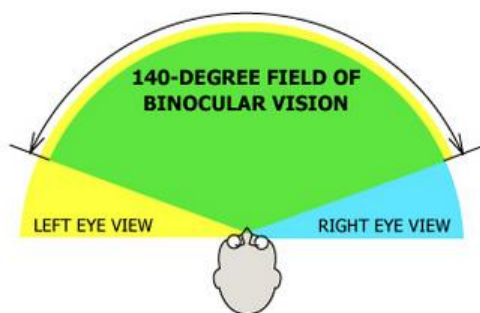


Kuva 2. Silmän rakenne (Mahola, 2013).

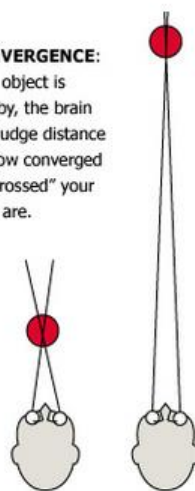
Ihmisen syvyysnäkö hyödyntää etäisyyksien arvioimiseksi useita erilaisia vihjeitä. Nämä vihjeet jaetaan monokulaarisiin eli yksisilmäisiin ja binokulaarisiin eli kaksisilmäisiin syvyysvihjeisiin. Todellinen syvyysnäkö eli stereonäkö perustuu aina kahden silmän muodostamien, eri kohdista ja kulmista yhtä aikaa aistittujen kuvien vertailuun ja niiden vastinpisteiden löytämiseen ja yhdistämiseen. (Davies 2005, 443-454.) Kuvassa 3 näytetään esimerkit yhdestä näköjärjestelmän käyttämästä monokulaarisesta syvyysvihjeestä eli liikeparallaksista ja binokulaarisesta syvyysvihjeestä eli silmien lähentymisestä.

Seeing in 3D

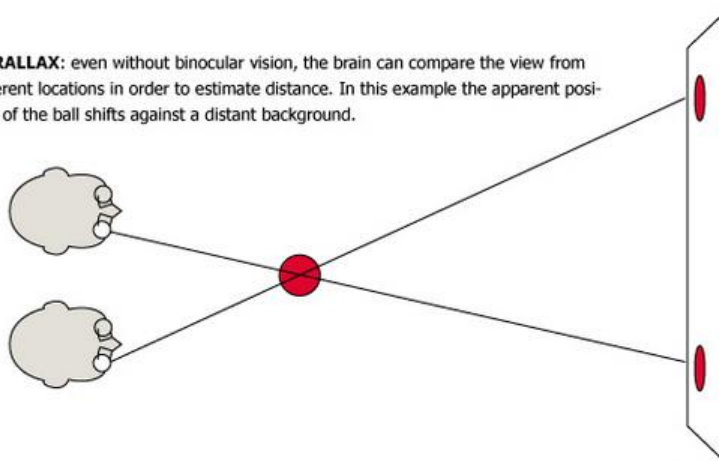
The brain uses several tricks to make us see in three dimensions. Despite our almost 180-degree field of view, binocular vision is only possible in the area where the view from both eyes overlap.



CONVERGENCE: if an object is nearby, the brain can judge distance by how converged or "crossed" your eyes are.



PARALLAX: even without binocular vision, the brain can compare the view from different locations in order to estimate distance. In this example the apparent position of the ball shifts against a distant background.



SOURCE: LIFE'S LITTLE MYSTERIES RESEARCH

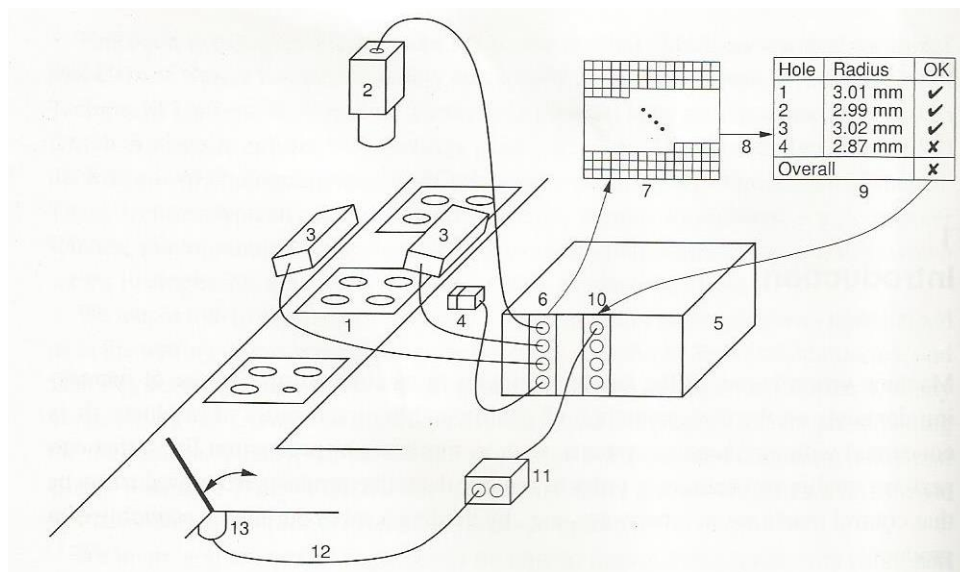
Graphic by Karl Tate

Life's Little Mysteries

Kuva 3. Esimerkit erilaisista syvyysvihjeistä (Live Science, 2010).

2.4 Konenäköjärjestelmän komponentit ja toiminta

Teollinen konenäköjärjestelmä koostuu useasta eri komponentista joiden tarkoituksena on joko kuvan ottaminen, kuvaolosuhteiden parantaminen, tiedonkäsittelyminen/siirtäminen tai kappaleeseen vaikuttaminen. Kuvassa 4 nähdään esimerkki tyypillisestä konenäköjärjestelmästä ja sen komponenteista:



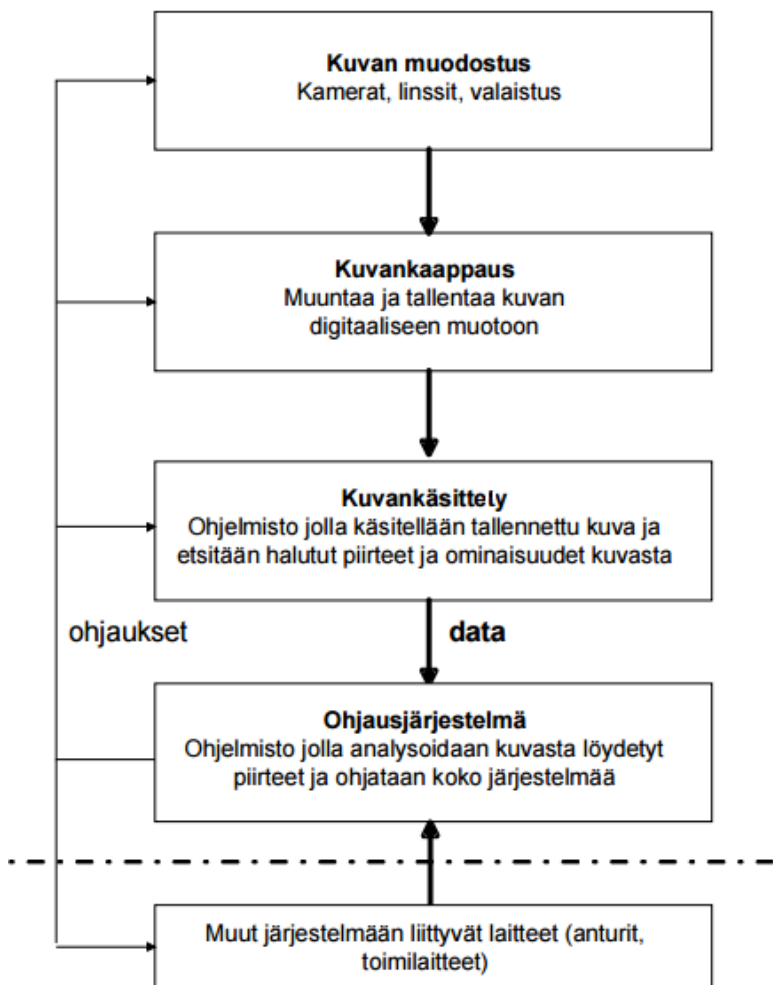
Kuva 4. Tyypillisen konenäköjärjestelmän komponentit (Steger ym. 2008, 2).

1. Tarkasteltava kohde tai kappale
2. Kamera
3. Valaisimet
4. Sensori eli kuvanoton "triggeri"
5. Tietokone
6. Kameran ja tietokoneen rajapinta, usein kuvankaappauskortti
7. Käsiteltävän kohteen tai kappaleen kuva
8. Konenäköohjelma
9. Kohteesta muodostettu arvio
10. Digitaalisten tulojen- ja lähtöjen rajapinta (DI/DO-rajapinta)
11. Ohjelmoitava logiikka (PLC)
12. Kenttäväylä
13. Toimilaite

Esimerkkijärjestelmän tarkoituksena on erottaa vialliset kappaleet virheettömistä kappaleista. Kappaletta (1.) kuljetetaan jollakin mekaanisella laitteella tässä tapauksessa liukuhihnakuljettimen avulla kameran (2.) alapuolelle. Tämän jälkeen valaisimet (3.) valaisevat kappaletta tuoden esiin kappaleesta sen haluttuja tarkasteltavia piirteitä. Valokennoanturi (4.) tunnistaa kappaleen lähestymisen kameran alapuolelle ja liipaisee (“triggers”) kuvanoton käynnistymisen. Kuvanoton jälkeen kamera välittää datan kuvankaappaimelle (6.), jonka laiteohjain muodostaa ja tallentaa kuvan (7.) tietokoneen (5.) muistiin. Kuvankaappainta voidaan käyttää myös ohjaamaan valaisimia. Konenäköohjelma (8.) tarkastelee kuvaa ja palauttaa arvion (9. hylätty/hyväksytty) kappaleesta ohjelmoitavalle logiikalle (11.), DI/DO-rajapinnan (10.) välityksellä. Ohjelmoitava logiikka ohjaa toimilaitetta (13.), tässä tapauksessa liukuhinnan jakajan sähkömoottoria, kenttäväylän (12.) välityksellä. Lopuksi liukuhinnan jakaja erottaa viallisen tuotteen virheettömistä. (Steger ym. 2008, 2-3.)

Monet näistä komponenteista ja toiminnoista voidaan korvata myös erityyppisillä ratkaisuilla. Esimerkiksi kuvankaappauskortin tilalla voidaan käyttää standardirajapintoja kuten Ethernet:iä tai USB 2.0:a, jolloin tietokone vastaa myös kuvankaappauksesta. Tietokoneena voidaan käyttää teollisuus-PC:tä tai älykameraa. (Steger ym. 2008, 1-3.)

Kuvassa 5 näytetään vielä konenäköjärjestelmän toiminnan pääkohdat. Tässä kuvan muodostuksella tarkoitetaan *optisen* kuvan muodostusta ja kuvankaappauksella *digitaalisen* kuvan muodostusta. Kuvankäsittelyllä voidaan myös viitata koko digitaalisen kuvankaappauksen ja analyysin teon prosessiin (Hornberg 2006, 492).



Kuva 5. Konenäköjärjestelmän toiminnan pääkohdat (TKK Automation Technology Laboratory n.d., 3).

2.5 Valaistus

Ehkä kaikista merkittävin osa-alue onnistuneessa konenäköjärjestelmässä on hyvä valaistus. Ilman selviä kontrastieroja kiinnostavan kohteen ja taustan välillä, sopivaa kirkkautta ja pinnanmuotojen erottumista, ei parhainkaan konenäköohjelma pysty arvioimaan kohdetta oikealla tavalla. Kaikki konenäköjärjestelmän kuvan muodostuksessa käsiteltävästä informaatiosta tulee valon kautta. Erilaisia valonlähteitä ja niiden suuntauksia käyttämällä saadaan kappaleesta tehokkaasti esiin haluttuja piirteitä ja vähennetään ei-toivottuja. (Hornberg 2006, 73-78.)

2.5.1 Valonlähteet

Konenäössä käytettävät keinotekoiset valonlähteet voidaan atomien viritystavan mukaan jakaa kahteen eri tyyppiin: lämpötila- ja luminesenssisäteilijöihin. Lämpötilasäteilijöitä ovat erilaiset hehkulamput, jotka tuottavat hehkulankaa kuumentamalla yhdistelmän valon eri aallonpituuksia. Luminesenssisäteilijät taas emittoivat valoa valikoidusti ja kapealla spektrivälillä. Luminesenssisäteilylähteitä ovat loistelamput, Ledit ja laserit. (Hornberg 2008, 88-89.)

2.5.1.1 Hehku-/halogeenilamppu

Hehkulamppu koostuu lasikuvusta, hehkulangasta, johtimista ja metallisesta kannasta. Lasikuvun sisällä voi olla tyhjiö tai passiivista kaasuseosta kuten typpeä ja argonia. Halogeenilamputta kaasuseokseen on lisätty jotakin halogeeniryhmään kuuluvaa alkuaineakaasua kuten jodia tai bromia, jotka reagoidessaan volframihehkulangan kanssa lisäävät lampun luminanssia ja kokonaiskäyttöikää.

Edut:

- suuri valovoima
- jatkuva spektri
- toimii matalalla jännitteellä
- korkea ympäristön lämpötilan sieto (n. 300 °C saakka).

Haitat:

- hyvin lyhyt käyttöikä (n. 300-2000 tuntia)
- huono valotehokkuus → tuottaa paljon lämpöä
- herkkä värähtelyille ja jännitteen muutoksille
- hidas toiminta → sopii ainoastaan staattiseen valaisemiseen

Käyttö konenäkösovelluksissa:

- ei yksittäistä käyttökohdetta

(Hornberg 2006, 89-91; Steger ym. 2008, 7.)

2.5.1.2 Ksenonlamppu

Ksenonlamput ovat kaasupurkauslamppuja, jossa valoa tuottaa ksenon-kaasun hehkuminen. Ksenonlamppu koostuu ksenonkaasua sisältävästä lasiputkesta, elektrodeista ja ohjauselektronikasta.

Edut:

- erittäin suuri valovoima
- korkea värilämpötila
- nopea välähdystaajuus n. 200 välähdystä sekunnissa
- käyttöikä, n. 10^8 eli 100 000 000 välähdyskertaa

Haitat:

- kallis
- korkea käyttöjännite, vaatii lisäsuojauksia
- vaihteleva intensiteetti välähdysten välillä (<10%)

Käyttö konenäkösovelluksissa:

- hyödyllinen värikuvien käsittelyssä suuren valomäärän vuoksi
- hyödyllinen nopeissa prosesseissa
- käytetään toimintaehtojen vuoksi teollisuudessa varauksin

(Hornberg 2006, 92-93; Steger ym. 2008, 7.)

2.5.1.3 Loisteputki/-lamppu

Loisteputki on kaasupurkauslamppu, joka hyödyntää valontuotossa ioni- ja fotoluminesenssia. Loisteputki sisältää elohopeahöyryä seoksena jossakin jalokaasussa kuten argonissa tai neonissa, jota sähköpurkauksella virittämällä saadaan tuottamaan UV-säteilyä. Kun UV-säteily osuu putken sisäpinnalle pinnoitettuun fosforisuolaan, loisteputki tuottaa putken ulkopuolelle nähtävää valoa.

Edut:

- halpa
- erilaisia värilämpötiloja (3000-6000 K)
- valaisee suuriakin alueita

Haitat:

- joustamaton muoto (putki, rengas)
- kohtalaisen lyhyt käyttöikä (2500-12000 h)
- soveltuu vain staattiseen valaisuun
- vaatii HF-liitäntälaitteen välähdystaajuuden kasvattamiseksi (≥ 22 kHz)

Käyttö konenäkösovelluksissa:

- suurten alueiden valaisuun
- matalakustannuksisiin sovelluksiin

(Hornberg 2006, 93-95; Steger ym. 2008, 8.)

2.5.1.4 LED

LED (Light Emitting Diode) eli hohtodiodi on pn-rajapintainen puolijohdekomponentti, joka myötäsuntaan kytkettynä tuottaa elektronien ja aukkojen yhdistyessä fotoneita (Inkinen ym. 2011, 577-578). Ledit ovat konenäön käytetyimpiä valonlähteitä niiden monipuolisten ominaisuuksien ja luotettavan toiminnan vuoksi. Kuvassa 6 esitetään esimerkkejä erilaisista LED-valaisimista.



Kuva 6. Kokoelma konenäössä käytettävistä LED-valaisimista (Phoenix Imagine n.d.).

Edut:

- pitkä käyttöikä (käyttöolosuhteista riippuen jopa yli 100 000 h)
- kestävä
- pystyy emittoimaan kaikkia värejä ja infrapunavaloa
- monipuoliset muodot ja koot
- matala virrankulutus ja lämmöntuotto

Haitat:

- suurin käyttölämpötila n. 60 °C

Käyttö konenäkösovelluksissa:

- soveltuu erittäin hyvin teolliseen käyttöön
- ideaalinen valonlähde useisiin konenäkösovelluksiin

(Hornberg 2006, 95-98.)

2.5.1.5 Laser

Laser (Light amplification by stimulated emission of radiation) tarkoittaa laitetta, joka tuottaa stimuloitua emissiota avulla koherenttia eli samanvaiheista, yhteen suuntaan etenevää valoa. Stimuloitussa emissiossa fotonin indusoi samaan energiatilaan virittyneistä atomista toisen samaan suuntaan etenevän fotonin. Laservalon synnyn edellytyksenä on stimuloitu emissio ja väliaineen käänteinen miehitys eli viritetyssä olevien atomien määrän tulee olla suurempi kuin perustilassa olevien määrän. Tällä tavalla tuotettu valo sisältää ainoastaan yhtä aallonpituutta eli se on niin sanottua monokromaattista säteilyä. (Inkinen ym. 2011, 488-492.)

Yleisimmät lasertyypit ovat joko kaasui- tai puolijohdelasereita. Koneäkösovelluksissa käytettävistä lasereista ovat kaikki lähes yksinomaan puolijohdelasereita. Puolijohdelaserissa laservalo saadaan syntyään sähkövirran ylittäessä tiettyyn aallonpituuteen, yleensä näkyvän valon ja infrapuna-alueen väliin (380nm...1 mm) keskittyneen pn-liitoksen eli laseriodin kynnyksen. Johtuen laservalon fyysisistä ominaisuuksista ja niiden soveltuvuuksista etäisyyksien mittaamiseen, käytetään lasereita koneäkösovelluksissa usein vain

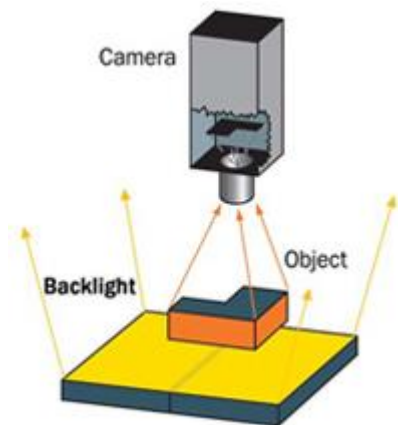
3D-kuvaustekniikoiden valaisulähteinä. (Hornberg 2006, 99-100; Inkinen ym. 2011, 578-579.)

Laserit jaetaan eri turvallisuusluokkiin niiden käyttäjiin mahdollisesti kohdistuvien silmä- ja ihovammojen riskien mukaisesti. Suomessa STUK (Säteilyturvakeskus) määrittelee laserien turvallisuusluokat 1 – 4, käytön vaarallisuuden kasvaessa luokan mukaisesti; luokan 1-lasereista aiheutuu mitätöntä vaaraa, luokan 4-laserit voivat kudoksiin kohdistuessaan aiheuttaa vakaviakin vammoja ja niiden voimakkuuksille ei ole määritelty ylärajaa. Luokan perään voidaan lisätä myös kirjainmerkinnät M (Magnification), C (Contact), R tai B, turvallisen käytön lisämäärittelemiseksi. (STUK 2015.)

2.5.2 Valaisutekniikoita

Valaisutekniikoiden luokittelemiseksi voidaan käyttää kolmea eri kriteeriä: valon suunnallisia ominaisuuksia, valaistuksen järjestelyä ja kuvatun valaisukentän (Field) tyyppiä. Suunnallisilla ominaisuuksilla tarkoitetaan diffusiivista, ohjattua, telesentristä tai rakenteellista valaisua. (Hornberg 2006, 154-157.)

Valaistuksen järjestelytavat jaetaan valonlähteen sijainnin suhteen kahteen eri ryhmään: tausta- ja etuvalaisuihin. Taustavalaisussa käytetään tyypillisesti valopaneeleja, jotka sijoitetaan eri puolelle kohdetta kuin missä kameran on. Kontrastit syntyvät näin ollen läpäisevien ja huonosti läpäisevien valonsäteiden välisistä luminanssieroista eli kameran havaitsemista harmaasävyeroista. Taustavalaisun valo voi olla diffusiivista, ohjattua tai kollimoitua eli tyypillisesti kameraa kohden telesentrisellä optiikalla ohjattua samansuuntaisesti etenevää valoa. Diffusiivista taustavalaisua käytetään joko litteiden, läpinäkymättömien kappaleiden siluettien muodostamiseen tai läpinäkyvien kappaleiden takaosien rakenteiden esiintuomiseen. Telesentrinen taustavalaisu soveltuu yhdessä telesentrisen objektiivin kanssa useisiin mittaus-, laadunvalvonta- ja asennontunnistustehtäviin. Valonlähteinä käytetään yleensä Ledejä tai loisteputkia. (Hornberg 2006, 157-158, 177-184; Steger ym. 2008, 15-17.) Kuvassa 7 näytetään taustavalaisun toimintaperiaate.



Kuva 7. Taustavalaisuun perustuva konenäköjärjestely (Microscan 2017).

Etuväläisutekniikoissa hyödynnetään valon heijastumista kappaleen pinnoilta kameralle. Tällöin valonlähde on siis samalla puolella kohdetta kuin kamerakin. Kontrasti syntyy heijastuneiden valonsäteiden harmaasävyeroista. Kun valonlähde on asetettu kohteeseen nähden sellaiseen kulmaan, jossa suurin osa emittoidusta valosta heijastuu kameralle, tarkoitetaan Bright Field-valaisua. Jos taas suurin osa valosta kohdistuu kamerasta pois päin, puhutaan Dark Field-valaisusta. Näiden väliin sijoittuvista epämääräisistä tapauksista voidaan käyttää nimityksenä Partial Bright Field-valaisua. Konenäköjärjestelmien käytetyimpiä etuväläisutekniikoita ovat:

- Diffuusivalaisu
- Aksiaalinen diffuusivalaisu
- Ohjattu valaisu
- Kohtisuora valaisu
- Sivuväläisus

(Hornberg 2006, 157-176.)

Valaistuksen ohjauksella säädetään valon kirkkautta ja toimintatiloja. Erilaisia toimintatiloja ovat staattinen, pulssi- ja strobo- eli salamaväläistus, joiden avulla välaistus optimoidaan kuvaustekniikkaan ja teollisuussovellukseen sopivaksi. (Hornberg 2006, 186-197.)

2.5.2.1 Diffuusivalaisu

Diffuusivalaisumenetelmässä (Partial tai Bright Field-valaisu) valoa tuotetaan kohteelle jossain määrin tasaisesti eri kulmista. Diffuusivalaisun tarkoituksena on estää varjojen syntymistä, sekä vähentää tai estää spekulaaarisia heijastumia. Tämä voidaan toteuttaa joko aksiaalisella diffuusivalaisimella tai puolipallon muotoisella kupoli- tai rengasvalaisimella. Kupolivalaisimessa diffusiivinen valo saadaan aikaiseksi heijastamalla valoa kupolin koveralta sisäpuolelta.

Käyttö:

- kolmiulotteisten kappaleiden tasaiseen valaisemiseen
- varjojen vähentämiseen
- tekstuurien pehmentämiseen tasaisella pinnalla

(Hornberg 2006, 169-173.)

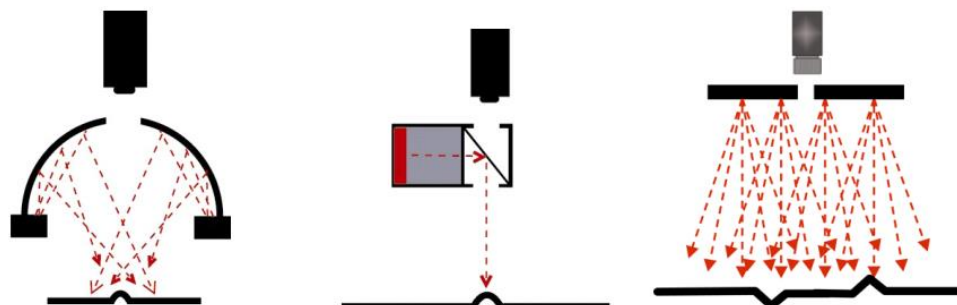
Aksiaalisessa diffuusivalaisimessa diffusiivinen valonlähde heijastaa valoa kohteelle saman puoliläpäisevän peilin kautta, josta kamera ottaa kohteesta kuvankin.

Käyttö:

- hieman epätasaisten pienten osien tasaiseen valaisemiseen
- spekulaaarisesti heijastavat kappaleet
- himmeät ja läpinäkyvät heijastavat pinnat

(Hornberg 2006, 160-162; Steger ym. 2008, 12-14).

Kuvassa 8 näytetään diffuusivalaisimien toimintamenetelmät.



Kuva 8. Kupoli-, aksiaalinen ja rengasdiffuusivalaisumenetelmät (Daryl 2012, 15).

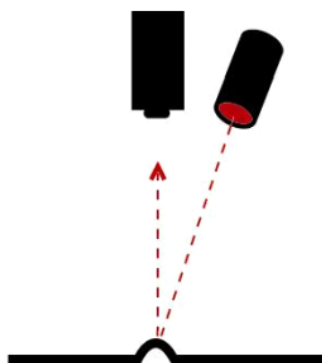
2.5.2.2 Ohjattu- ja kohtisuoravalaisu

Kuvan 9:n mukaisessa ohjatussa valaisussa (Partial Bright Field) valo suunnataan kappaleeseen nähden suuresta kulmasta, jolloin saadaan tehokkaasti esiin sen topografisia yksityiskohtia ja kontrasteja.

Käyttö:

- diffusiivisesti heijastavien kappaleiden tasaiseen valaisuun
- syvennyksien esiin tuomiseen
- heijastusten tietoiseen hyödyntämiseen

(Hornberg 2006, 162-163.)



Kuva 9. Ohjatunvalaisun toimintaperiaate (Daryl 2012, 16).

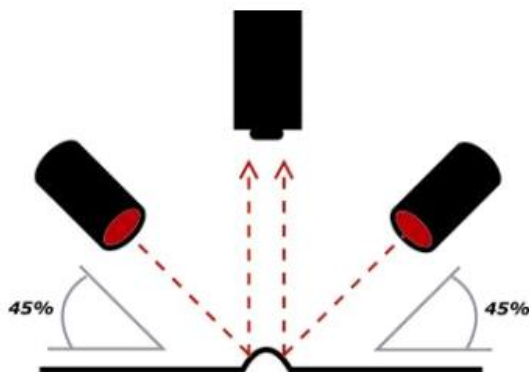
2.5.2.3 Sivuväläisy

Sivuväläisyssä (Dark Field) asetetaan valonlähde kappaleeseen nähden niin pieneen kulmaan, että valoa heijastuu kameralle vain pinnan kohoumista ja syvennyksistä. Tämän takia tasaiset pinnat näkyvät kuvissa tummina ja esimerkiksi reunat, halkeamat ja tekstuurit vaaleina. Kuvassa 10 esitetään sivuväläisyä toimintaperiaate.

Käyttö:

- ääri viivojen, muotojen, rakenteiden, tekstuurien ja reunojen esiintuomiseen
- tekstuurisiin vikoihin, kuten naarmuihin ja halkeamiin
- yksittäisiin yksityiskohtiin

(Hornberg 2006, 173-176.)

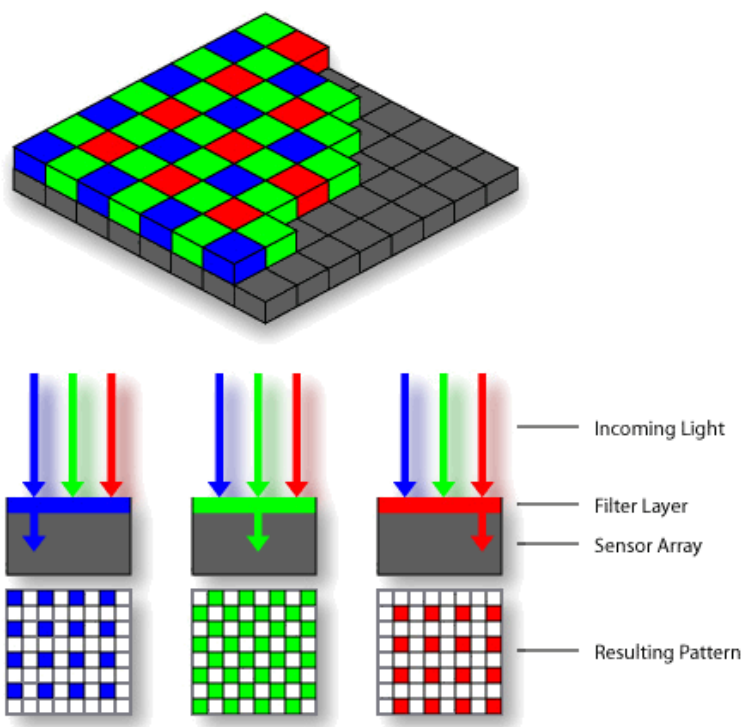


Kuva 10. Sivuväläisun toimintaperiaate (Daryl 2012, 16).

2.6 Kameran ja kennot

Kamera on optinen laite, joka muodostaa tarkasteltavan kuvan kohdistamalla valoa linssien eli objektiivin avulla filmille tai sensorille. Sensori eli kuvakenno voi valoherkkien tunnistimiensa, kuten fotodiodien järjestelytavan mukaan olla joko viiva- tai matriisikenno. Viiva- eli lineaarikennossa fotodiodit on aseteltu yhteen riviin, jolloin kaksiuulotteisen kuvan muodostamiseksi tarvitaan kohteen ja kameran välistä suunnattua suhteellista liikettä. Matriisi- eli aluekennolla kuva voidaan kuitenkin muodostaa yhdellä valotuskerralla, koska tunnistimet on sijoitettu kennon suorakulmion muotoiselle alalle. Kuvakennojen toiminta perustuu fotodiodissa valosähköisen ilmiön synnyttämän varauksen ja valon intensiteetin suoraan verrannollisuuteen, joka muunnetaan numeeriseksi arvoksi eli harmaasävyarvoksi. Nykyaikaisissa digitaalikameroissa käytetään yleisesti kahta eri kennotekniikkaa: CCD- ja CMOS-kennoja. (Hornberg 2006, 361-426.) Joissakin laser-skannereissa käytetään analogisia PSD-sensoreita (Position Sensitive Detector) (Kennedy n.d., 2-5).

Toisin kuin ihmisen silmä, kennot havainnoivat vain valon intensiteettiä eli ne eivät erota spektrin eri värejä toisistaan. Värikuvan muodostamista varten käytetäänkin erilaisia värisuodatin- ja kameraratkaisuja, kuten kuvan 11 Bayer-suodatinta, moottorilla vaihtuvaa suodatinta tai valon jakoa kolmelle kennolle. Vertaamalla pikseliin tiedetyn värisuodattimen läpi kohdistuvaa intensiteettiä viereisiin pikseleihin kohdistuviin intensiteetteihin, pystytään päättämään kunkin pikselin omat väriarvot. (Steger ym. 2008, 42-45.) Kennojen lähettämät väriarvot ilmaistaan yleensä RGB- tai YUV-arvoina. (Hornberg 2006, 480). Taulukossa 2 määritellään kameran tärkeimmistä ominaisuuksista kertovia termejä.



Kuva 11. mosaiikkikuvioinen Bayer-suodatin eli CFA (Fotorico n.d.).

Taulukko 2. Kamerateknologian termien määritelmiä (Hornberg 2006, 42-45; Korpela n.d.; Steger ym. 2008, 18-30).

Termi	Määritelmä/kaava	Selitys
Pikseli, alipikseli	-	Digitaalisen kuvan tai fyysisen kennon pienin kuvaelementti. Kuvan pikseli voi myös koostua alipikseleistä.
Vokseli	-	3D-mallin pienin kuvaelementti.
FOV eli Field of View	FOV = maksimi kappaleen koko + paikoituksen toleranssi + marginaali + sopeutus kennon kuvasuhteeseen	Kuvakoko. Tarkoitetaan joissakin tapauksissa myös kuvakulmaa (Angle Of View).
DOF eli Depth of Field	Useita, riippuu käytetystä objektiivista.	Kameran terävyysalue eli etäisyysväli jossa kuva näkyy vielä hyväksyttävän terävänä.
Kameran resoluutio	$R_k = P_{\text{pikselit pystyrivi}} * P_{\text{pikselit vaakarivi}}$	Kameran pikselien yhteenlaskettu määrä.
Spatiaalinen resoluutio	$R_s = \frac{FOV}{R_k}$ $[R_s] = mm/pix$	Pikselin leveyttä tai pituutta vastaava todellinen leveys tai pituus.
Mittaustarkkuus	-	Pienin mitattavissa oleva piirre esim. pikseleinä.
Peittoaste	$F = \frac{A_{\text{valoherkkä}}}{A_{\text{kok.}}} * 100\%$	Pikselin valoherkän alueen alan suhde sen viemään kokonaisalaan. Voidaan kasvattaa esimerkiksi käyttämällä mikrolinssejä.
f -luku	$F = \frac{f}{D}$	Aukkosuhde, ilmoittaa polttovälin f suhdetta valontuloa eniten rajoittavan aukon halkaisijaan D .

2.6.1 CCD-kenno

CCD-kennossa (Charge-Coupled Device-sensor) fotodiodissa kuvan valotusajan aikana muodostuneet pikselivaraukset siirretään puolijohdekondensaattoreja hyödyntävien siirtoporttien (Transfer gate) kautta sarjaliikennelukijarekistereihin (Serial Readout Registers). Näitä kahta komponenttia kutsutaan CCD:ksi. Sarjaliikennelukijarekisterien jälkeen varaukset muunnetaan jännitteiksi, vahvistetaan ja konvertoidaan joko analogi- tai digitaalisignaaleiksi, joista kohteen digitaalinen kuva voidaan myöhemmin muodostaa. CCD-kennot pystyvät tuottamaan hyvin pienen kohinan omaavia kuvia. Teknisinä heikkouksina CCD-kennoissa ovat mahdolliset varausten “vuodot” kondensaattorilta toiselle sekä häiriöt varausten siirrossa (“smearing” ja “blooming”). (Steger ym. 2008, 36-40; Hornberg 2006, 364-373, 378.)

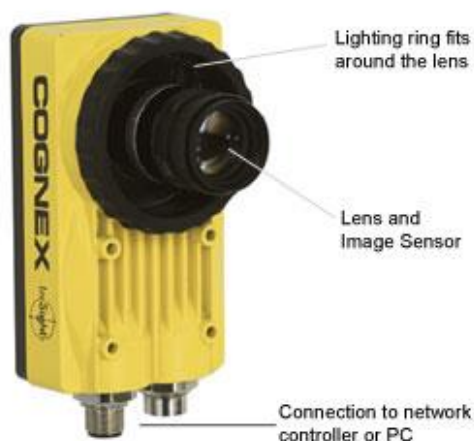
2.6.2 CMOS-kenno

CMOS-kenno (Complementary Metal-Oxide-Semiconductor-sensor) eroaa toiminnaltaan CCD-kennosta lähinnä pikselivarauksen erilaisella käsittelytavalla. Toisin kuin CCD-kennossa, CMOS-kennon pikselivaraukset muunnetaan jännitearvoiksi jo fotodiodien kohdalla. Kun pikseli sisältää vahvistimen sitä kutsutaan aktiiviseksi pikseliksi (APS). Yksittäisen aktiivisen pikselin jännitearvot voidaan lukea muista pikseleistä erillään sen sisältämien rivi-sarake-osoitteen ja noin 3-4 transistori-kytkimien avulla, mikä lisää kuvankäsittelyn suoritusnopeutta. CMOS-kennolla on useita CCD-kennohin nähden parempia nopeuden, virrankäytön ja rajapintojen yhteensopivuuksien ominaisuuksia. CMOS-kennot soveltuvatkin loogisten kytkintoimintojensa vuoksi hyvin älykameroiden kennotyypeiksi. (Hornberg 2006, 373-379.)

2.6.3 Älykamera

Konenäössä älykameralla tai älysensorilla tarkoitetaan kameraa, johon on yhdistetty useita eri konenäköjärjestelmän toimintoja ja ominaisuuksia. Älykamera voi esimerkiksi huolehtia itse valaistuksesta, linssien kohdistuksesta, kalibroinnista ja kuvankaappauksesta. Yksi älykameran määrittlevimmistä komponenteista on edistyne-

seen kuvankäsittelyyn kykenevä integroitu tietokone, jonka avulla kamera pystyy tunnistamaan ja mittaamaan kuvien piirteitä ja ominaisuuksia, sekä kommunikoimaan niiden tietojen pohjalta muiden laitteiden kanssa. Älykairoita käytetään sekä 2D- että 3D-konenäkötekniikoihin perustuvissa sovelluksissa. Älysensori on tiettyyn kuvaus- tekniikkaan keskittynyt älykaierasovellus. (Shi & Real 2010.) Kuvassa 12 nähdään tyypillinen monitoiminen älykamera.



Kuva 12. Cognex Insight-älykamera (Pena 2010).

2.7 Kuvankäsittely ja graafinen esittäminen

Kuvankäsittelyn ja graafisen esittämisen avulla konenäköjärjestelmä ja käyttäjä pystyvät ymmärtämään kamerasaamaa visuaalista tietoa ja toimimaan sen pohjalta halutulla tavalla. Kuvankäsittely vastaa kennon lähettämien analogisten tai digitaalisten signaalitietojen muuntamisesta digitaalseksi kuvaksi ja siitä tehtävästä analyysistä. Graafisella esittämisellä tai kuvan visualisoinnilla tarkoitetaan niitä tapoja, miten kuva esitetään käyttäjälle ja kuinka siihen voidaan vuorovaikuttaa. Käytettäviä kuvatyyppejä ovat esimerkiksi binäärikuvat eli mustavalkokuvat, harmaasävy- ja värikuvat, sekä kolmiulotteiset intensiteetti- ja topografiset mallit (heightmap). (Davies 2005, 20, 22; Hornberg 2006, 492; LMI Technologies 2015, 115-116, 117-118; TKK Automation Technology Laboratory n.d., 8.)

2.7.1 Kuvankaappaus ja 3D-mallin muodostus

Kuvankaappauksella tarkoitetaan kuvan muuntamista ja tallentamista digitaaliseen muotoon. Tämä voi tapahtua erillisellä kuvankaappauskortilla tai tietokoneella. Kuvankaappaustavat jaetaan kuvaustilojen eli kerta- ja jatkuvakuvauksen, sekä niiden käyttämien puskurimuistityyppien perusteella neljään eri tyyppiin: Snap-, Grab-, Sequence- ja Ring-tyyppeihin. Snap- ja Sequence-kuvankaappaustavat ovat kertaluonteisia, joten niiden käynnistämiseksi käytetään asynkronista tiedonsiirtoa eli liipaisu-toimintoja. Grab- ja Ring-tyypit käyttävät synkronista eli jatkuvatoimista kuvaustekniikkaa (videokuvausta). (Hornberg 2006, 481-485, 497-500; TKK Automation Technology Laboratory n.d.)

Digitaalinen kaksiulotteinen kuva tallennetaan tietokoneen muistiin kennon pikseleihin liitettyjen kuvakoordinaattien ja harmaasävyarvojen tai värien, yleensä RGB-arvojen, bittimuotoina. Pikselien XY-kuvakoordinaatiston origo voidaan kuvitteellisesti mieltää kuvan vasempaan ylälaitaan, Y-akselin positiivisen suunnan osoittaessa alaspäin ja X-akselin oikealle. Harmaasävyarvot esitetään joko 8-bittisen eli 0 – 255 tai harvemmin käytetyn 16-bittisen muodon eli 0 – 65535 välisillä arvoilla. Myös 10- tai 12-bittisten kameroiden käyttäminen on alkanut yleistyä. RGB-arvot esitetään kanavittain tyypillisesti kunkin päävärin 8-bittisellä arvolla. (Hornberg 2006, 485-497.)

Kuvankäsittelyohjelmaa tai visualisointia varten tarvitaan kameran lähettämien muistin yksittäisten pikselitietojen uudelleenjärjestämistä kuvaksi eli kuvan rekonstruktiota. Laiteohjain toteuttaa rekonstruktion tallentamalla pikselitiedot DMA:n (Direct Access Memory) kautta oikeisiin kohtiin tietokoneen muistia. (Hornberg 2006, 492-496, Steger ym. 2008, 61-63.) Laiteohjain vastaa myös kuvankaappauksen ohjaamisesta (Hornberg 2006, 475).

Riippuen käytettävästä 3D-kuvaustekniikasta, kolmiulotteisen mallin luominen vastaa pääpiirteittäin kaksiulotteista kuvankaappausta, ainoastaan koordinaattiarvot saavat lisäksi myös Z-akselin mukaisia arvoja. Käytetyistä pikseli- tai vokseliarvoista kartoitetaan ensin niin sanottu 3D-pistepilvi, jolle voidaan kuvan esittämistä varten renderöidä esimerkiksi pintatriangulaatiolla halutunlainen pinta eli 3D-malli. 3D-mallin

muodostus voi olla kertaluonteista eli passiivista tai jatkuvatoimista eli aktiivista. (Szeliski 2010, 584-589.) 3D-mallit esitetään graafisesti usein tietokoneen 3D-mallinnuksina, joita voidaan halutessa kiertää, lähentää ja loitontaa. Myös erilaisten simulaatioiden toteutukset ovat mahdollisia. (Frankling 2008.)

Joskus kuvan laatu ei ole riittävä sen tulkintaa varten, jolloin tarvitaan erilaisia kuvan ehostusmenetelmiä. Näitä ovat esimerkiksi harmaasävymuunnokset, radiometrinen eli ei-lineaarisen intensiteetti-varaussuhteen korjaus ja kuvan tai pinnan tasoitukset (“smoothing”). (Steger ym. 2008, 70-89.)

2.7.2 Kuvan analysointi ja algoritmit

Kuvan analysoinnin tarkoituksena on löytää ja käsitellä kuvassa olevia piirteitä. Tätä tarkoitusta varten käytetään erilaisia matemaattisia algoritmeja. Piirteiden käsittely koostuu yleensä piirteiden paikantamisesta, erottamisesta ja tulkinnasta. Piirteiden paikantamiseen ja erottamiseen voidaan käyttää muun muassa seuraavia perusalgoritmeja: blob-analyysiä eli alueiden tunnistamista, tekstuuri-analyysiä, pattern matching:iä eli hahmontunnistusta ja reunan tunnistusta. Piirteiden tulkinnassa selvitetään eroteltujen piirteiden ominaisuuksia tai suureita, kuten lukumääriä, kulmakiertymiä, pinta-aloja tai pituuksia. (Hornberg 2008, 54-55.)

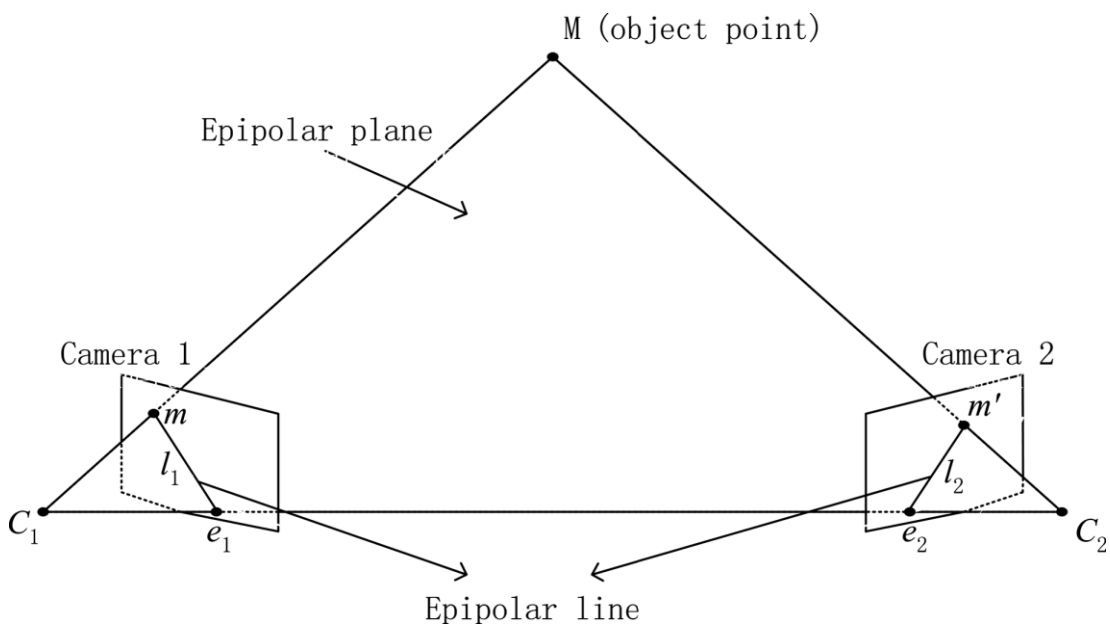
Analysointia voi myös edeltää kuvan esikäsittely, jolla piirteiden paikantamista voidaan helpottaa. Esikäsittelyssä käytettäviä tekniikoita ovat esimerkiksi kynnystys, segmentointi ja gammakorjaus. (Hornberg 2008, 504, 544-550.)

2.8 3D-Kuvaus

3D-kuvauksen tarkoituksena on selvittää syvyysvihjeistä pisteiden etäisyydet kameran ja muodostaa kohteesta 3D-malli. Syvyys määritetään useimmissa 3D-tekniikoissa triangulaatiomittauksella tai hyödyntämällä jotain valon optista ominaisuutta, kuten etenemisnopeutta tai polttoväliä.

2.8.1 Stereokuvaus

Stereokuvauksessa kohteen 3D-malli muodostetaan vähintään kahdesta eri kuvakulmasta otetusta kuvasta. Tämä toteutetaan tyypillisesti kahdella toisistaan kiinteällä etäisyydellä sijaitsevalla kameralla, jotka ottavat vasemman ja oikeanpuoleisen kuvan kohteesta. Stereo-kuvion tunnistuksessa (Stereo matching) etsitään koordinaattimuunnosten ja mahdollisesti mukana käytettävän rakenteellisen valaisukuvion perusteella kuvien vastinpisteet, joista kohteen syvyys ja kuvien yhdistäminen on mahdollista selvittää. Stereokuvaus jäljittelee ihmisen binokulaarista syvyysnäköjärjestelmää. Kuvassa 13 nähdään stereokuvauksen vastinpisteen löytämisen periaate, pisteiden muodostamien epipolaaristen geometrioiden avulla. Vastinpisteen löytämistä ja sen syvyyden eli kamerakoordinaatiston Z-koordinaatin laskemista voidaan helpottaa oikaisemalla kuvatasot matemaattisesti samantasoisiksi. (Hornberg 2008, 653-666.)

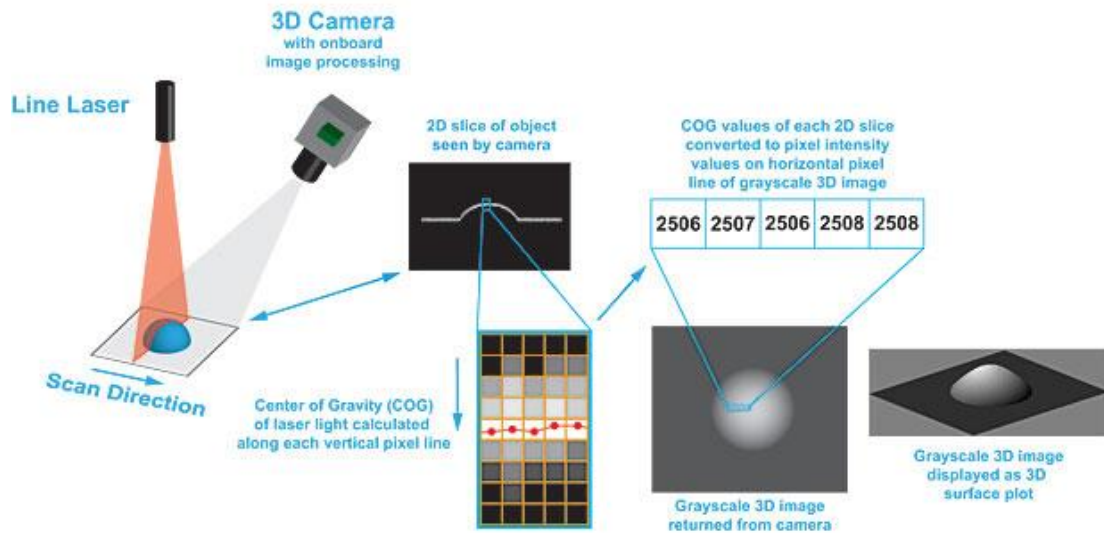


Kuva 13. Kahden kameran stereokuvauksen järjestely (Liu, Liang, J., Liang, X.H. & Tang. 2010).

2.8.2 Rakenteellinen valaisu

Rakenteellisella valaisulla tarkoitetaan 3D-kuvausmenetelmää, jossa projektorilla heijastetaan kappaleeseen ennalta määriteltyjä valo- tai laserkuvioita, jotka kuvakulmasta ja kappaleen muodoista johtuen esiintyvät kameralle vääristyneinä. Vääristymiä ja alkuperäisiä kuvioita vertaamalla, kuvankäsittelyohjelma määrittää mittauspisteiden kolmiulotteiset sijainnit ja muodostaa niiden pohjalta kappaleen 3D-pistepilven tai -mallin. Valokuvioina käytetään esimerkiksi yksinkertaisia piste- ja binääriiviivakuvioita, sinimuotoisia intensiteettiiviivakuvioita tai monimutkaisempia värispektrikuvioita. (Texas Instruments 2016, 2-4.)

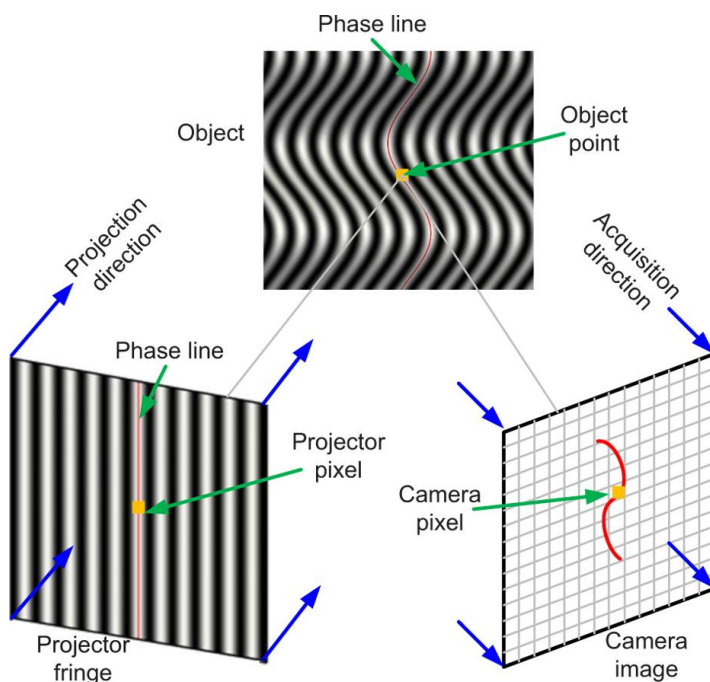
Kun rakenteellisena valaisukuviona käytetään yksittäistä viivakuvioita, puhutaan viivaskannauksesta. Viivaskannauksessa hyödynnetään syvyyden määrittämiseksi triangulaatiomittausta, tyypillisesti käyttämällä laseria valonlähteenä. Kuvassa 14 esitetään 3D-mallin luomisen periaate laserviivaskannauksen avulla. Skannaus alkaa kun laser emittoi laserviivan kohtisuoraan kappaleen pinnalle, jolloin eri korkeudella olevat pinnat heijastavat valoa eri kohtaan kameran matriisikennoa. CMOS- ja CCD-kennoissa viivan tarkan sijainnin määrittämiseksi käytetään intensiteettihiipun etsimisen-, kynnystämisen ja laserviivan Center Of Gravity:n eli COG:n löytämisen algoritmeja. Koska kamera ja siihen heijastuneet laserviivat ovat aina samassa triangulaatiokulmassa emitteriin nähden, voidaan pintojen etäisyydet kameraan laskea trigonometri-sien yhtälöiden tai epipolaaristen geometrioiden avulla. Yhdellä laserin mittauskerralla muodostetaan kohteen sivusta yksi profiilikuva eli niin sanottu "2.5D-malli", ja useampaa profiilia yhdistämällä kohteen pistepilvi ja 3D-malli. (Tarin 2013.)



Kuva 14. Laserviivaskannauksen toiminta CMOS- tai CCD-kennolla (Tarin, 2013).

Laserin tuottaman monokromaattisen valon vuoksi kameroissa voidaan käyttää erilaisia kaistanpäästö- ja polarisaatiosuodattimia, joilla poistetaan häiritseviä heijastumia ja rekisteröidään pelkästään laservalon aallonpituuksia (Alrad n.d., 1-4). Laserskanneri tunnistaa helpoiten vaaleita mattapintaisia esineitä, huonoiten tummia, kiiltäviä ja läpinäkyviä esineitä (Yusuf 2016). Katso kohta 4.1.2, Gocator-mallin laserviivaskannerin toiminnan yksityiskohtaisempaan esittelyyn.

Toinen rakenteellisen valaisun menetelmä on käyttää tunnetun tasokuvion heijastamista kohteen pinnalle. Yhtenä käytetyimpinä tasokuvioina ovat Fringe-kuviot, jotka koostuvat pystysuorista, sinimuotoisista intensiteettiiviivoista. Tietokoneella luotu ja analysoitava Fringe-kuvio heijastetaan projektorilla kuvan 15 mukaisesti kohteen pinnalle, josta kamera kuvaa sen vääristyneet muodot. Lisättäessä kuvioiden määrää saadaan kohteista yhä tarkempia malleja, mutta samalla lisätään kuvauksen kestoa ja tietokoneelta vaadittua prosessointitehoa. Yhden Fringe-kuvion käytössä hyödynnetään niin sanottua Fourierin menetelmää, jossa viivojen vaihe-erot ja niistä johdettavat 3D-koordinaatit lasketaan Fourier muunnosten ja triangulaatiomittauksien avulla. (Karpinsky & Chang 2010, 1-11.)

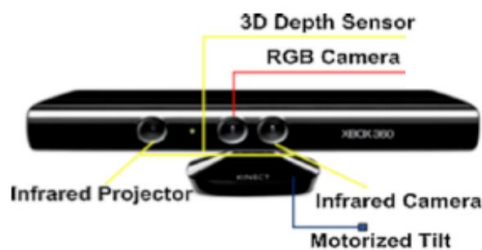


Kuva 15. Fringe-kuvion projisoinnin ja kuvauksen periaate (Karpinsky & Chang 2010b).

Rakenteelliset valaisut kuuluvat Bright Field-valaisutekniikoihin (Hornberg 2006, 165-169).

2.8.3 Kinect

Kinect on Microsoftin alun perin Xbox360-pelikonsolin viihdekäyttöä varten kehitetty 3D-kamera, jolla pystytään tunnistamaan kehon ja kasvojen piirteitä ja liikkeitä, sekä käyttäjien ääniä. Kuvassa 16 näkyvä Kinect koostuu värikamerasta, syvyyskamerasta, neljän mikrofoniryhmästä ja kallistusmoottorista. Koska Kinect'in toiminnan yksityiskohdat eivät ole julkisesti tiedossa, syvyyskameran *arvellaan* hyödyntävän 3D-mallin luomiseksi useaa eri 3D-valaisutekniikkaa, kuten rakenteellista valaisua, Depth from focus-tekniikkaa ja stereokuvausta. Kameran infrapuna-projektori lähettää kuvausalueellensa astigmaattisen linssin läpi ympyrämaisesta soikiomaisiksi muodostuvia pilkkukuviota, jotka eri etäisyydellä olevista kohteista heijastuessaan näyttävät kameralle erilaisilta suuntautumiensa, muotojensa ja sumentumiensa suhteen. Näistä muutoksista päättelemällä Kinect pystyy luomaan ympäristöstään 3D-malleja. Kinect on muihin 3D-kuvaustekniikoihin verrattuna kustannuksiltaan edullinen.

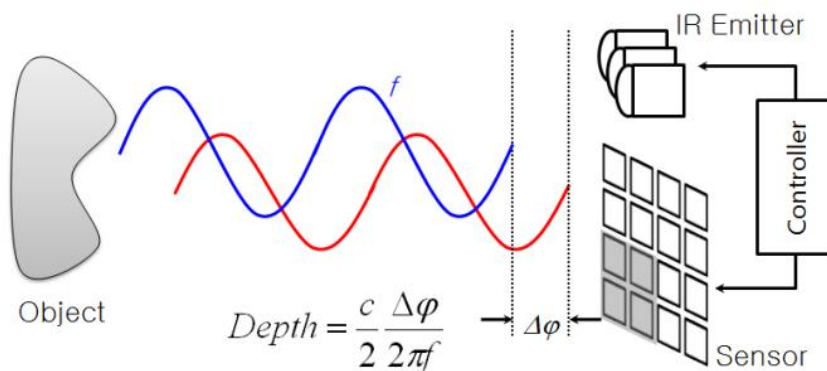


Kuva 16. Microsoft Kinect:in osat ja kameran havaitsema projisoitu pilkkukuvio (Beleboni 2014).

Microsoftin julkaistua Windows:ille Kinect:in ohjelmistokehityspakkauksen (SDK:n), sen käyttömahdollisuudet ovat laajentuneet huomattavasti. Kinect:in sovelluskohteita löytyy esimerkiksi robotiikassa, lääketieteessä ja virtuaalitodellisuudessa. (Beleboni 2014, 1-6; MacCormick 2011, 1-52.)

2.8.4 Time of Flight

Nimensä mukaisesti Time of flight-tekniikka käyttää kohteen syvyyden mittaamiseen valon tai sähkömagneettisen säteilyn etenemiseen kulunutta aikaa. TOF-kamera koostuu tyypillisesti infrapuna-emitteristä ja kamerasta, joilla etenemiseen kulunut aika laskeaan joko lähetetyn ja vastaanotetun signaalin vaiherosta tai elektronisilla sulkijoilla ja digitaalisella laskurilla. Vaihe-eron mittaamiseen perustuvassa menetelmässä lähetetään kohteelle kuvan 17 mukaista jatkuva-aaltoista sinimuotoista signaalia, josta heijastuessaan kameran jokainen pikseli muodostaa neljä toisiinsa 90° vaihe-erossa olevaa ohjaussignaalia. Nämä ohjaussignaalit vastaavat pikselien neljää sähkövarausta, joista lähetetyn ja vastaanotetun signaalin vaihe-ero ja siten syvyys on mahdollista määrittää. Menetelmän heikkoutena on epäselvyys vaiheiden päällekkäistymisessä, joka rajoittaa käytön luotettavuutta etäisyyden kasvaessa liian suureksi. (Hansard, Lee, Choi & Horaud 2012, 8-35.)



Kuva 17. Vaihe-eron mittaamiseen perustuvan Time Of Flight-kameran toiminta (Hansard ym. 2012, 9).

Elektronisen sulkijan käyttöön perustuvassa menetelmässä lähetetään voimakkaita, yksittäisiä laservalopulsseja, jotka heijastuessaan kohteesta kennon APD-diodeille käynnistävät pikselin sulkijatoimisen CMOS-ajastinpiiriin. Ajastinpiiri laskee valonkulkuaan menneen ajan vertaamalla kaikille pikseleille yhteistä aloitussignaalin eli lähetyspulssein lähtöhetkeä sen saapumishetkeen. (Clément, Evangelidis, Hansard & Horaud 2016, 2-5.)

2.8.5 Shape from X ja Shape from focus

Kappaleen 3D-muodon päättelämiseksi voidaan käyttää myös niin sanottuja *monokulaarisia* syvyysvihjeitä, kuten varjostusta, tekstuuria ja kohdistusta. Näitä vihjeitä hyödyntäviä kuvaustekniikoita kutsutaan Shape from X -tekniikoiksi ja yksittäisiä tekniikoita vastaavasti Shape from shading, Shape from texture ja Shape from focus -tekniikoiksi. (Davies 2005, 445-448; Szeliski 2010, 580.)

Shape from focus -tekniikoissa hyödynnetään tietoa kuvien tarkkuuksien ja sumentumien riippuvuudesta kameran kohdistusetaisyteen. Yhden kameran järjestelmässä tämä toteutetaan yleensä muuttamalla kameran kohdistusparametreja samalla pitäen kameran näkökulmaa muuttumattomana. Käytetyimpiä Shape from focus -tekniikoita ovat Depth from focus- ja Depth from defocus -tekniikat. Edellä mainitussa tekniikassa kameran parametreja muutetaan dynaamisesti pinnan määrittämisen aikana, jälkimmäisessä taas ei. Jos tekniikan yhteydessä käytetään apuna myös rakenteellista valaisua,

tarkoitetaan silloin aktiivista Shape from focus-tekniikkaa, muussa tapauksessa passiivista. (Favaro 2002; Szeliski 2010, 606.)

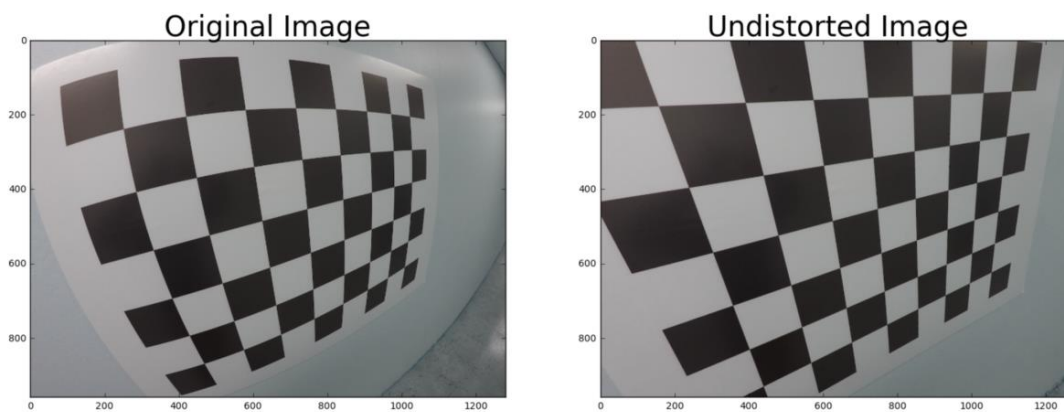
Koska kuvan tarkentumat ja sumentumat muuttuvat sekä kohteen etääntyessä tai lähestyessä kameran DOF:in (Depth Of Field) välillä olevaa polttotasoa kohden, tarvitaan Depth from defocus -tekniikassa kohteen 3D-mallin luomiseen vähintään kahta tai useampaa, eri kohdistuksella otettua kuvaa. Tekniikka voidaan toteuttaa toimimaan myös reaaliaikaisesti, käyttämällä puoliläpäisevää peiliä ja kahta yhteisen optisen rei-tin omaavaa, eri objektiivilla varustettua kameraa. (Szeliski 2010, 606-607.)

2.9 Kalibrointi

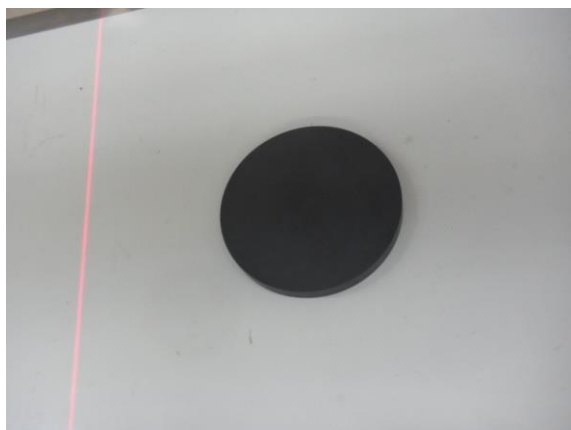
Kalibroinnilla tarkoitetaan konenäössä prosessia, jossa kameran pikselikoordinaatisto muutetaan vastaamaan maailmakoordinaatistoa. Kalibrointi määrittelee siis kameran pikseleissä mitatun etäisyyden suhdetta todelliseen, yleensä metreissä tai tuumissa, mitattuun etäisyyteen ja pikselikoordinaattien sijainteja maailmakoordinaatistossa. Konenäköjärjestelmän kalibrointi suoritetaan kahdessa vaiheessa: kameran sisäisten parametrien säädöllä eli kameran sisäisellä suuntauksella, ja sisäisesti suunnatun kamerakoordinaatiston maailmakoordinaatistomuunnoksella eli ulkoisella suuntauksella. Ulkoista suuntausta tarvitaan liitettäessä konenäköjärjestelmään jokin eri koordinaatistoa hyödyntävä ohjauslaite, kuten teollisuusrobotti tai erotin. Konenäköjärjestelmän kalibrointi suoritetaan useimmissa 2D- ja 3D-konenäkösovelluksissa. (Adaptive Vision 2017; MicroScan 2017b.)

Sisäinen suuntaus on prosessi, jossa luodaan kuvan vääristyneisyyden malli, josta päättelämällä palautetaan kohteesta halutun perspektiivin mukainen vääristelemätön kuva. Kuvan vääristyneisyyden malli käsittää kaikki ne optisista syistä ja kappaleen fyysisistä muodoista johtuvat vääristymät, jotka voidaan korjata kameran sisäisten parametrien säädöillä eli kuvankäsittelyllä. Optisilla vääristymillä tarkoitetaan linssi- ja suodatinvääristymiä kuten tynnyri-, neulatyyny- ja värivääristymiä. Jos kuvasta halutaan erityisesti analysoida kohteen pintaa, voidaan geometriset muodot kuten sylinterimäisten ja pallomaisten kappaleiden pinnat tasoittaa kuvan muotoihin sopiviksi. Ku-

van perspektiivi saadaan oikeanlaiseksi esimerkiksi kiertämällä ja suurentamalla kuvaa tarkasteltavan kappaleen mukaiseksi. Sisäisen suuntauksessa suorittamiseen voidaan käyttää esimerkiksi kuvan 18 mukaista kalibroitinkuviota tai kuvan 19 tapaista kalibroitinesinettä. Kameran kalibroinnilla tarkoitetaan joskus pelkästään kameran sisäistä suuntausta. (Adaptive Vision 2017; MicroScan 2017b.)



Kuva 18. “Shakkilauta”-kalibroitinkuvio ennen ja jälkeen sisäisen suuntauksen suorittamista (Inspro9 2016).



Kuva 19. Laserprofiili-kameran kalibroitinkiekkko.

Yhden kamerajärjestelmän ulkoisessa suuntauksessa tarvitaan sisäisesti suunnatun kamerakoordinaatiston ja ohjauslaitteen käyttämän maailmakoordinaatiston väliset kolme kiertymä- ja kolme siirtymäparametria. Monen kameran järjestelmässä joudutaan ennen ulkoisen suuntauksen suorittamista tekemään lisäksi kamerakoordinaatistojen väliset suhteelliset suuntaukset.

Ulkoisen suuntauksen suorittamiseksi voidaan käyttää esimerkiksi point-to-point-menetelmää, jossa koordinaatistomuunnokset suoritetaan etsimällä riittävä määrä koordinaatistojen välisiä vastinpisteitä. Käytännössä menetelmä tehdään viemällä ohjauslaitteen, kuten robotin työkalu kalibrointipisteisiin, jonka koordinaatteja verrataan kameran antamiin pisteisiin. Näistä pisteiden välisistä eroista pystytään laskemaan koordinaatistojen väliset kiertymät ja paikkavektorit. Monet älykamerat ja niiden kuvankäsittelyohjelmat tarjoavat valmiita ratkaisuja sisäisen ja ulkoisen suuntauksen suorittamiseksi. (Adaptive Vision 2017; MicroScan 2017b.)

3 ROBOTIIKKA

3.1 Robotiikan ja robotin määritelmät

Robotiikalla tarkoitetaan tieteen- ja tekniikanalaa, jonka tarkoituksena on robottien suunnitteleminen, valmistaminen ja soveltaminen (ISO 8373 2012, 2). Robotiikkaan sisältyy muun muassa mekaniikkaa, ohjelmointia, säätötekniikkaa ja konetekniikkaan liittyvien turvallisuusasioiden noudattamista (Aalto ym. 1999, 4).

Robotin määritelmäksi ei ole olemassa mitään yksiselitteistä, yleisesti hyväksyttyä kuvausta. Kansainvälisen standardisoimisjärjestö ISO:n (International Organization for Standardization) mukaan robotiksi määritellään: “Teollisuuden automaatio-sovelluksissa käytettäväksi tarkoitettu automaattisesti ohjattu, uudelleen ohjelmoitavissa oleva monikäyttöinen käsittelylaite, jonka akseleista vähintään kolme on ohjelmoitavissa ja joka voi olla kiinteästi asennettu tai liikkuva”. (ISO 10218-1 2011; Malm ym. 2008, 1-2.) Kuitenkin esimerkiksi Japanissa roboteiksi voidaan laskea myös manipulaattorit eli vain yksinkertaiseen ohjaukseen ja kappaleen siirtotehtäviin soveltuvat laitteet, kuten nosturit ja kauko-ohjattavat robottikäsi- ja varret (Aalto ym. 1999, 181; Malm ym. 2008, 1).

3.2 Teollisuusrobottien historia

Nykykaikaisten teollisuusrobottien aikakauden katsotaan alkaneen George C. Devol'in vuonna 1954 suunnittelema uudelleen ohjelmoitavasta manipulaattorista. Manipulaattorin pohjalta Devol kehitti yhdessä Joseph F. Engelbergerin kanssa General Motors:ille metallivalukappaleita käsitelleen Unimate-robotin, joka oli tällöin ensimmäinen teollisessa tuotannossa käytetty robotti. (Malm ym. 2008, 1.)

Robotit eivät vielä 60- ja 70-luvulla yleistyneet teollisuudessa niiden vähäisen taloudellisen kannattavuuden takia. Tällöin kuitenkin kehitettiin vielä nykyaikaisesti toimivien robottien prototyyppijä, sekä tehtiin merkittäviä muutoksia niiden mekaniikkiin ja toimintoihin. Näitä olivat esimerkiksi kuuden ja sitä suurempien vapausas-

teiden käyttö ja ohjausjärjestelmien muuttaminen analogiatekniikasta mikroprosessori- ja tietokonepohjaiseksi. Lisäksi merkittävät robotteja valmistavat yritykset, kuten KUKA, NACHI ja ABB, aloittivat omien robottimalliensa valmistamisen. Robottien kehitystyö nykyaikaisiksi robottityypeiksi tapahtui enimmäkseen Euroopassa ja Japanissa. (International Federation of Robotics 2017; Malm ym. 2008, 1.)

80-luvulta 90-luvulle teollisuusrobottien kustannukset laskivat ja niiden sovellusmahdollisuudet laajenivat. Robotteja käytettiin nyt tekemään yksitoikkoisia, tarkkuutta vaativia tehtäviä, ihmisille haastavissa tai jopa mahdottomissa olosuhteissa. Robottien käyttäminen lisäsi liukuhihnatuotannon nopeutta. Toisin kuin 1960-luvulla, robotit pyrittiin saada erikoistumaan erilaisiin tehtäviin materiaalin siirrosta maalaamiseen ja hitsaamiseen. Myös ensimmäiset edistyneitä painesensoreita ja konenäköä hyödyntävät robottijärjestelmät tulivat mukaan teolliseen valmistukseen. (Malm ym. 2008, 1; Wallen 2008, 11-14.)

Nykyään robotteja kehitetään teollisuuden lisäksi lääketieteen, pelastus- ja etsintätehtävien, sekä viihde- ja palvelutekniikan käyttöön. Teollisuudessa robotit vähentävät tehtaan kustannuksia, lisäävät tuottoa sekä parantavat tuotteiden laatua. (Wallen 2008, 14-15.)

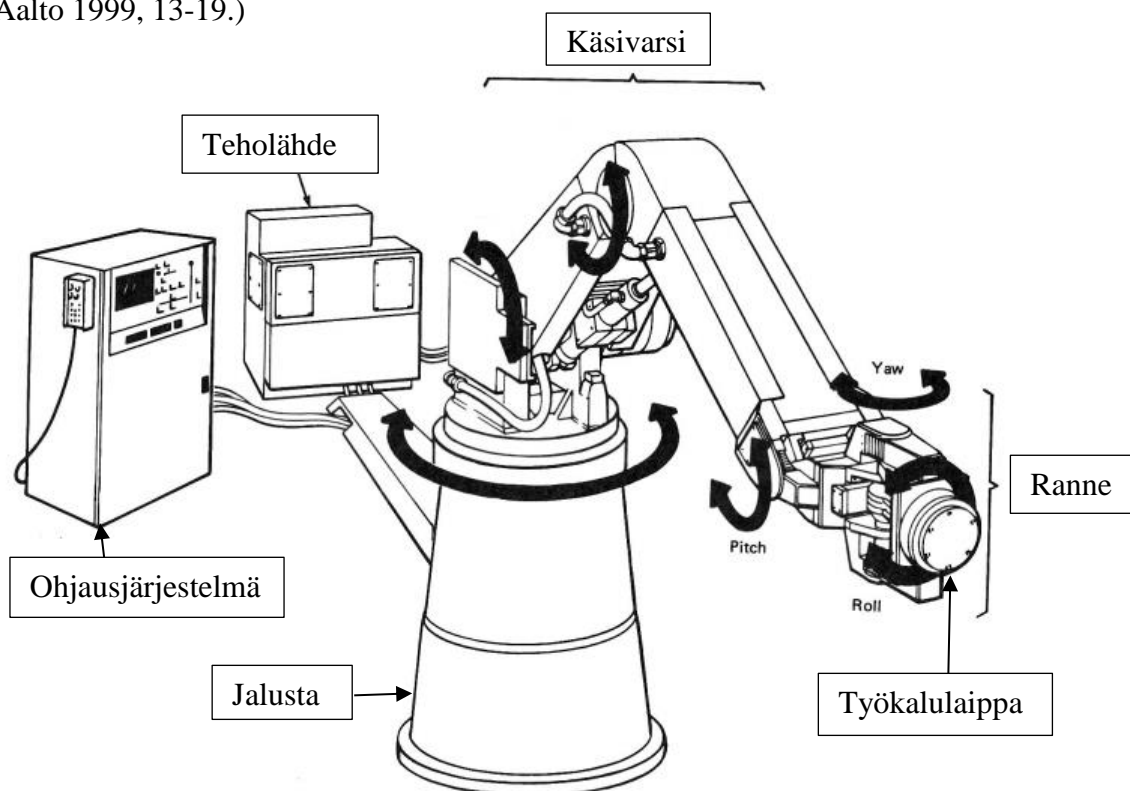
3.3 Standardit ja turvallisuus

Robotteihin sovelletaan sekä yleisiä koneturvallisuusstandardeja että robotin ominaisuuksiin, määrittelyihin ja käyttöturvallisuuteen liittyviä standardeja. ETA-maissa noudatetaan konedirektiivi 2006/42/EY:n tai sen uusimman version mukaisia turvallisuusstandardeja, kuten SFS-EN 10218-1:stä. (Malm ym. 2008, 12–17.) ISO 9787 ja ISO 8373 määrittelevät robotiikan yleisimpiä termejä ja toimintoja (Aalto ym. 1999, 12-15).

3.4 Rakenteet ja osat

Teollisuusrobotit koostuvat kuvan 20 tapaan kahdesta pääkomponentista: mekaanisesta laitteesta ja sen ohjausjärjestelmästä. Mekaaniseksi laitteeksi voidaan käsittää

esimerkiksi jalustaan kiinnitetty robottikäsi tai ohjailtava tukivarsikehikko. (Aalto 1999, 13-19.)



Kuva 20. Käsivarsirobotin komponentit (Korem n.d., 12) (muokattu).

3.4.1 Ohjausjärjestelmä

Ohjausjärjestelmällä tarkoitetaan robotin toimilaitteita ohjaavaa reaaliaikaista prosessitietokonetta. Sen tehtäviin kuuluu muun muassa tiedonsiirtäminen robotin ja muiden järjestelmän laitteiden välillä, robottiohjelmien tulkitseminen liikekäskyiksi toimilaitteille, anturien tilatietojen käsitteleminen takaisinkytkennän aikaansaamiseksi ja itse-diagnosoinnit virhetilojen tunnistamista sekä muita turvallisuustoimintoja varten. Ohjausjärjestelmä koostuu tyypillisesti keskusyksiköstä, massamuistista, robotin teholähteestä ja erilaisista liitäntärajapinnoista. (Aalto ym. 1999, 34-46.)

3.4.2 Mekaaniset osat ja rakenteet

Teollisuusrobottien mekaanisesta liikkeestä vastaavat toimilaitteet ja niiden voimansiirtomekanismit, jotka ohjaavat robotin tukivarsia ja niveliä. Yleisin toimilaitte on ta-

vallisesti jarrulla varustettu tasavirta- tai vaihtovirtaservomoottori. Joissakin tapauksissa voidaan hyödyntää myös askelmoottoreita tai servopiireillä varustettuja pneumaattisia ja hydraulisia toimilaitteita. Voimansiirrossa käytetään työntötankoja, hammashihnoja ja -pyörästäjä, vipuja, kuularuuveja tai muita sopivia mekanismeja. Robotin liike syntyy toimilaitteen muuttaessa voimansiirtomekanismin välityksellä kahden tukivarren asentoa tai asemaa tarkoituksenmukaisesti valitun suoran eli nivelen suhteen. Tämä perusliikkeen mahdollisuus eli vapausaste (DOF, Degree Of Freedom) voi olla suoran suuntaista eli lineaarista tai sen ympärikiertyvää. (Aalto ym. 1999, 15-16, 19-20.)

Robotit jaetaan tukivarsistojensa kytkentätavoista riippuen avoimiin kinemaattisiin rakenteisiin ja suljettuihin kinemaattisiin rakenteisiin. Avoimessa kinemaattisessa rakenteessa robotin tukivarret on kytketty kiinni toisiinsa peräkkäin, kun taas suljetussa kinemaattisessa rakenteessa tukivarret on kytketty rinnakkain. Molemmilla on omat etunsa: avoimella kinemaattisella rakenteella saavutetaan suurempi ulottuvuus ja kulkoketteruus, suljetulla keveys ja rakenteellinen kestävyys. (Aalto ym. 1999, 15-16, 19-20; Hägele, Nilsson & Pires 2007, 14-15.)

3.4.3 Asema-anturit

Robotin tukivarsien välisten asentojen ja asemien mittaamista varten käytetään asemaantureita. Asemaanturien avulla ohjausjärjestelmä pystyy reaaliaikaisesti seuraamaan akselikulman tai lineaarisen liikkeen asemaa, nopeutta ja kiihtyvyyttä. Asemaantureita on kolmenlaisia: inkrementtiantureita eli pulssiantureita, absoluuttiantureita ja resolvejä. (Aalto ym. 1999, 30-33.)

3.4.4 Kaapelit ja työkalut


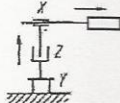


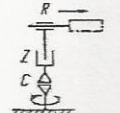




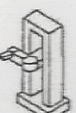
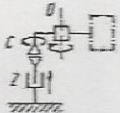


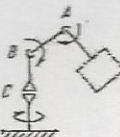


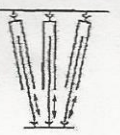

Työkalu on robotin työkalulaippaan kiinnitettävä mekaaninen laite, joka vastaa jostakin robotin oleellisesta tehtävästä, kuten kappaleen poiminnasta tai sen muokkauksesta. Työkalut voidaan jakaa tarraimiin tai tiettyihin prosesseihin tarvittaviin laitteisiin.

Tarraimen pääasiallisia tehtäviä ovat kappaleeseen tarttuminen, kappaleen asennon hallitseminen siirron aikana ja oikeanlainen irrotus. Eräs keino jakaa tarraintyypit eri ryhmiin on määrittää niiden nostovoiman synnyttämiseen käytettävät tavat. Mekaaniset tarraimet hyödyntävät puristuksissa olevan kappaleen ja tarraimen sormien välisiä kitka- ja tukivoimia, tyhjiötarraimet alipainetta ja magneetitarraimet magneettista vuorovaikutusta. Muut tyypit, kuten muovautuvat ja adheesiota hyödyntävät tarraimet, luokitellaan erikoistarraimiksi. (Aalto ym. 1999, 60-77.)

Robotin tarraimeksi tulisi valita tartuttavan kappaleen poimintaominaisuuksien ja robotin yhteensopivuuden kannalta parhaiten soveltuva työkalu. Esimerkiksi tasapintaisen lasin käsittelyä varten on käytännöllisempää valita alipainetarttuja, mutta pyöreän metallitangon poimintaan kaksisormitarttuja soveltuukin paremmin. (Aalto ym. 1999, 64-69.)

3.5 Robottityypit

Teollisuusrobotit voidaan rakenteidensa ja työalueidensa perusteella jakaa kuuteen eri tyyppiin: suorakulmisiin, sylinteri-, napakoordinaatisto-, SCARA-, kiertyvänivelisiin ja rinnakkaisrakenteisiin robotteihin. Kuvassa 21 näytetään nämä yleisimmät robottityypit ja niiden perusominaisuudet.

Nimitys pääakseleiden mukaan	Rakenne	Kinemaattinen kaavio	Työalue
Suorakulmainen robotti			
Sylinterirobotti			
Napa-koordinaatisto-robotti			
Scara-robotti			
Kiertyvänivelinen robotti			
Rinnakkaisrakenteinen robotti			

Kuva 21. Yleisimmät robottityypit rakenteiden mukaan (Aalto ym. 1999, 12).

3.5.1 Suorakulmaiset robotit

Suorakulmaisilla roboteilla (rectangular robots) on yhteistä kolmen ensimmäisen vapausasteen lineaarisuus. Yleisintä mallia kutsutaan portaalirobotiksi (gantry robot). Isoja portaalirobotteja käytetään esimerkiksi logistiikka- ja varastointisovelluksissa kuten kuvan 22 esimerkissä, pienempiä kevyissä työstötehtävissä, esimerkiksi 3D-tulostimien suuttimien ohjauksessa. Portaalirobotin etuja ovat kappaleen paikoitustarkkuus ja helppo ohjelmoitavuus, joka johtuu niiden liikeakselien yhdensuuntaisuuksista

XYZ-koordinaatiston akselien kanssa. (Aalto ym. 1999, 16; Miniweb 2016; RobotWorx 2016.)



Kuva 22. Suuri portaalirobotti (Cimcorp, n.d.).

Suorakulmaista robottia voidaan kutsua myös karteesiseksi (cartesian coordinate robot) tai lineaariseksi robotiksi (linear robot) (RobotWorx 2016).

3.5.2 Sylinterirobotit

Sylinterirobotin (cylindrical robot) nimitys tulee robotin käyttämästä sylinterikoordinaatistosta. Nykyään harvinaisia sylinterirobotteja käytettiin aikaisemmin muun muassa kokoonpano- ja pistehitsaustehtävissä. Sylinterirobottien etuina ovat hyvä soveltuvuus pituusakselinsa suhteen symmetristen kappaleiden käsittelyyn, kuten putkien hitsaamiseen. (Aalto ym. 1999, 16; Robotpark 2016.)

3.5.3 Napakoordinaatistorobotit

Napakoordinaatistorobotit (spherical robots) käyttävät työkalupaikan määrittämiseen nimensä mukaista koordinaatistotyyppiä. Niiden työalue on pallomainen. Napakoordinaatistorobotteja ei käytetä nykyisin enää teollisissa sovelluksissa, johtuen niiden joustamattomuudesta erilaisiin tehtäviin. Historialliselta arvoltaan ne ovat kuitenkin merkittäviä: kuvassa 23 näkyvä maailman ensimmäinen teollisuusrobotti Unimate luokitellaan napakoordinaatistorobotiksi. (All on robots 2016.)

Unimate 2000 series Robot



Kuva 23. Napakoordinaatistorobotti Unimate (Roboticsbible 2016).

3.5.4 Scara-robotit

Scara-robotti eli Selective Compliance Assembly Robot Arm koostuu kuvan 24 tapaan kolmesta kiertyvästä nivelestä ja yhdestä lineaarisesta pystyakselistä. Scara-robotteja käytetään yleensä pienten kuormien tehtävissä, kuten kokoonpanossa, pakkaamisessa ja materiaalinhallinnassa. Vaikka Scara-robotin työprofiili muistuttaa sylinterirobotin työprofiilia, se eroaa huomattavasti siitä rakenteensa ja toimintansa takia. Scara-robotti jäljittelee enemmän ihmisen käden toimintaa kahdelta nivelvarreltaan. Scaran etuja ovat nopeus, hyvä työskentelykyky ahtaissa tiloissa ja korkea toistotarkkuus. (Aalto ym. 1999, 16; Strujik 2011, 189.)



Kuva 24. Scara-robotti akseleineen ja liikkumissuuntineen (Chocholak 2015).

3.5.5 Kiertyväniveliset robotit

Teollisuuden käytetyimmät robottityypit eli kiertyväniveliset robotit (articulated robots) koostuvat noin 4 - 10 nivelestä, joista kaikki ovat kiertyviä. Tämän ansiosta kiertyväniveliset robotit omaavat laajan työskentelyalueen ja monipuoliset käyttömahdollisuudet. Kiertyvänivelisiä robotteja käytetäänkin lukuisissa erilaisissa kappaleenkäsittely- ja prosessitehtävissä, kuten hitsauksessa, maalauksessa ja palletoinnissa. Tyyppillinen kuuden nivelen kiertyvänivelinen robotti nähdään kuvassa 25. (Aalto ym. 1999; RobotWorx 2016b.)



Kuva 25. Kuuden vapausasteen kiertyvänivelinen robotti (Robotics Automation Systems 2016).

3.5.6 Rinnakkaisrakenteiset robotit

Rinnakkaisrakenteisiksi roboteiksi (parallel robots) luetaan kaikki suljettua kinemaattista rakennetta käyttävät robotit, kuten kuvan 26 Delta-robotti tai teleskooppijalkainen “Stewartin alusta”. Yhteistä rinnakkaisrakenteisille roboteille on monien nivelien passiivisuus eli niiden ohjaaminen epäsuorasti päätukivarsien avulla. Rinnakkaisrakenteiset robotit ovat painoonsa nähden kestäviä ja erityisesti Delta-robotit soveltuvat nopeisiin ja tarkkuutta vaativiin pick-and-place tehtäviin. (Aalto ym. 1999; Mecademic Inc. 2017.)



Kuva 26. ABB:n rinnakkaisrakenteinen Delta-robotti (Direct Industry, viitattu 2017).

3.5.7 Erikoisrobotit

Teollisuuden erikoisrobotteja ovat esimerkiksi erilaiset liikkuvat robotit, kuten kuvan 27 rengasliikkeiset vihivaunut (AGV eli Automated Guided Vehicle), sähkömagneettiliikkeiset robotit ja muut liikkuva-alusteiset teollisuusrobotit. Myös palvelurobotit ja humanoidirobotit voidaan lukea erikoisrobotteihin, vaikka niillä ei vielä ole huomattavaa käyttöä teollisuudessa. (Aalto ym. 1999 18 -19; Egemin Group NV, 2016; Malm ym. 2008, E1 – E29.)



Kuva 27. Laser-skannerilla varustettu vihivaunurobotti (Wheelomania 2011).

3.5.8 Yhteistyörobotti

Yhteistyörobotilla tarkoitetaan ihmisen ja robotin väliseen, turvalliseen vuorovaikutteiseen työskentelyyn kykenevää robottia. Nykyaikainen turvallisuus- ja anturitekniikka mahdollistavat robotin ja ihmisen työskentelyalueiden osittaisen yhdistämisen. Verrattuna robotin työskentelyalueen turva-aidoittamiseen, ihminen-robotti-yhteistyö lisää tuotantojärjestelmien mukautumiskykyä eri tuotevarianteille, parantaa tuotteiden laatua ja lisää tuotannon tehokkuutta. Ihmisen havaitsemiseksi yhteistyörobotti voi hyödyntää erilaisia konenäkö- ja optisia ilmaisimenmenetelmiä, ultra-ääniantureita, RFID-paikannusta tai voima- ja paineantureita, kuten keinoihoa. (Malm ym. 2008, 2, 18-35.)

Kaikissa teollisuusroboteissa on nykyään sisäänrakennetut törmäysentunnistimet, jotka tarkkailevat toimilaitteiden virran nousua tukivarsiin kohdistuvien vääntöjen ja voimien tunnistamiseksi. Jos ihmisen tunnistaminen perustuu ainoastaan voima-anturien käyttöön, yhteistyökykyisen robotin on oltava turvallisen törmäyksen mahdollistamiseksi riittävän pienimassainen, hidasliikkeinen ja sisällettävä tarpeeksi herkkiä voima-antureita. Robotissa, työkalussa ja sen kantamassa esineessä ei saa myöskään olla ihmisille vaarallisia osia ja kohtia, kuten teräviä kulmia. Hyvä yhteistyökykyinen robottiratkaisu pystyy turvallisesti välttämään yhteentörmäyksiä hidastamalla liikettä, pysähtymällä tai muuttamalla liikerataa ihmisen ollessa kontaktietäisyydellä. (Malm ym. 2008, 29-30, 38-39.)

3.6 Toiminta ja kinematiikka

Teollisuusrobotin tarkoituksena on hallita työkalua ja sen asemaa prosessissa halutulla tavalla. Työkalunsiirtoa varten tulee tietää työkalun lähtöasema ja kohdeasema peruskordinaatistossa. Lähtöaseman selvittämistä robotin vapausasteiden asennoista, kutsutaan robotin kinemaattiseksi tehtäväksi. Sopivien vapausasteiden selvittämistä työkalun saattamiseksi kohdeasemaansa, kutsutaan käänteiseksi kinemaattiseksi tehtäväksi. (Aalto ym. 1999, 20.)

3.6.1 Koordinaatistot

Teollisuusrobotit käyttävät yleensä kolmea eri koordinaatistoa: maailmakoordinaatistoa, peruskordinaatistoa ja työkalukoordinaatistoa. Kyseiset koordinaatistot ovat “suorakulmaisia ortonomeerattuja oikeakätisiä koordinaatistoja”. Maailmakoordinaatistolla tarkoitetaan yleensä robotista erillään sijaitsevaa, työympäristön tiettyyn kohtaan tai kohteeseen kuten kuljettimeen sijoitettua koordinaatistoa. Peruskordinaatisto on robotin jalustaan kiinnitetty koordinaatisto jossa “Z-akseli yhtyy ensimmäisen vapausasteen akseliin, X-akseli osoittaa ensimmäisen nivelen työalueen keskikohtaan ja XY-taso yhtyy lattiaan”. Työkalukoordinaatisto määritetään yleensä työkalulaipan keskikohdasta (englanniksi TCP eli Tool Center Point) työkalumäärittelyllä eli niin sanotulla työkalun offset:illä, haluttuun kohtaan työkalua tai tarraimessa olevaa kappaletta. (Aalto ym. 1999, 22-28.)

Tärkeä matemaattinen menetelmä robotin kinemaattisten tehtävien ratkaisemiseksi on homogeenisten siirromatriisien käyttö. Siirromatriisin avulla voidaan koordinaatiston sijainti ja asento esittää jonkin toisen koordinaatiston suhteen, esimerkiksi työkalukoordinaatisto suhteessa peruskordinaatistoon tai peruskordinaatisto suhteessa maailmakoordinaatistoon. Käytettävien koordinaatistojen ei tarvitse rajoittua vain robotin kolmeen peruskordinaatistoon, vaan myös muiden soluyksiköiden kuten kokenäköjärjestelmien koordinaatistot soveltuvat hyvin matriisimuunnoksiin (vrt. kokenäköjärjestelmän kalibrointi). Usean matriisimuunnoksen eli tässä tapauksessa koordinaatistomuunnoksien tekemisessä, koordinaatistojen siirromatriisit ketjutetaan

eli kerrotaan peräkkäin, kunnes haluttu koordinaatisto tai sen piste (yleensä työkalupiste) on saatu esitettyä halutussa koordinaatistossa. 3D-kamerajärjestelmän koordinaatisto esitetään peruskoordinaatistossa seuraavanlaisesti:

$$PerusT_{Kamera} = \begin{bmatrix} x_x & y_x & z_x & p_x \\ x_y & y_y & z_y & p_y \\ z_z & y_z & z_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \bar{x} & \bar{y} & \bar{z} & \bar{p} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \text{ josta koordinaattipisteeksi saadaan:}$$

$$Perus\bar{p} = PerusT_{Kamera} * Kamera\bar{p} = \begin{bmatrix} x_x & y_x & z_x & p_x \\ x_y & y_y & z_y & p_y \\ z_z & y_z & z_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Siirromatriisin XYZ-muuttujien arvot ovat koordinaatistojen akseleiden kiertymiä ilmaisevia trigonometrisiä funktioita, p-muuttujat akselittaisia paikkasiirtymiä ja vakioarvot mahdollistavat matriisien keskinäisen kertomisen. (Aalto ym. 1999, 22-28.)

Liitteessä 2 esitetään menetelmän käyttäminen kameran ulkoisen suuntauksen suorittamiseksi.

3.6.2 Työkalun siirtäminen

Työkalunsiirto alkaa ohjelman liikekäskyllä, jolla määritetään työkalun etenemistapa. Liikekäskyjä ovat esimerkiksi liikkeet tiettyihin peruskoordinaatiston asemiin ja suhteelliset liikkeet perus- tai työkalukoordinaatiston akseleihin nähden. Liikekäskystä ohjelmaa lukeva tulkki käynnistää interpolaattorin, joka määrittää robotin liikeradan laskemalla lähtöaseman ja kohdeaseman välisiä välikohdeasemia. Asema-arvot muunnetaan servo-ohjainten avulla toimilaitteiden PI-säätöisiksi virran- ja tehon ohjearvoiksi, kunnes työkalu saavuttaa kohdeasemansa riittävällä tarkkuudella. Käytetty interpolaatioliiketapa määrää onko eteneminen esimerkiksi lineaarista tai ympyräliikettä. Taulukossa 3 esitetään vielä robotin toiminnan yksityiskohtiin liittyvien ominaisuuksien ja ongelmien selityksiä. (Aalto ym. 1999, 20-22, 34-40.)

Taulukko 3. Robotin toimintaan liittyviä termejä ja niiden selityksiä (Aalto ym. 1999, 14-40).

Termi	Selitys
Absoluuttitarkkuus	Laskennallisen pisteen ja todellisesti saavutetun pisteen välinen ero peruskoordinaatistossa.
Toistotarkkuus	Aikaisemmin ohjelmoidun pisteen ja siihen palatun pisteen välinen keskimääräinen ero.
Kaksoismerkitysongelma	Samaan kohdeasemaan pääsemiseksi robotilla on valittavana useita erilaisia asentoja.
Singulariteettiongelma	Vapausasteen käytön menettäminen tietyissä, yleensä tukivarsien yhtenevissä asennoissa.
Sisäisen paikan mittauksen ongelma	Tukivarsien taipumisesta johtuva paikkavirhe, jota robotti ei ota huomioon.

3.7 Ohjelmointi

Robottien ohjelmointitavat jaetaan yleensä kahteen pääluokkaan: off-line- ja on-line-ohjelmointeihin. Off-line-ohjelmoinnissa robotin ohjelmointi suoritetaan ilman oikean robotin osallisuutta ohjelmoinnin tekoon eli etäohjelmoimalla. Tämä voidaan toteuttaa esimerkiksi käyttämällä robotin graafista 3D-mallia simulointitilassa tai tekstipohjaista etäohjelmointia. On-line-ohjelmoinnilla tarkoitetaan fyysisen robotin hyödyntämistä ohjelmoinnissa, kuten tallentamalla muistiin robotin asentoja, liikeratoja tai koordinaattipisteitä todellisessa ympäristössä. Taulukossa 4. näytetään yleisimpiä ohjelmointimenetelmiä eri pääluokissaan. Huomaa, että jotkin ohjelmointitavat ovat hyvin lähellä toisiaan, mutta eroavat lähinnä oikean robotin tai konenäön osallisuudella itse ohjelmointiin. Off-line-ohjelmoinnissa tulee aina lopuksi suorittaa kalibrointi ohjelman ja todellisen maailman välillä. (Malm ym. 2008, 95-99.)

Taulukko 4. Robottien ohjelmointitavat (Malm ym. 2008, 95-99).

ON-Line-ohjelmointi	OFF-Line-ohjelmointi
Opettamalla ohjelmointi	Graafinen etäohjelmointi
Näyttämällä ohjelmointi	Automaattinen etäohjelmointi
Konenäköpohjainen ohjelmointi	“Etäohjaimella” ohjelmointi
Tekstipohjainen ohjelmointi	Tekstipohjainen etäohjelmointi
Oliopohjainen ohjelmointi	Oliopohjainen etäohjelmointi

4 GOCATOR 2380 3D-PROFIILIKAMERA

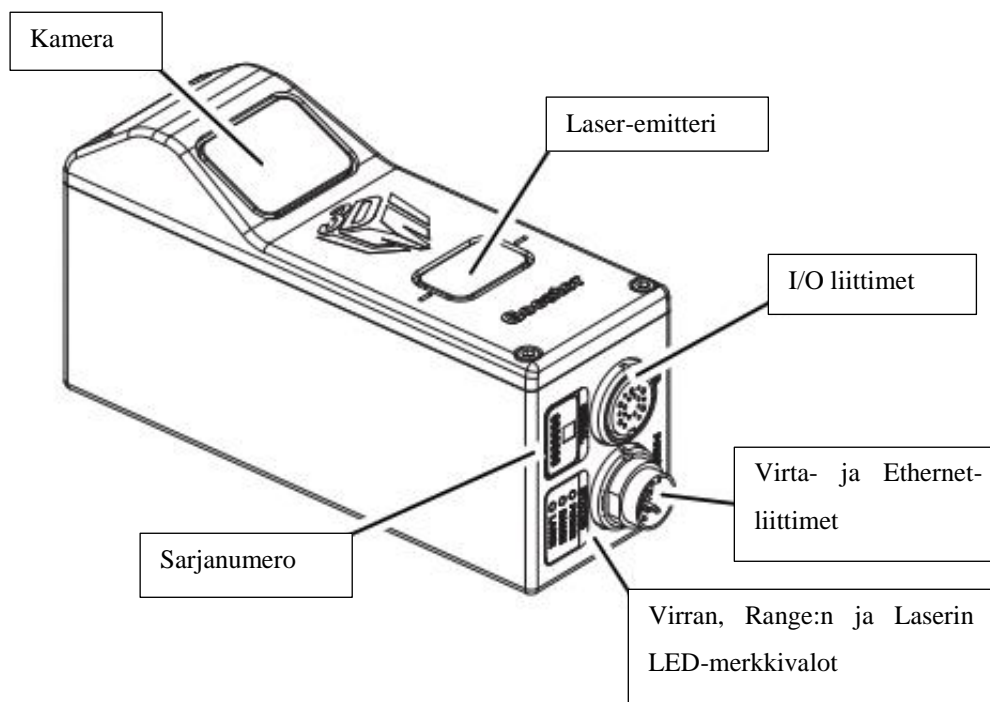
4.1 Esittely ja käyttötarkoitus

Gocator 2380 kuuluu laserprofiilia hyödyntäviin Gocator 2300-sarjan 3D-älykameroihin. Sarjan kameroita käytetään mittaus- ja ohjaussovelluksissa. Gocator 2380 voidaan liittää useisiin eri tulo- ja lähtölaitteisiin. Konfiguraatio toteutetaan Ethernet-yhteydellä, internet-selaimen välityksellä. Käyttäjän manuaalista löytyvät kattavat englanninkieliset ohjeet kameran asennukseen (*Installation*, sivut 25-29), kalibrointiin (*Network setup*, sivut 30-36), konfiguraation ja käyttöön (*Gocator Web Interface*, sivut 42-92 ja 97-192). (LMI Technologies 2015, 1-10.)

Liitteessä 3 esitetään 2380:n skannauksen pikakäyttöohjeet liukuhihnasovelluksessa, sekä laitteeseen liittyvät yleiset laser- ja sähköturvallisuusasiat suomenkielellä.

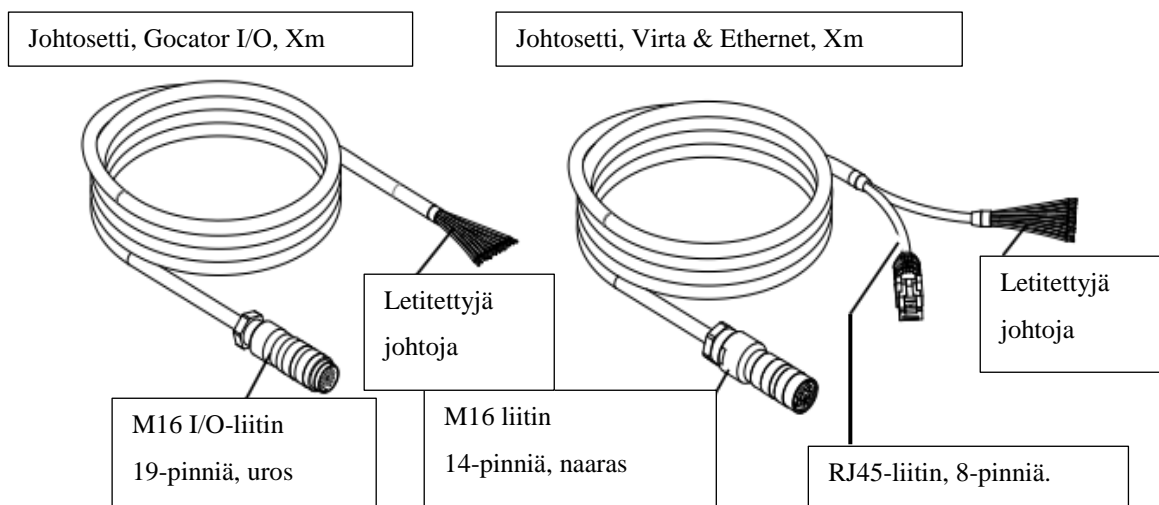
4.1.1 Komponentit ja yleisnäky

Kuvassa 28 nähdään Gocator 2300-sarjaan kuuluvan Gocator 2330-kameran kiinteät komponentit. Kamera vastaa komponenteiltaan Gocator 2380:n komponentteja. Laseremitterinä toimii turvallisuusluokan 3B puolijohdelaser. Käytetty kenno- ja linssiteknikka ei ole yleisesti tiedossa. Virran (Power) LED-merkkivalo loistaa sinisenä kameran virransyötön ollessa kytkettynä päälle, Range-merkkivalo vihreänä kameran tunnistaessa kohdealueellansa laser-valoa ja Laser-merkkivalo keltaisena turvatulon ollessa aktiivisena.

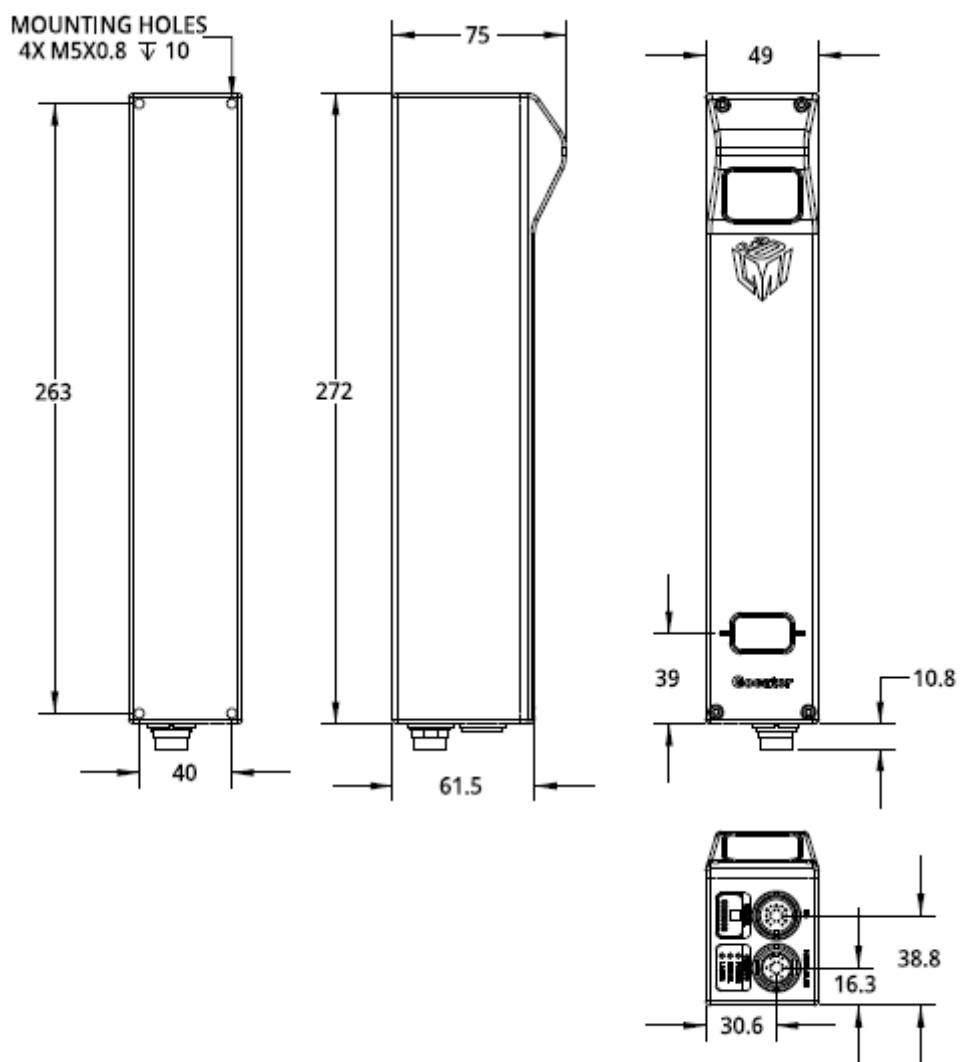


Kuva 28. Gocator 2300-sarjan kameroiden kiinteät komponentit (LMI Technologies 2015, 20) (muokattu).

Gocator 2300-sarjan kamerat käyttävät kahdenlaisia johtosettejä: virta ja Ethernet-kaapeliyhdistelmää sekä I/O-kaapelia. Virta- ja Ethernet-johdot välittävät virran ja laserin turvakytken, sekä tarjoavat tiedonsiirtoon 1000 Mbit/s Ethernetin, RJ-45 standardiliittimellä. I/O-kaapelista löytyvät input/output-yhteydet, liitännät enkooderille, RS-485-sarjaliitännälle ja analogilähdölle. Kuvassa 29 näytetään johtosetit ja niiden liittinosat, kuvassa 30 yleisnäkymä Gocator 2380:sta ja sen mitoista. (LMI Technologies 2015, 20.)

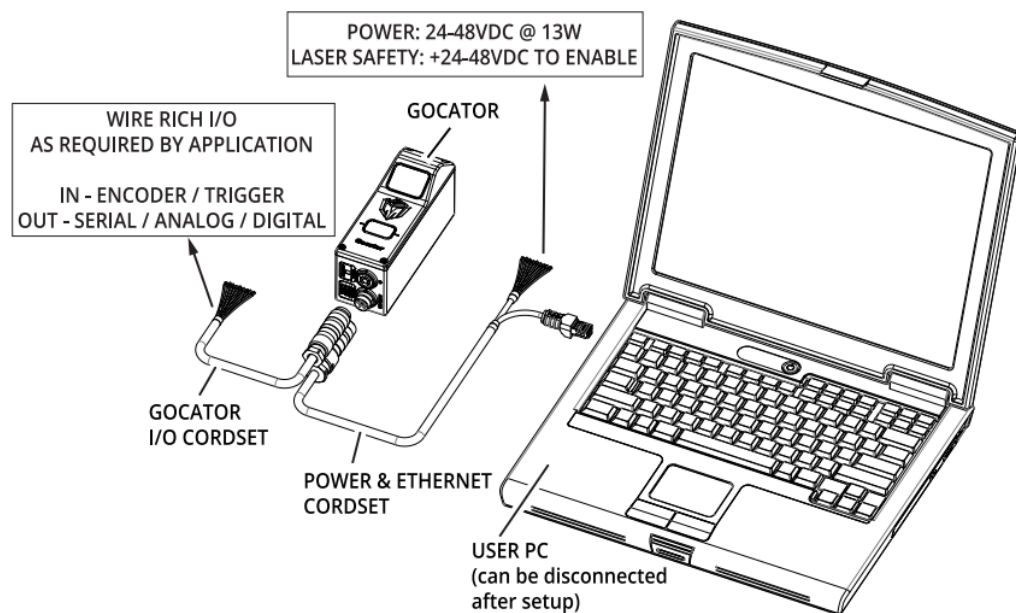


Kuva 29. Gocator 2300-sarjan johtosetit (LMI Technologies 2015, 20) (muokattu).



Kuva 30. Gocator 2380:n yleisnäkymä. (LMI Technologies 2015, 336).

Gocator 2300-sarjan kameroista muodostetaan käytettävien kameroiden lukumäärän perusteella kolmenlaisia konenäköjärjestelmiä: itsenäisiä, kaksois- ja monikamerajärjestelmiä. Kuvan 31 itsenäistä järjestelmää käytetään yhden kameras ja PC:n kanssa. PC voidaan irrottaa järjestelmästä sen jälkeen kun kamera on asetettu ja valmis automaattiseen tiedonsiirtoon ja toimintaan muiden laitteiden kanssa.

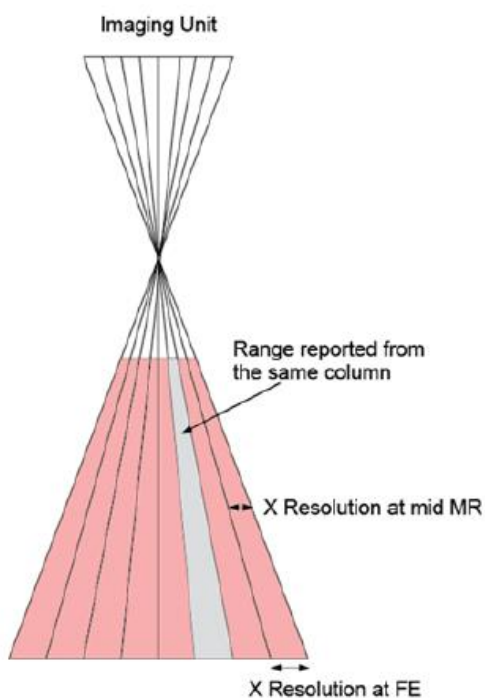


Kuva 31. Itsenäinen Gocator-järjestelmä (LMI Technologies 2015, 17).

Kaksoisjärjestelmässä käytetään kahta kameraa ja monijärjestelmässä kolmea tai useampaa kameraa. Näissä järjestelmissä kamerat asennetaan eri kulmiin kappaleeseen nähden, jolloin kappaleesta saadaan rakenteellisen valaisun periaatteiden mukaisesti niin sanottuja “aitoja” 3D-kuvia, yhdellä läpivalaisukerralla. Molemmissa vaaditaan kameran lisäksi yksi tai useampi Gocator-Master ja Master-johtosetit kameraen ja laitteiden välisen kommunikaation ja yhteisen profiilikuvan muodostamiseksi. Monijärjestelmässä vaaditaan lisäksi käyttäjä-PC. (LMI Technologies 2015, 17-21.)

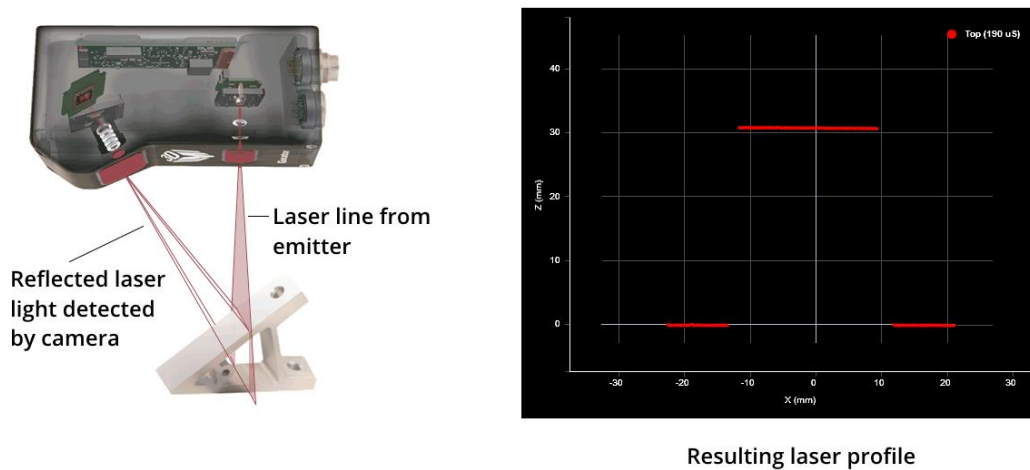
4.1.2 Toiminta

Gocatorin toiminta perustuu rakenteellisen valaisun lasertriangulaatiomittaukseen (ks. kohta 2.9.2 Rakenteellinen valaisu). Valaistavan kappaleen tulee sijaita kameran mitta-alueella (Measurement Range eli MR) eli välyksen (Clearance Distance) ja loppupään (Far End) välillä. Kamera mittaa laser-profiiliviivasta mittauspisteitä, joiden laserserviivan suuntaiset etäisyyserot määrittävät kuvan 32 mukaisesti kameran X-akselin suuntaisen resoluution. Z-akselin suuntaiseen resoluutioon vaikuttavat kameran kuvantaminen ja elektronikka. Mitä lähempänä kohde on laseria ja mitä paremmin kamera tietää Y-akselin suuntaisen suhteellisen etenemisnopeuden, sitä tarkemman 3D-mallin Gocator pystyy muodostamaan. (LMI Technologies 2015, 42-44.)



Kuva 32. Gocator:in mittausväli ja X-resoluutio (LMI Technologies 2015, 43).

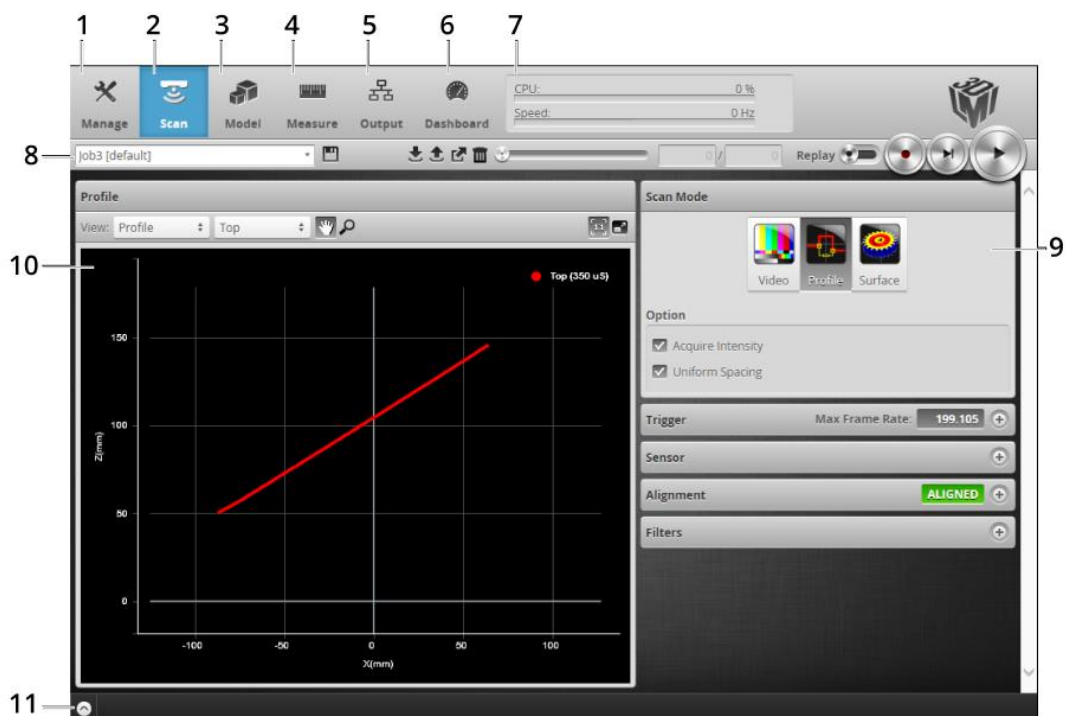
Kuvassa 33 näytetään kohteesta muodostettu yksittäinen laser-profilisiivu, joita yhdistelemällä Gocator rakentaa kohteen 3D-mallin (LMI Technologies 2015, 42).



Kuva 33. Gocator:in kohteesta muodostama profilikuva (LMI Technologies 2015, 42).

4.2 Käyttöliittymä

Kuvassa 34 näytetään Gocatorin selaimella avatun käyttöliittymän yleisnäkymä. Numeroselytykset löytyvät tekstistä kuvan 34 jälkeen. Katso liitteestä 3 ohjeet käyttöliittymän avaamiseen ja manuaaliseen skannaukseen.



Kuva 34. Gocator Web Interface (LMI Technologies 2015, 47).

1. *Manage*-sivu sisältää asetukset kamerasysteemin valinnalle ja asettamiselle, verkkoyhteyksille, enkooderille tai kameran suhteelliselle etenemisnopeudelle (Motion and Alignment), job:ien hallinnalle ja ohjelmiston ylläpidolle.
2. *Scan*-sivulta (näkyvä valittuna kuvassa) löytyvät asetukset skannaustilalle, trigger:ille, yksityiskohtaiselle konfiguraatiolle ja kohdistuksen toiminnalle (alignment).
3. *Model*-sivulla asetetaan kappaleen tunnistaminen (Part matching).
4. *Measure*-sivu sisältää mittaustyökalut ja niiden asetukset.
5. *Output*-sivu sisältää asetukset lähtöprotokollien konfiguroimiseksi, joita käytetään mittaustietojen lähettämiseksi ulkoisille laitteille.
6. *Dashboard*-sivulla voidaan monitoroida mittaustietoja ja kameran tiloja.
7. *CPU Load and Speed*-palkki näyttää kameran suorituskyvyn prosentuaalisia arvoja.

8. *Toolbar*-palkilla ohjataan kameran toimintaa, hallitaan job:eja ja toistetaan tallennettuja mittaustietoja.
9. *Configuration area*-ikkunassa ohjataan skannauksen konfiguraatiota ja mittaustyökalujen asetuksia.
10. *Data viewer*-ikkunassa näytetään kameran tietoja ja mittauksia sekä hallitaan työkalujen asetuksia.
11. *Log*-valikko esittää kameran tilaviestejä, kuten virheilmoituksia, varoituksia ja muuta tietoja. (LMI Technologies 2015, 47-48.)

4.3 Tiedonsiirto

Gocator:illa on mahdollista välittää tietoa laitteille neljällä eri tiedonsiirtomenetelmällä: Ethernet:illä, digitaali- ja analogisignaaleilla sekä RS-485 sarjaliikenneväylällä. Ethernet:in yhteyskäytäntöjä ovat Gocator- (TCP), ASCII-, Modbus TCP- ja Ethernet TCP/IP-protokollat. Yhteyksien lähdöt asetetaan *Output*-sivulla. Gocator:in käyttöliittymään saadaan tietokoneella muodostettua aina kuitenkin Ethernet-yhteys. (LMI Technologies 2015, 209-219.)

Modbus TCP-protokolla soveltuu mittaus- ja kameratietojen välittämiseen, job:iien eli Gocator:in asetustietojen vaihtamiseen, sekä skannauksen ja sisäisen kohdistuksen ohjaamiseen. Gocatoriin eli järjestelmän serveriin pystytään yhdistämään kahdeksan samanaikaisesti toimivaa Clientia, kuten PLC:eitä tai muita I/O-laitteita. Viestien lähettämiseksi, kullekin tulo- ja lähtöviestille osoitetaan/valitaan Register-kohdasta omat 16-bittiset rekisteriosoitteet. 32-bittiset mittaustiedot jaetaan kahdelle, välillä 1000 – 1060 olevalle vierekkäiselle rekisteriosoitteelle. Kuvassa 35 esitetään tärkeimpiä rekisteriosoitteita ja niiden toimintoja. (LMI Technologies 2015, 342-347.)

Register Map Overview

Register Address	Name	Read/Write	Description
0 - 124	Control Registers	WO	Registers for Modbus commands. See <i>Control Registers</i> below for detailed descriptions.
300 - 899	Sensor States	RO	Report sensor states. See <i>State</i> on the next page for detailed descriptions.
900 - 999	Stamps	RO	Return stamps associated with each profile or surface. See <i>State</i> on the next page for detailed descriptions.
1000 - 1060	Measurements & Decisions	RO	20 measurement and decision pairs. See <i>Measurement Registers</i> on page 346 for detailed descriptions.

Command Register Values

Value	Name	Description
0	Stop running	Stop the sensor. No effect if sensor is already stopped.
1	Start Running	Start the sensor. No effect if sensor is already started.
2	Align (stationary target)	Start the alignment process. State register 301 will be set to 1 (busy)
3	Align (moving target)	Start alignment process and also calibrate encoder resolution. State register 301 will be set to 1 (busy) until the motion calibration process is complete.
4	Clear Alignment	Clear the alignment.
5	Load Job	Activate a job file. Registers 1 - 21 specify the filename.

State Register Map

Register Address	Name	Type	Description
300	Stopped / Running		Sensor State: 0 - Stopped 1 - Running
301	Busy		Busy State: 0 - Not busy 1 - Busy Registers 302 to 363 below are only valid when the Busy State is not Busy
302	Alignment State		Current Alignment State: 0 - Not aligned 1 - Aligned
303 - 306	Encoder Value	64s	Current Encoder value (ticks).
307 - 310	Time	64s	Current time (μ s).
311	Job File Name Length	16u	Number of characters in the current job file name.
312 - 371	Live Job Name		Current job Name. Name of currently loaded job file. Does not include the extension. Each 16-bit register contains a single character.

Stamp Register Map

Register Address	Name	Type	Description
976	Buffer Advance		If buffering is enabled this address must be read by the PLC Modbus client first to advance the buffer. After the buffer advance read operation, the Modbus client can read the updated Measurements & Decisions in addresses 1000-1060.
977	Buffer Counter		Number of buffered messages currently in the queue.
978	Buffer Overflow		Buffer Overflow Indicator: 0 - No overflow 1 - Overflow

Kuva 35. Modbus rekisteriosoitteita (LMI Technologies 2015, 344-346) (muokattu).

Jos Gocatorin IP-osoite tai salasana on unohtunut, voidaan kamera löytää verkosta uudelleen tai/ja palauttaa takaisin tehdasasetuksiinsa Sensor Discovery Toolin avulla. Asentamista ja käyttöä varten ks. Gocator User Manual:in kohta *Sensor Recovery Tool*. (LMI Technologies 2015, 273-274.)

4.4 Tekniset tiedot

Kuvasta 36 löytyvät Gocator 2380:n yleiset tekniset tiedot.

MODEL	2380
Data Points / Profile	1280
Linearity Z (+/- % of MR)	0.04
Resolution Z (mm)	0.092-0.488
Resolution X (mm) (Profile Data Interval)	0.375 - 1.100
Repeatability Z (μm)	12
Clearance Distance (CD) (mm)	350
Measurement Range (MR) (mm)	800
Field of View (FOV) (mm)	390 - 1260
Recommended Laser Class	3B
Other Laser Classes	
Dimensions (mm)	49x75x272
Weight (kg)	1.3

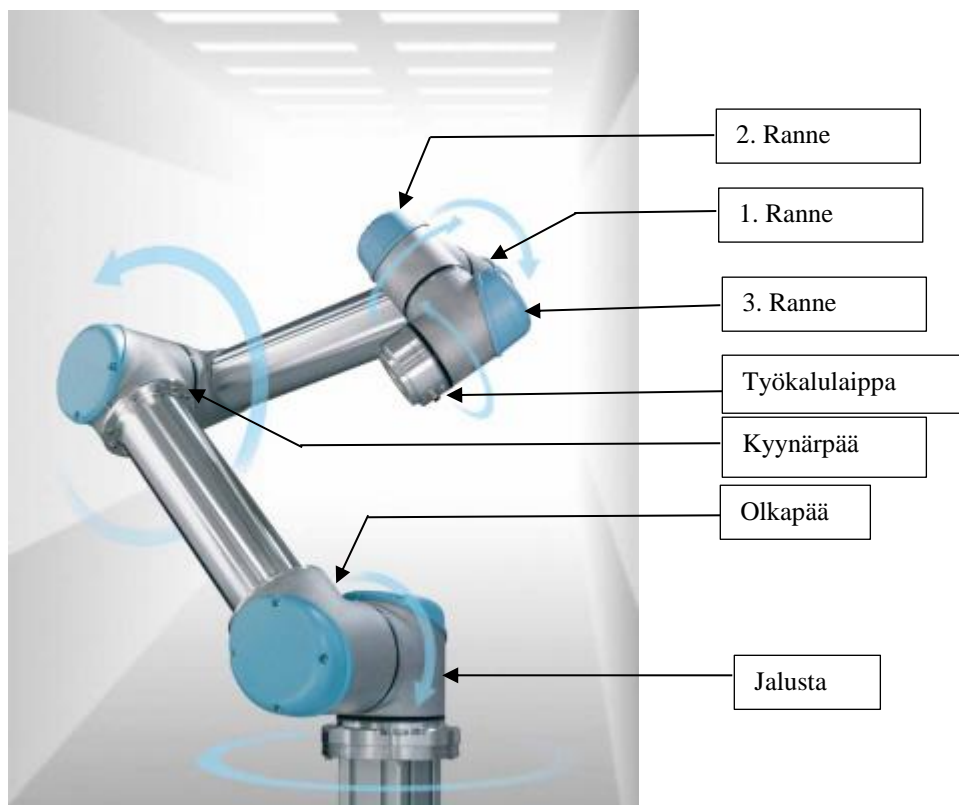
Kuva 36. Gocator tekniset tiedot (LMI Technologies 2015, 385) (muokattu).

5 UNIVERSAL ROBOT UR5 KÄSIVARSIROBOTTI

5.1 Yleisesitys

Universal Robot UR5 on ihmisen ja robotin väliseen yhteistyöhön soveltuva kuuden vapausasteen käsivarsirobotti. UR5 kuuluu sarjaan yleiskäyttöisiä teollisuusrobotteja, joiden numeroliite vastaa niiden enimmäiskantokykyä kilogrammoina. UR5 kantokyky on siis noin 50 N. UR5 sisältää alustaan kiinnitetyn käsivarren, ohjausjärjestelmän (CB3) ja käsiohjaimen eli pendantin. Käsivarsi koostuu lähinnä kevyistä alumiiniputkista ja -nivelistä, toimilaitteista ja niiden voimansiirtomekanismeista. Ohjausjärjestelmä eli ohjauslaatikko sisältää muun muassa I/O-rajapinnat ja Ethernet-liitännän, jotka soveltuvat monien erilaisten laitteiden kuten PLC:iden, turvalaitteiden ja robotityökalujen väliseen tiedonsiirtoon. Ohjausjärjestelmä sisältää myös erillisen liitännän käsivarren toimilaitteiden ohjaamiseen sekä virransyötön ja käsiohjaimen kytkennät. Käsiohjaimessa on kosketusnäytöllä varustettu graafinen käyttöliittymä PolyScope, jolla robotin ohjelmointi voidaan pääasiassa suorittaa. (Universal Robots 2014, 11, 29-45, 77.)

Robotin käsivarren eri osat ja niiden toispuoleiset kiertymissuunnat nähdään kuvassa 37. Kaikki nivelet pystyvät kiertymään molempiin suuntiin 360° , nopeimmillaan noin $180^\circ/\text{s}$ kulmanopeudella.

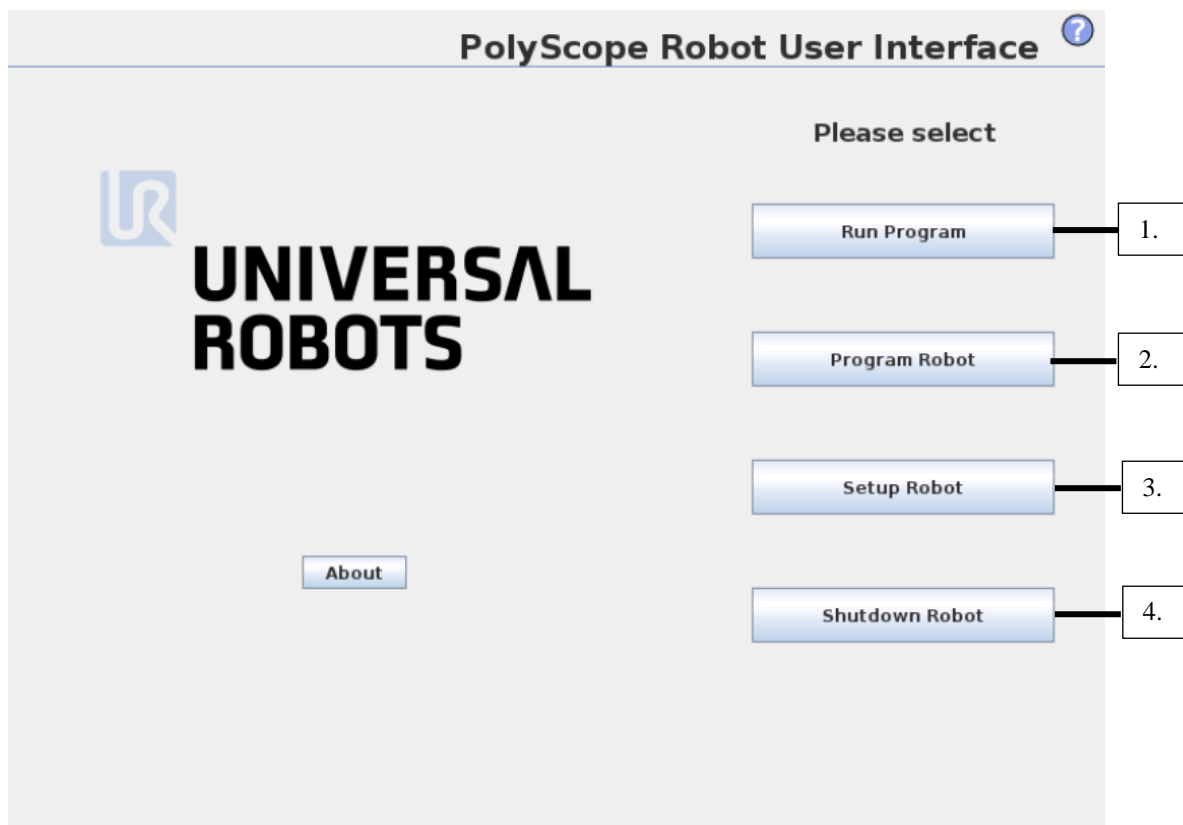


Kuva 37. UR5 robotin käsivarsi ja sen osat (Industrial Vision 2017) (muokattu).

Toimilaitteina käytetään kestmagneettisia AC-tahtimoottoreita ja niiden ohjaamiseen kaksoisenkooderi-järjestelmään perustuvia servonohjaimia, jotka on integroitu jokaiseen eri niveleeseen. Sisäisenä väylänä käytetään RS-485 kierrekaapelilla. (Universal Robots 2015.)

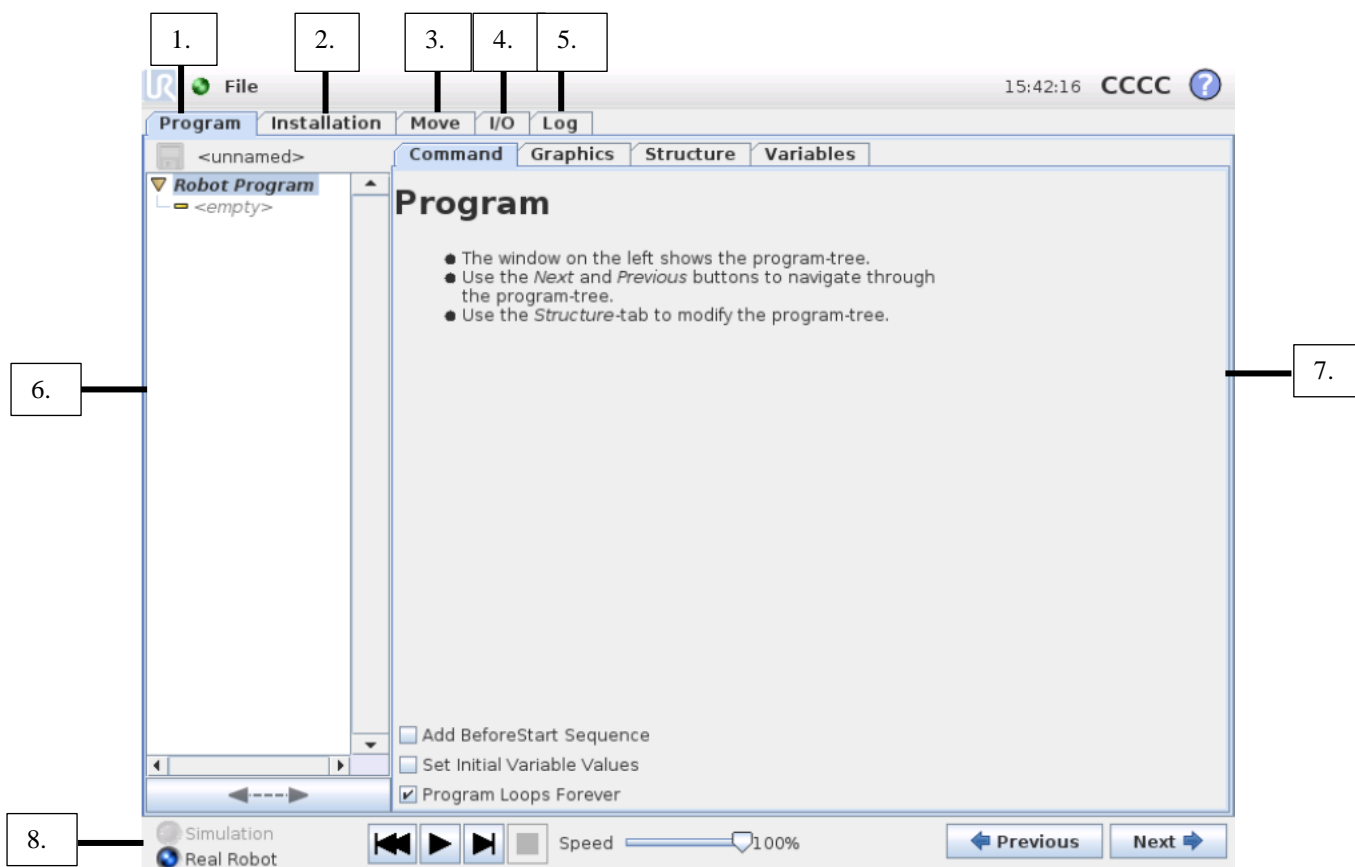
5.2 Käyttöliittymä ja ohjelmointi

UR5:en PolyScope-käyttöliittymän hallintaan käytetään ohjauslaatikkoon kiinnitettyä, kosketusnäyttöllistä käsiohjainta. Tekstipohjaisessa ohjelmoinnissa ohjaimeen voidaan tekstin kirjoittamisen helpottamiseksi kiinnittää USB-liitännällä myös erillinen näppäimistö ja osoitinlaite (hiiri). Kuvissa 38 ja 39 on käytön ja ohjelmoinnin kannalta tärkeitä graafisen käyttöliittymän (GUI eli Graphic User Interface) ikkunaruujuja. PolyScope-ikkunoiden valinnat toimivat hierarkkisen rakenteen mukaisesti. Tarkemmat ohjeet PolyScope:n käyttöön ja ohjelmointiin löytyvät UR5:n käyttäjän manuaalista sivuilta 75-173.



Kuva 38. PolyScope:n aloitusnäyttö (Universal Robots 2014, 82) (muokattu).

1. *Run Program*-valikosta valitaan ennestään tehtyjä robottiohjelmiä ja käynnistetään niiden lukemiset eli ohjelman ajot.
2. *Program Robot*-valikossa suoritetaan robotin GUI-pohjainen ohjelmointi. Valikosta voidaan vaihtaa tai luoda uusia ohjelmia.
3. *Setup Robot*-valikko tarjoaa robotin asennukseen liittyviä valintoja, kuten salasanan asettamisen, ohjelmiston päivittämisen ja kosketusnäytön kalibroimisen.
4. *Shutdown Robot*-painikkeesta sammutetaan robotin käsivarsi ja ohjauslaatikko.



Kuva 39. PolyScope-käyttöliittymän ohjelmointitilan yleisnäkymä (Universal Robots 2014, 114) (muokattu).

1. *Program*-välilehdellä suoritetaan robotin ohjelmointi ja näytetään ladatun ohjelman tietoja.
2. *Installation*-välilehti kattaa kaikki ne kohdat miten robotin käsivarsi ja ohjauslaatikko on asetettu toimimaan työskentely-ympäristössään. Tähän sisältyy esimerkiksi mekaaniset asennukset, sähköiset yhteydet ja robotin ohjelmaan liittyvät valinnat.
3. *Move*-välilehdeltä ohjataan suoraan robotin käsivarren asentoa (jogging). Välilehti sisältää robotin ja sen pisteiden graafisen 3D-mallin.
4. *I/O*-välilehdellä monitoroidaan ja asetetaan ohjauslaatikon lähes reaaliaikaisia (10 Hz) tulo- ja lähtösignaaleja.
5. *Log*-välilehdeltä nähdään robotin tilatietoja ja ilmoituksia.
6. *Program tree* eli ohjelmointipuukaavio näyttää ohjelman listana käskyjä.

7. Oikeanpuoleisessa tilaruudussa nähdään *Program*-välilehden valitun tietotyypin (tässä tapauksessa käskyjen) sen hetkiset tilat ja asetukset.
8. *Dashboard*-palkista voidaan käynnistää ja lopettaa ohjelman jatkuvatoiminen lukeminen tai edetä askel kerrallaan. Ohjelman lukunopeus suhteessa sen maksiminopeuteen on muutettavissa nopeussäätimen avulla. Vasemman reunan näppäimistä vaihdetaan ohjelman lukua simulaatiotilan ja todellisen työympäristön välillä.

UR5 käyttää kolmitasoista ohjelman lukujärjestelmää: graafisen käyttöliittymän tasoa (GUI Level), Script-tasoa (URScript Level) ja alemmantason ohjaintensäätö-tasoa (C++-kieleen pohjautuva C-API Level). GUI-tason ohjelmointi riittää moniin eri sovelluksiin. (Universal Robots 2015.)

5.3 Tiedonsiirto ja tekniset tiedot

UR5:lla on käytössä tiedonsiirtoon viisi erilaista rajapintaa: Ohjain I/O:t, työkalu I/O:t, Ethernet, pääliitännät ja robotti-liitännät. Kuvassa 40 näytetään UR5 robotin yleisimmät ominaisuudet ja tekniset tiedot.

6-axis robot arm with a working radius of 850 mm / 33.5 in

Weight:	18.4 kg / 40.6 lbs		
Payload:	5 kg / 11 lbs		
Reach:	850 mm / 33.5 in		
Joint ranges:	+/- 360°		
Speed:	All joints: 180°/s. Tool: Typical 1 m/s. / 39.4 in/s.		
Repeatability:	+/- 0.1 mm / +/- 0.0039 in (4 mils)		
Footprint:	Ø149 mm / 5.9 in		
Degrees of freedom:	6 rotating joints		
Control box size (WxHxD):	475 mm x 423 mm x 268 mm / 18.7 x 16.7 x 10.6 in		
I/O ports:		Controlbox	Tool conn.
	Digital in	16	2
	Digital out	16	2
	Analog in	2	2
	Analog out	2	-
I/O power supply:	24 V 2A in control box and 12 V/24 V 600 mA in tool		
Communication:	TCP/IP 100 Mbit: IEEE 802.3u, 100BASE-TX Ethernet socket & Modbus TCP		
Programming:	Polyscope graphical user interface on 12 inch touchscreen with mounting		
Noise:	Comparatively noiseless		
IP classification:	IP54		
Power consumption:	Approx. 200 watts using a typical program		
Collaboration operation:	15 Advanced Safety Functions Tested in accordance with: EN ISO 13849:2008 PL d EN ISO 10218-1:2011, Clause 5.4.3		
Materials:	Aluminum, PP plastic		
Temperature:	The robot can work in a temperature range of 0-50°C		
Power supply:	100-240 VAC, 50-60 Hz		
Cabling:	Cable between robot and control box (6 m / 236 in) Cable between touchscreen and control box (4.5 m / 177 in)		

Kuva 40. UR5 robotin tekniset tiedot (Universal Robots 2015b).

6 KAPPALEEN KÄSITTELYJÄRJESTELMÄN SUUNNITTELU JA TOTEUTUS

6.1 Laitteiston suunnittelu ja valinta

Projektin ensisijaisena tarkoituksena oli kehittää kevyiden kappaleiden tunnistus- ja poimintamenetelmä, kun kappaleet sijaitsevat laatikossa erilaisissa satunnaisissa asennoissa. Järjestelmä tehtiin niin, että sen ensiesittely oli Lounaisrannikon automaatiomessuilla heinäkuussa 2016. Projektin yhteydessä tutkittiin myös opetuskäyttöä varten suunniteltua 3D-konenäköliukuhihna-sovelluksen käyttöä ja laadittiin siihen soveltuva pikakäyttö-opas.

6.1.1 Tehtävät ja edellytykset

Järjestelmän ensimmäisenä tehtävänä on tunnistaa skannattavasta laatikosta ennalta määriteltävän kappaleen 3-ulotteinen paikka ja asento muiden samanlaisten kappaleiden joukosta. Paikalla tarkoitetaan jotakin kappaleen poiminnan kannalta kiinnostavaa XYZ-koordinaattipistettä ja asennolla kappaleen kiertymiä tai kallistumia joidenkin siihen soveltuvien akselien/tasojen suhteen. Toisena tehtävänä on kappaleen poimiminen ja palletoiminen sen uuteen sijoituspaikkaan, kuten toiseen laatikkoon. Tätä tunnistus-poiminta tehtäväkiertoa jatketaan niin kauan kunnes skannattava laatikko on tyhjennetty kappaleista.

Järjestelmä suunnitellaan toimivaksi kameran suhteen liikkumattomalla alustalla, mutta sen tulee myös soveltua liukuhihnan kanssa toimivaksi. Jos järjestelmä ei tunnista kappaleita, tulee asiasta joko antaa ilmoitus tai muuttaa kappaleiden asentoa ja paikkaa niiden uudelleen tunnistamista varten. Toteutuksen tulee kuitenkin olla suhteellisen toimintavarma, esimerkiksi niin että yhden kuvanoton jälkeen tehtäisiin n. >95 % todennäköisyydellä vähintään yksi kappaleen tunnistus-poiminta, jotta järjestelmä olisi sovellettavissa teolliseen käyttöön. Varsinaista järjestelmään sijoitetun pääoman tuottoa (Return Of Investment) ei kuitenkaan lasketa, koska projekti toteutetaan enimmäkseen opetuskäyttöä varten.

6.1.2 Kappaleiden ja olosuhteiden esittely

Pomittavan kappaleen valinnassa harkittiin ensin pesuharjojen muovisia varsia, mutta niiden poiminnan kannalta hankalista muodoista johtuen, päädyttiin lopulta Ultimaker 3D-tulostimella printattuihin kuusiomuovimuttereihin. Kuvassa 41 näkyvä sisäkierteinen nylonmuovi-mutteri on ulkohalkaisijaltaan n. 6 cm (reunasta reunaan), sisähalkaisijaltaan n. 3 cm ja korkeudeltaan n. 3,5 cm. Muttereita printattiin valkoisina, vaaleansinisinä ja -punaisina n. 20 kappaletta, joista yhdeksän valittiin testaamista varten niin että jokaista väriä olisi kolmen mutterin verran. Mutterin pinta on tekstipainaumien puolelta tasaista, sivuilta epätasaisempaa. Pinta heijastaa valoa enimmäkseen diffusivisesti, mutta mukana on myös hieman valonjakaumaa spekulaaarisesti herkissä kulmissa, erityisesti tekstittömällä pohjapuolella.



Kuva 41. 3D-printattu valkoinen mutteri.

Koska tavallisten pahvi- ja muovilaatikoiden pinnan heijastelut olisivat voineet aiheuttaa häiriötä kuvanmuodostuksessa ja laatikoiden korkeat, heikot reunat haasteita laatikon ja kappaleen käsittelyssä, päätettiin laatikko valmistaa itse. Kuvan 43 itse tehdyn laatikon, 30x40 cm kokoinen pohja on harmaantummaa mattapintaista PVC-muovia, reunat mustaa kiiltävöpintaista teknistä kumia korkeudeltaan n. 12 cm. Kantokahvat ovat alumiinireikänauhaa. Mutteri voi sijaita laatikossa kyljellään eli reuna vasten laatikon pohjaa, pohjallaan tai erilaisissa kallistuneissa asennoissa. Kallistuneessa asennossa mutteri nojaa joko toisiin muttereihin tai/ja laatikon reunoihin eli mutterit voivat

olla myös kosketuksissa toisiinsa. Pohja-asento on kaikkein stabiilein, muissa asennoissa mutteri voi kaatua tai liikahtaa pienistäkin kosketusvoimista ja värähtelyistä johtuen.

Järjestelmä testattiin ammattikorkeakoulun loisteputkilla valaistussa sisähuonetilassa, jossa auringonvalon tuloa oli vähennetty ikkunan tummennuksilla. Messuhallissa kattoikkunan kautta säteilyt luonnonvalo lisäsi ympäristön luminanssivaihtelua.

6.1.3 Kamera ja valaistus

Koska tarkoituksena oli esitellä 3D-konenäkötekniikkaan pohjautuvaa robottijärjestelmää tavallisessa sisätilavalauksessa, tulisi kameran ja sen käyttämän varsinaisen valaisun pystyä luomaan tarpeeksi kontrastia kappaleen piirteiden tunnistamiseksi. Tämän lisäksi haluttiin perehtyä älysensorin ja laserprofiili-skannerin käyttöön ja toimintaan. Näistä syistä johtuen kameraksi valittiin ammattikorkeakoulun käytöstä löytynyt Gocator 2380-3D-kamera. Älysensorin useita eri konenäön komponentteja yhdistävien ominaisuuksien ja nopeaa käyttöönottoa tukevan konfiguroinnin myötä säästettiin monilta perinteisen konenäköjärjestelmän aikaa vieviltä suunnitteluvaiheilta.

FOV eli Gocator:in tapauksessa kameran x-akselin tarkastelualueen pituus määriteltiin laatikon pohjan teon yhteydessä ja kameran korkeuden säädön aikana. Häiritsevän valaistuksen vähentämiseksi harkittiin myös laatikon varjostamista tukikehikolla, mutta kameran todettiin toimivan riittävän hyvin vapaassa valaistuksessakin. Kamera pystyy erottamaan heijastuneen laservalon, kunhan intensiteetti on säädetty kohdilleen ja kappale ei heijasta spekulaaarisesti liikaa.

6.1.4 Ohjelmisto

Kuvankäsittelyohjelmistoiksi olivat tarjolla Gocator:in oma ja MVTec:in Halcon-ohjelmisto. Halcon:in etuja ovat monipuoliset esimerkkiohjelmat ja käyttömahdollisuudet. Huonona puolena Halcon:issa ovat kalliit lisenssimaksut kokonaisvaltaisille käyttöoikeuksille. Lisäksi edistyneet ratkaisut voivat vaatia onnistuakseen laajaa asiantuntemusta

Gocator 2380:en omilla kuvan analysoinnin mittaustyökaluilla saadaan helposti mitattua kappaleiden tärkeimpiä ominaisuuksia, kuten pituuksia, sijainteja ja kaltevuuksia, sekä tehtyä joitakin piirteiden tunnistuksia, kuten ympyröitä, reunan muotoja ja pinnan-alueita. Gocatorilla on lisäksi hyvät valmiudet mittaustietojen lähettämiseksi muille laitteille. Gocator-kuvankäsittelyohjelma ei välttämättä sovellu edistynyttä kuvankäsittelyä vaativaan järjestelmään.

6.1.5 Ohjausjärjestelmä

Kappaleen käsittelemiseksi valittu UR5 käsivarsirobotti soveltuu hyvin pienimassaisien kappaleiden siirtämiseen ja herkän turvapysäytyksen ansiosta se toimii erinomaisesti myös ihmisläheisessä työskentelyssä. Myös robotin työalue (säteeltään 850 mm) on riittävä pieniin siirtotehtäviin, kuten mutterin siirtämiseksi laatikolta toiselle. Koska robotin haluttiin pystyvän tarttumaan mutteria halutulla puristusvoimakkuudella sen sisä- ja ulkopuolelta, sekä tunnistamaan työkaluun kohdistuvia ulkoisia voimia ja vääntöjä, valittiin tarttujaksi kuvan 42 voimasensorilla varustettu Robotiq-kaksisormitarttuja. Tarttuja ja voimasensori on suunniteltu toimimaan erityisesti UR-robottien kanssa. Molemmat asennettiin peräkkäin robotin työkalulaippaan ruuveilla, liitännät kiinnitettiin USB- ja RS-485-adapterien kautta ohjauslaatikkoon. Lisäksi robotin ohjelmaan asennettiin USB-muistitikun mukana tullut laiteajuri, joka tarjosi lisätoimintoja robotin työkalupisteiden opettamisen helpottamiseksi.



Kuva 42. Adaptiivinen kaksisorminen 85 mm Robotiq-tarttuja FT 300 voimasensorilla.

6.2 Järjestelmän toiminnan suunnittelu ja toteutus

Laitteistoihin tutustumisen jälkeen tehtiin eräitä järjestelmän toimintoihin liittyviä etukäteisarvioita. Onnistuneen mutterin poiminnan edellytyksenä arvioitiin olevan seuraavia huomioon otettavia pää-asioita:

- mutterin sijainnin ja asennon löytyminen
- riittävä tarkkuuden saavuttaminen mittauksissa ja kalibroinnissa
- onnistunut tiedonsiirto (“kättely”-signaalit ja XYZ-koordinaatit/-asennot)
- varma tarttuminen ja kiinnipitäminen

Onnistuneen kuvankäsittelyn edellytyksenä arvioitiin taas olevan:

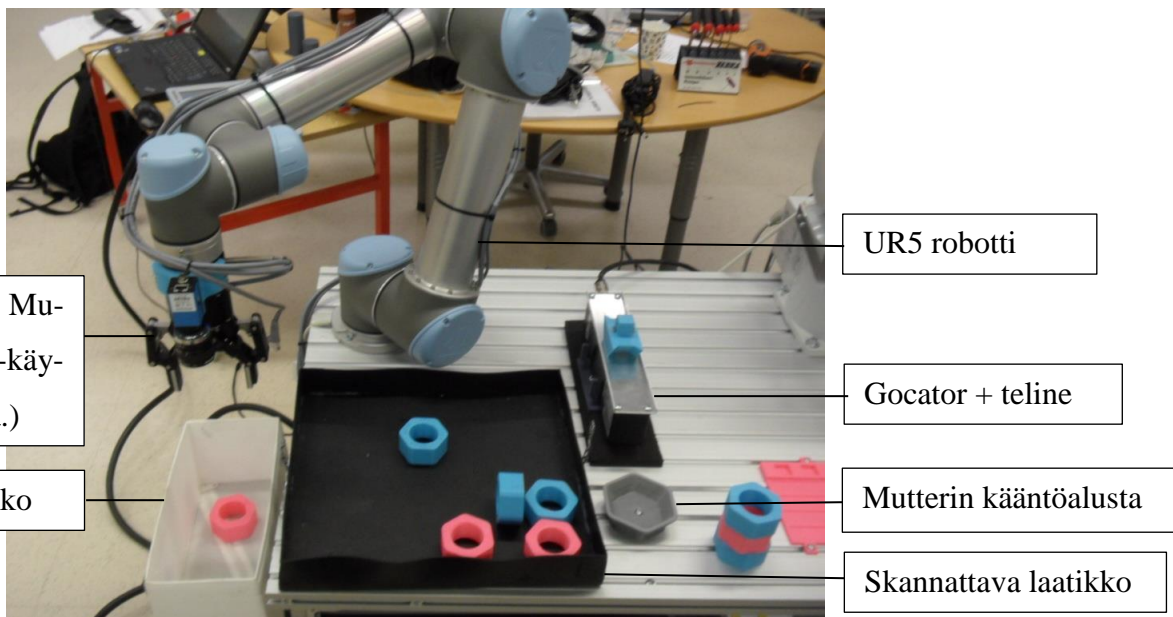
- kameran tasainen liike skannauksen aikana
- sopivat laserin säätöparametrit (valotustyyppi, -aika ja intensiteetti)
- hyvät filttorien säätöparametrit (Gap filling, median, smoothing ja decimation)
- toimivat tunnistusalgoritmit

Ohjelmaa lähdettiin aluksi tekemään MVTec:in Halcon-konenäköohjelmistolla, sen monipuolisen ja kattavan ohjelmistokirjaston vuoksi. Ohjelman teko jäi kuitenkin ajanpuutteen vuoksi kesken. Ohjelma olisi perustunut kappaleen valmiin CAD-mallin

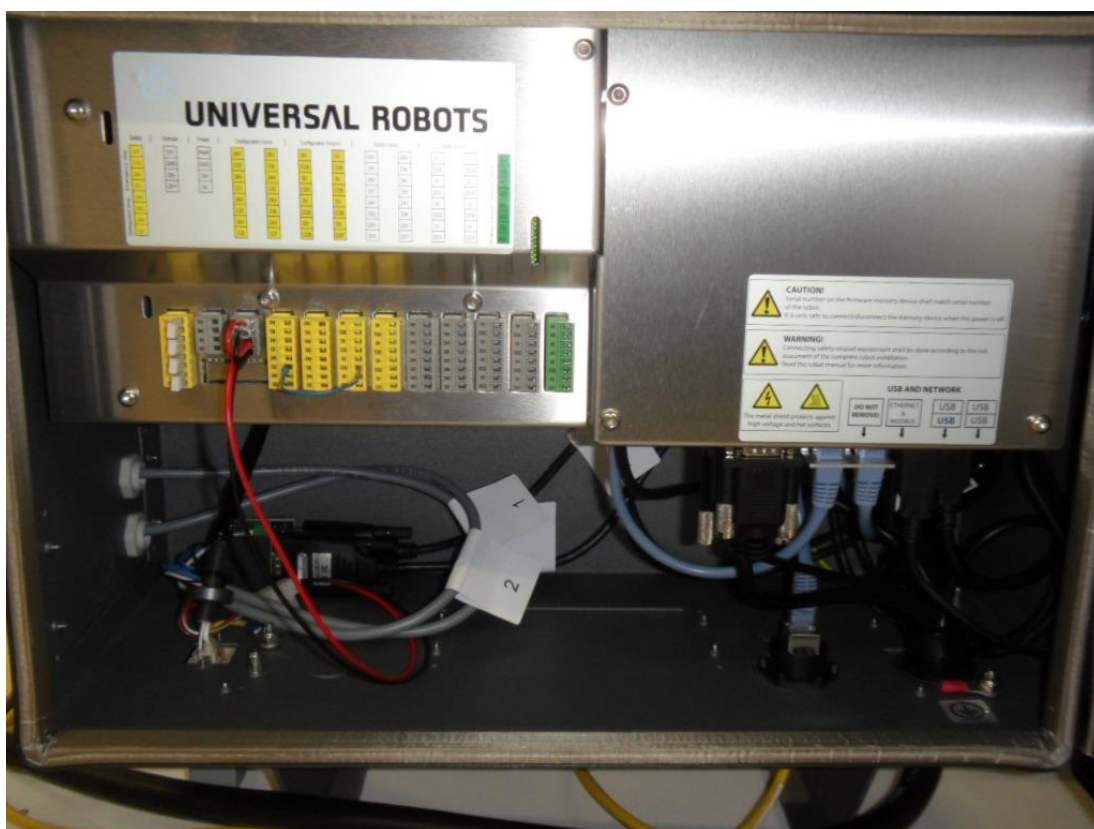
tai skannattujen 3D-mallien väliseen pintapohjaiseen 3D-kuviontunnistusmenetelmään. Näin järjestelmästä olisi tullut yleiskäyttöisempi kuin yksilökohtaisten piirteiden tunnistusalgoritmien käyttöön perustuvasta järjestelmästä, sillä vaihdettaessa poimittavaa esinettä olisi tarvinnut vaihtaa vain etsittävän kappaleen 3D-mallia (OM3-tiedostoa) ja sen poimintakohtaa.

Kappaleen tunnistukset ja mittaukset päätettiin siis toteuttaa Gocatorin omalla kuvankäsittelyohjelmistolla, joiden pohjalta PolyScope:lla ohjelmoitu UR5 suorittaisi poiminat ja sijoittamiset. UR5:n ja Gocatorin väliseksi tiedonsiirtoprotokollaksi valittiin Modbus-TCP, koska se havaittiin yksinkertaiseksi ja luotettavaksi kättely- ja mittaus-tietojen tiedonsiirtomenetelmäksi. Järjestelmässä Gocator toimi tiedonsiirron Serverinä ja UR5 Clientinä. 32-bittiset mittaustiedot lähetettiin kahtena 16-bittisenä datana.

Koska käytössä oli vain yksi käsivarsirobotti ja Gocatorin kiinteästä kiinnityksestä robotin tukivarteen olisi voinut seurata poiminta- ja mittausepävarmuuksia, valmistettiin Gogatorille oma alumiininen poimintajalusta ja sen tartuntakahva. Tartuntakahvan muotojen vuoksi Gocator asettuu aina samaan asentoon työkaluun nähden. Mutterille tehtiin lisäksi vielä kääntöalusta, joka asettaa pystyasentoisen mutterin palletointia varten sopivaan vaaka-asentoon. Kuvasta 43 nähdään järjestelmän komponentit. Kuvan 44 ohjauslaatikko sijoitettiin jalustakehikon alapuolelle. Tartuntakahva ja kääntöalustan CAD-mallit suunniteltiin Solidworks-ohjelmalla, joiden pohjalta esineet tulostettiin Ultimaker-3D-tulostimella.



Kuva 43. Järjestelmän komponentit.



Kuva 44. UR5:n ohjauslaatikko ja sen kytkennät.

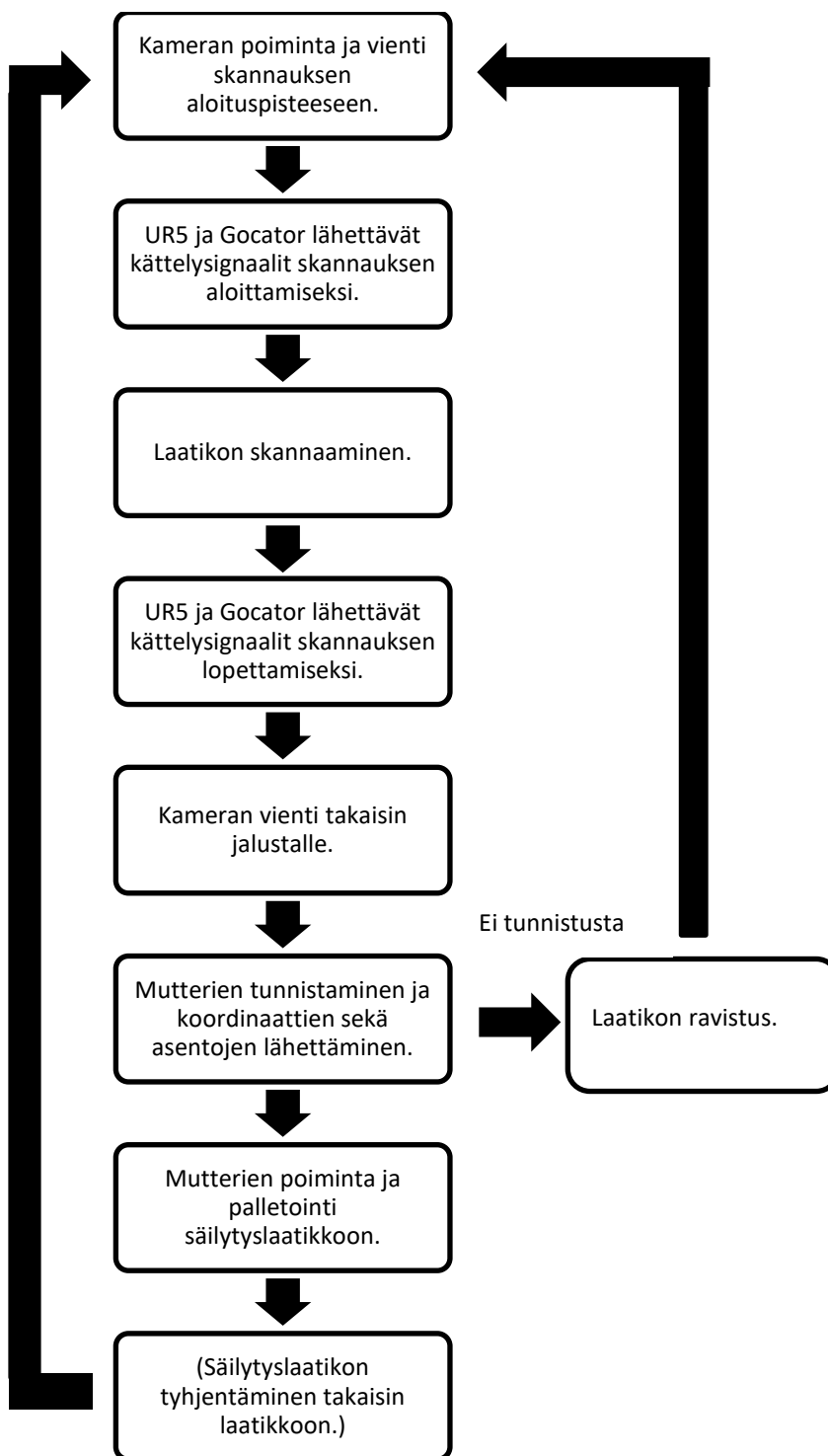
Gocatorin sisäinen suuntaus tehtiin skannaamalla halkaisijaltaan 100 mm kalibrointikielkkoa alignment-toiminnon aikana. Tämän jälkeen tehtiin aktiivisen mittausalueen rajaaminen (Active area), sekä laser- että kuvankäsittelyparametrien säädöt (*Exposure* → *Dynamic*, *Spacing*, *Material* → *Diffuse* ja *Filters*). Koska LMI Tec:illa ei ollut tarjota valmista menetelmää Gocatorin ulkoisen suuntauksen suorittamiseksi, toteutettiin se viemällä UR5:n maailmakoordinaatisto samaan paikkaan ja asentoon kuin sisäisesti suunnattu kamerakoordinaatisto (*installation*-välilehti → *feature* → *plane*). Tätä varten mitattiin mutterin paikka ja asento kolmessa eri kohtaa, sekä kameralla että robotin työkalulla. Laskemalla näiden koordinaattien ja kiertymien eri koordinaatistojen erotusten keskiarvot voitiin homogeenisten siirrosmatrisien avulla laskea työkalun kamerakoordinaatiston arvot UR5:n peruskoordinaatistossa. Katso Liitteestä 2 kaavan johto ja Python-ohjelma, joka laskee koordinaattimuunnokset. Tasomaisen maailmakoordinaatiston luomiseksi tarvitaan kolme UR5:lle opetettavaa alipistettä. Ensimmäinen alipisteen (*subpoint-feature*) sijainti lasketaan kamerakoordinaatiston origon peruskoordinaattimuunnoksesta, toinen jonkin kamerakoordinaatiston y-akselilla sijaitsevan pisteen ja kolmas jonkin x-akselilla sijaitsevan pisteen koordinaattimuunnoksesta. Toinen tapa ulkoisen suuntauksen suorittamiseksi olisi selvittää silmämääräisesti arvioimalla ja mittavälineellä mittaamalla koordinaatistojen erotukset ja kiertymät, mitä testivaiheessa myös tehtiin.

Säätöjen ja testien yhteydessä huomattiin kameran valkoisten pystyasentoisten muttereiden havaitsemisen olevan heikkoa. Tämän takia valkoiset mutterit korvattiin sopivalla määrällä vaaleanpunaisia ja -sinisiä muttereita. Valkoisia muttereita ei ehkä tunnusteta niin helposti, koska valkoiset mutterit olisi valmistettu heijastavammasta muovimateriaalista kuin muut mutterit tai laserin dynaaminen valotus ei osaa säätää oikealla tavalla mustaa taustaa ja valkoista kappaletta kohden. Muuten valkoisten esineiden pitäisi näkyä laserskannauksissa hyvin.

UR5:n PolyScope-ohjelma (katso Liite 1) koostuu kolmesta osasta: alustuksesta (BeforeStart), robotin pääohjelmasta (Robot Program) ja voima-anturin säieohjelmasta (Thread_1). Alustuksessa tehdään muuttujien määrittely, tarttujan aliohjelmien ja robotin Script:ien kutsuminen/lukeminen sekä robotin kotiasemaan ajo. Pääohjelmassa suoritetaan jatkuvatoimisesti Kaavion 1 toimintasekvenssiä. Mutterin poiminnan ai-

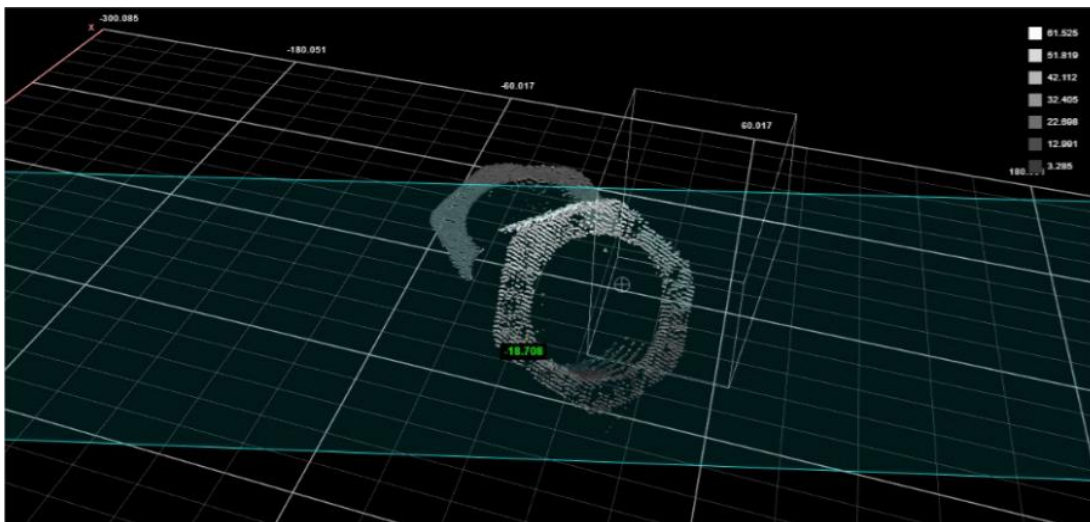
kana säieohjelma tarkkailee tarttujaan kohdistuvia kohtisuoria voimia. Asetetun voiman kynnyksarvon ylittyessä, säieohjelma keskeyttää poiminnan ja ohjaa robotin tekemään uusintaskannauksen, ilman että UR5:n sisäinen turvapysäytys kytkeytyisi tarpeettomasti päälle. Vakiopisteet ja suhteelliset liikkeet ohjelmoitiin opettamalla. Palletoimiseen käytettiin UR5:n "Pallet"-liikekäsä.

Kaavio 1. Järjestelmän toimintasekvenssikaavio.



Kun mutteri on vaakatasossa tai vinossa asennossa poiminta suoritetaan viemällä robotin työkalu mutterin kierteiselle sisäpuolelle ja avaamalla työkalu. Mutterin reiän tunnistamiseen käytettiin Gocator:in Surface Hole-mittaustyökalua. Tähän ankkuroitiin Surface Plane- ja Surface Bounding Box-työkalut, pinnan kallistumien tunnistusta, (X- ja Y-akselien suhteen) ja globaalien eli maailmakoordinaatiston mukaisten XY-

koordinaattien löytämistä varten. Kuvassa 45 nähdään vinoasentoinen mutteri ja sen pinnan kaltevuutta mittaava tunnistustyökalu.



Kuva 45. Gocator-käyttöliittymän graafinen esitys vinosta mutterista.

Mutterin ollessa pystyasennossa poiminta toteutetaan asettamalla avattu tarttuja mutterin reunojen kanssa yhdensuuntaiseksi ja tarttumalla mutteria sen sivuista kiinni. Mutteri viedään sen jälkeen kääntöalustalle (katso kuva 43), josta se poimitaan sisäpuolelta tarttumalla. Pystyasennossa mutterin Z-akselin kiertymän tunnistamiseen käytettiin Ellipse-työkalua, johon ankkuroitiin Surface Bounding Box-työkalu. Korkeuden oletettiin pysyvän vakiona, joten UR5 ohjelmassa asetetaan Z-koordinaatin offset-arvoksi noin puolet mutterin pystyasennon korkeudesta.

Gocator:in Measurement Script -ohjelmassa määritetään mittaustyökalujen Id-tunnuksen perusteella muuttujiksi mutterin perusasento (vaaka/vino tai pystyasento) ja koordinaattiasentoarvot. Muuttujille annetaan *Output*-välilehdellä omat Modbus-lähtökisteriosoitteet tiedonsiirtoa varten. Kun muttereita tunnistetaan useita samalla skannauksella, mittausarvot bufferoidaan eli tallennetaan vuoroittain puskurimuistiin. Koordinaattien lukemiseksi UR5-ohjelma lukee (kohdassa Buffer_count) Advance register-tulosta, jonka seurauksena Gocator lähettää puskurijonossa seuraavat koordinaatti- ja asentotiedot UR5:lle luettavaksi.

Koska Gocatorin kuvankäsittelyohjelma ei tarjoa valmista, toimivaa kolmiulotteisen kappaleen tunnistusmenetelmää, se ei pysty erottamaan vinosti toisiaan koskettavia kappaleita toisistaan. Tämän takia Script-ohjelma laskee myös kappaleiden pinta-alaa ja tunnistaa toisiaan koskettavat kappaleet yksittäisistä kappaleista. Tunnistuksen jälkeen ohjelma kynnystää yksittäisten poimintakohtien korkeuksia ja kallistumia, joista poimintajärjestys päällimmäisestä alimpaan kappaleeseen voidaan päätellä.

Gocator:in Output-tietojen määrittelyjen jälkeen UR5:lle asetettiin käytettävien kätteleysignaalien ja vastaanotettavien mittaustietojen Modbus-I/O-rekisteriosoitteet (*installation*-välilehti → *Modbus CLIENT IO Setup*). Kahteen 16-bittiseen Modbus-osoitteeseen jaetut 32-bittiset mittaustiedot muunnetaan liitteen 1 *Hae_mutterit* jälkeisissä muuttujien määrittelyissä takaisin kymmenlukujärjestelmän XYZ-millimetrikoordinaateiksi ja akselien kiertymä-arvoiksi. Jos hyväksytyjä mittauservoja ei löydy, suoritetaan laatikon ravistus ja uusintaskannaus, muussa tapauksessa jatketaan poimintojen suorittamista.

Messunäyttelyn jälkeen ohjelmaan tehtiin muutamia robotin syklinopeutta ja vuorovaikutteisuutta lisääviä muutoksia, kuten nopeus- ja voimantarkkailuparametrien säätöjä ja toruvan eleen toteuttamista käyttäjän siirtäessä skannattua kappaletta ennen kuin robotti ehtii poimimaan sitä.

7 YHTEENVETO

Projektin tavoitteet ja edellytykset tulivat suurimmalta osin täytetyiksi. Projekti eteneminen komponenttien valinnoista ja niihin tutustumisesta kohti valmiin järjestelmän esittelyä automaatiomessuilla määräaikaan mennessä, tarjosi useiden haasteiden ratkaisemista ja testauksia. Käytettävien menetelmien valinnassa annettiin varsin vapaat kädet, jonka ansiosta luovienkin ratkaisujen löytäminen, kuten 3D-tulosteisten työkalujen käyttäminen, oli mahdollista.

Järjestelmä pystyy poimimaan kappaleen useista eri asennoista, hyödyntäen tunnistuksessa ja poiminnassa kappaleelle sen ominaisia piirteitä. Poimittavaa kappaletta vaihdettaessa joudutaan etsimään uudet tunnistusalgoritmit ja poimintakohdat. Tämän välttämiseksi olisi vaadittu edistyneempiä kuvankäsittelyalgoritmeja, kuin mitä Gocator:in sen hetkisen version käyttöliittymästä löydettiin. Toisaalta näin tehtiin samalla myös perehtyminen kameran omiin käyttömahdollisuuksiin. Tiedonsiirtäminen TCP Modbus -protokollan avulla oli yksinkertainen ja toimiva ratkaisu robotin sekvenssi-mäisiin toimintoihin, joita teollisuusrobotit yleensä suorittavatkin.

Järjestelmän toimintavarmuutta ja nopeutta tulisi kuitenkin vielä säätää paremmaksi ennen sen teollisuuteen vientiä. Lisäksi olisi lisättävä toiminnot laatikon reunojen välttämiseksi, sekä tehtävä valmiudet liukuhihnojen, robotin ja muiden soluyksiköiden välisten yhteyksien kalibroinneille ja konfiguraatioille. Kappaleen kolmiulotteisen asennon tunnistamiseen olisikin ehkä parempi ratkaisu käyttää laser-skannerin sijaan jotain tasomaiseen valaisukuvioon perustuvaa kuvaustekniikkaa, kuten Fringe-kuvausta. Näin välttyttäisiin muun muassa suhteellisen liikkeen ja laservalon käytön tuomilta ongelmilta ja vaaroilta, sekä vähennettäisiin kuvanoton kestoa. 3D-tekniikan nopean kehittymisen myötä useat konenäköjärjestelmien valmistajat, kuten Canon ja SICK, tarjoavatkin yrityksille jo valmiita kolmiulotteisia tunnistus- ja poimintajärjestelmiä. Tekijälle projekti antoi kuitenkin erinomaisen perehdytyksen robotiikan ja konenäön teorian lisäksi myös käytännön toimenpiteisiin ja moni asia selveni usein vasta yrityksen ja erehdyksen kautta.

LÄHTEET

Aalto, H., Heilala, J., Hirvelä, T., Kuivanen, R., Laitinen, M., Lehtinen, H., Lempiäinen, J., Lylynoja, A., Renfors, J., Selin, K., Siintoharju, T., Temmes, J., Tuovila, T., Veikkolainen, M., Vihinen, J. & Virtanen, A. 1999. Robotiikka. Vantaa: Talentum Oyj/ Metallitekniikka.

Aaltomuoto. n.d. Viitattu 17.10.2016. <https://aaltomuoto.wordpress.com/valo/nayttamovalaisun-perusteet/valo-ja-varit/>

Adaptive Vision. 2017. Viitattu 31.3.2017. http://docs.adaptive-vision.com/4.7/studio/machine_vision_guide

All on robots. 2016. Viitattu 12.10.2016. <http://www.allonrobots.com/spherical-robots.html>

Alrad. n.d. Viitattu 28.2.2017. <http://www.alrad.co.uk/datasheets/Imaging/Laser%20Light%20Structures%20in%20Machine%20Vision%20Applications.pdf>

Beleboni, M. 2014. Viitattu 10.03.2017. <http://mms.ecs.soton.ac.uk/2014/papers/2.pdf>

Birchfield, S. 2016. Image Processing and analysis. Gengage Learning. Viitattu 17.5.2017. <https://books.google.fi>

Chocholak, F. 2015. Viitattu 12.10.2016 <http://www.kirp.chtf.stuba.sk/~kvasnica/blog/scara-robots/>

Choi, O., Hansard, M., Horaud, R. & Lee, S. 2012. Viitattu 15.3.2017. <https://hal.inria.fr/hal-00725654/PDF/TOF.pdf>

Cimcorp. 2016. Viitattu 12.10.2016. <http://cimcorp.com/media/news/press-release/cimcorp-unveils-new-dream-factory-solution>

Clément, M., Evangelidis, G., Hansard, M. & Horaud, R. 2016. Viitattu 19.3.2017. <http://www.eecs.qmul.ac.uk/~miles/h/Horaud-TOF-HAL.pdf>

Daryl, M. 2012. Viitattu 28.1.2017. http://www.rauscher.de/downloads/public/datenblaetter/Machine-Vision-Lighting_Practical-Guide_2012.pdf

Davies, E. 2005. Machine Vision Theory, algorithms, practicalities. 3. painos. San Francisco: Morgan Kauffman.

Direct Industry. n.d. Viitattu 29.3.2017. <http://www.directindustry.com/prod/abb-robotics/product-30265-169123.html>

Egemin Group NV. 2016. Viitattu 5.5.2017. http://www.egemin-automation.com/en/automation/material-handling-automation_ha-solutions/agv-systems

Epic Systems, Inc. n.d. Viitattu 20.10.2016. <https://www.epicsysinc.com/blog/machine-vision-history>

Favaro, P. 2002. Viitattu 22.3.2017. http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/FAVARO1/dfdutorial.html

Fotorico. n.d. Viitattu 26.3.2017. http://fotoportico.com/mp/index.maml?t_LectureID=143

Frankling, C. 2008. Viitattu 24.4.2017. <http://computer.howstuffworks.com/3dgraphics.htm>

Hornberg, A. 2006. Handbook of Machine Vision. Darmstadt: GmbH.

Hägele, M., Nilsson, K. & Pires, J. K. 2007. Viitattu 10.08.2017. https://www.researchgate.net/publication/226964084_Industrial_Robotics

Industrial Vision. Viitattu 5.1.2017. <http://www.industrialvision.co.uk/applications/robot-control>

Inkinen, P., Manninen, R. & Tuohi, J. 2011. Momentti insinöörifysiikka 2. 2.-4. painos. Keuruu: Otava.

Inspro9. 2016. Viitattu 3.4.2017. <https://inspro9.wordpress.com/2016/12/14/correcting-for-distortion/>

International Federation of Robotics. 2017. Viitattu 24.3.2017. <https://ifr.org/robot-history>

ISO 10218-1:2011. Robots and robotic devices. Safety requirements for industrial robots. Part 1: Robots. 2011. Viitattu 22.9.2016. <https://www.iso.org>

ISO 8373:2012. Robots and robotic devices. Vocabulary. 2012. Viitattu 22.9.2016. <https://www.iso.org>

Karpinsky, N. & Chang, S. 2010. Viitattu 20.3.2017. <http://hanquier.m.free.fr/Worcester/references/Others%20papers/fringe%20projection/High-resolution,%20real-time%203D%20imaging%20with%20fringe%20analysis.pdf>

Karpinsky, N. & Chang, S. 2010b. Viitattu 21.3.2017. <http://opticalengineering.spiedigitallibrary.org/article.aspx?articleid=1096370>

Kennedy, W. n.d. Viitattu 16.5.2017. http://www.disensors.com/downloads/products/The%20Basics%20of%20Triangulation%20Sensors_1161.pdf

Korem, Y. n.d. Viitattu 10.7.2017. http://www-personal.umich.edu/~ykoren/uploads/Robotics_For_Engineers_new.pdf

Korpela, J. n.d. Viitattu 30.3.2017. <https://www.cs.tut.fi/~jkorpela/foviord/sanasto.html>

Leino, M. 2015. Viitattu 17.10.2016. https://moodle2.samk.fi/pluginfile.php/162926/mod_resource/content/5/Johdanto%20konen%C3%A4k%C3%B6%C3%B6n.pdf

Liu, J.W., Liang, J., Liang, X.H. & Tang, Z.Z. 2010. Viitattu 5.6.2017. <http://opticalengineering.spiedigitallibrary.org/article.aspx?articleid=1096281>

Live Science. 2010. Viitattu 22.4.2017. <http://www.livescience.com/32580-why-do-we-see-in-3-d.html>

LMI Technologies. 2015. Viitattu 3.11.2016. http://lmi3d.com/manuals/gocator/gocator-4.1/pdf/15159-4.1.4.12_MANUAL_User_Gocator-2300-2880-Series.pdf

MacCormick, J. 2011. Viitattu 10.3.2017. <https://users.dickinson.edu/~jmac/selected-talks/kinect.pdf>

Mahola. 2013. Viitattu 20.4.2017. <https://mahola.vuodatus.net/sivut/kuvituksia>

Malm, T. ym. 2008. Vuorovaikutteisen robotiikan turvallisuus. Helsinki: Hakapaino Oy.

Mecademic Inc. 2017. Viitattu 5.3.2017. <http://www.mecademic.com/What-is-a-parallel-robot.html>

Microscan. 2017. Viitattu 27.1.2017. <http://www.microscan.com/en-us/products/ner-lite-machine-vision-lighting/edge-to-edge-backlights>

MicroScan. 2017b. Viitattu 31.3.2017. http://files.microscan.com/helpfiles/visions-capetools_help_file/visionscape_tools_reference-04-1.html

Miniweb. 2016. Viitattu 20.10.2016. http://miniweb.lpt.fi/automaatio/opetus/luennot/pdf_tiedostot/Robotiikka_yleinen.pdf

Pena, C. 2010. Viitattu 26.2.2017. <https://www.dmcinfo.com/latest-thinking/blog/id/185/selecting-the-right-machine-vision-software-and-platform>

Phoenix Imagine. n.d. Viitattu 31.3.2017. <http://phoenix-machine-vision.com/>

Robotics Automation Systems. 2016. Viitattu 25.3.2017. <http://www.roboticautomationssystem.com/6-axis-robots.html>

Roboticsbible. 2016. Viitattu 12.10.2016. <http://www.roboticsbible.com/robot-drive-systems.html>

Robotpark. 2016. Viitattu 12.10.2016. <http://www.robotpark.com/academy/all-types-of-robots/stationary-robots/cylindrical-robots/>

RobotWorx. 2016. Viitattu 12.10.2016. <https://www.robots.com/articles/viewing/articulated-robots-more-axes-equal-more-possibilities>

RobotWorx. 2016b. Viitattu 12.10.2016. <https://www.robots.com/faq/show/what-are-gantry-robots>

Shi, Y. & Real, F. Chapter 2. Smart Cameras: Fundamentals and Classification. 2010. Viitattu 26.2.2017. www.springer.com

Soini, A. n.d. Viitattu 12.5.2017. <http://automaatioseura.planeetta.com/index/tiedostot/Konenako.pdf>

Steger, C., Ulrich, M. & Wiedemann, C. 2007. Machine Vision Algorithms and Applications. Darmstadt: GmbH.

Strujik, B. 2011. Viitattu 12.10.2016. <http://www.zmne.hu/aarms/docs/Volume10/Issue1/pdf/15.pdf>

STUK. 2015. Viitattu 5.3.2017. <http://www.stuk.fi/aiheet/laserit/laserluokat>

Szeliski, R. 2010. Viitattu 20.2.2017. http://szeliski.org/Book/drafts/Szeliski-Book_20100903_draft.pdf

Tarin, M. 2013. Viitattu 26.2.2017. <http://www.qualitymag.com/articles/91010--d-imagining-enters-the-machine-vision-world>

Teledyne Dalsa. 2014. Viitattu 17.10.2016. http://www.automation.com/pdf_articles/dalsa/210414_wp_BasicsMVision.pdf

Texas Instruments. 2016. Viitattu 26.2.2017. <http://www.ti.com/lit/ug/tiduc48a/tiduc48a.pdf>

TKK Automation Technology Laboratory. n.d. Viitattu 31.3.2017. http://automation.tkk.fi/attach/AS-0-2230/lab3c_teorja.pdf

Universal Robots. 2014. User manual UR5/CB3 Version 3.0.

Universal Robots. 2015. Viitattu 3.3.2017. <https://www.universal-robots.com/how-tos-and-faqs/faq/>

Universal Robots. UR5 Technical specification. Item no. 110105. 2015b.

Wallen, J. 2008. Viitattu 24.3.2017. <http://liu.diva-portal.org/smash/get/diva2:316930/FULLTEXT01.pdf>

Wheelomania. 09.2011. Viitattu 5.5.2017. <http://www.wheelomania.com/2011/09/mobile-robots-to-replace-labour-at.html>

Vision-systems. 2013. Viitattu 20.10.2016. <http://www.vision-systems.com/articles/print/volume-18/issue-8/features/keystones-of-machine-vision-systems-design.html>

Yusuf, B. 2016. Viitattu 28.2.2017. <https://all3dp.com/3d-scanner-scanning-101/>

UR5 PolyScope ohjelma-script muodossa

Program

Init Variables

BeforeStart

MoveJ

Home

rq_obj_detect:=0

Script: rq_before_start.script

Script: rq_script.script

rq_speed:=255

rq_force:=255

Pala_skannauks:=0

Gocator_kyydiss:=0

poiminta_epaonn:=0

ProtectiveS_est:=0

Mutteri_har_alu:=0

Poimintaa_yrit:=0

Call SubP_rq_activat

Call SubP_rq_set_for

Call SubP_rq_set_spe

Call SubP_rq_close_a

Wait: 0.5

Call SubP_rq_open_an

Robot Program

'Varsinainen ohjelma'

If Ei_palikkaa² 1

MoveJ

Home

Pala_skannauks:=0

Popup

If poiminta_epaonn² 2

"Ravistus" jos poiminta epäonnistuu samalla kappaleella kahdesti.'

MoveJ

Home

Call SubP_rq_open_an

MoveJ

Ravistus_yla

Ravistus_ala

Call SubP_rq_close

Wait: 1.0

Waypoint_9

Ravistus_ala

Call SubP_rq_open_an

Ravistus_yla

Ravistus_v_yla

Ravistus_v_ala

Call SubP_rq_close_a

Wait: 1.0

Waypoint_10

Wait: 0.5

Ravistus_v_ala

Call SubP_rq_open_an

Ravistus_v_yla

poiminta_epaonn:=0

Call Kuvaa_Gocator

Wait: 0.01

If Palaa_skannauks[?]=0

Call Hae_mutterit

Wait: 0.01

Thread_1

'Jos tyokaluun kohdistuva voima ylittää sallitun arvo, robotti palaa kotipisteeseen ja aloittaa ohjelman alusta -> skannauksesta'

'Huom! Moven nopeuksia voi joutua saatamaan.'

Voima:=force()

Wait: 0.01

If Voima>80 and Gocator_kyydiss=0 and rq_obj_detect=0 and ProtectiveS_est=1

'Popup'

'Nyt ohjelma suorittaa skannauksen, voidaan ehka tehdä myös ravistelu.'

Wait: 0.01

MoveL

Waypoint_11

Wait: 0.01

poiminta_epaonn:=poiminta_epaonn+1

Wait: 0.01

ProtectiveS_est:=0

Wait: 0.01

Palaa_skannauks:=1

Wait: 0.01

Kuvaa_Gocator

Set CO[2]= True

Gocator_kyydiss:=1

MoveJ

'Home'

Home

Call SubP_rq_open

Over_scanner

MoveL

Scanner_down

Call SubP_rq_close_a

Wait: 0.5

Way_over_scanne

MoveJ

Scan_start

'Tehdaan kattelysignaalit Gocatorin skannauksen aloittamiseksi.'

Set gocator_start=1

Wait gocator_ready=1

MoveL

Scan_end_1

'Lopetetaan skannaus'

```

Set gocator_start=0
Wait gocator_ready≠1
MoveJ
    Way_over_scanne
MoveL
    Scanner_down
    rq_speed:=0
    Call SubP_rq_set_spe
    Call SubP_rq_open_an
    rq_speed:=255
    Call SubP_rq_set_spe
    Over_scanner_ou
    Palaa_skannauks:=0
Hae_mutterit
Set CO[2]= False
Gocator_kyydiss:=0
MoveJ
    X_paikka:=-1
    X_paikka_pysty:=-1
    Counter:=0
    Buffer_count:=Buffer_count_1
    'Jos Gocator ei tunnista kappaletta, X_koordinaatit tulevat negatiivisina.'
    Loop X_paikka<0 and X_paikka_pysty<0 and (Buffer_count≠Counter)
        var_1:=Advance_gocator
        Counter:=Counter+1
        Wait: 0.5
        X_paikka:=(X_a*65536+X_b)/1000000
        X_paikka_pysty:=(X_a_pystykpl*65536+X_b_pystykpl)/1000000
        If X_paikka>-2148 and X_paikka<-2146
            Ei_palikkaa:=1
        If X_paikka_pysty>-2148 and X_paikka_pysty<-2146
            Ei_palikkaa:=1
        If X_paikka=0 and X_paikka_pysty=0

```

```

    Ei_palikka:=1
Else
    Ei_palikka:=0
If X_paikka<0 and X_paikka_pysty<0
    'Nostetaan alustaa jos ei tunnisteta kappaletta'
    Call SubP_rq_open_an
    Ravistus_yla
    Ravistus_ala
    Call SubP_rq_close
    Wait: 1.0
    Waypoint_9
    Ravistus_ala
    Call SubP_rq_open_an
    Ravistus_yla
    Ravistus_v_yla
    Ravistus_v_ala
    Call SubP_rq_close_a
    Waypoint_10
    Wait: 0.3
    Ravistus_v_ala
    Call SubP_rq_open_an
    poiminta_epaonn:=0
Loop X_paikka>0 or X_paikka_pysty>0
    'Vaaka- ja vinokappaleen muuttujat'
    X_paikka:=(X_a*65536+X_b)/1000000
    Y_paikka:=(Y_a*65536+Y_b)/1000000
    Z_paikka:=(Z_a*65535+Z_b)/1000000
    XYZ:=X_paikka+Y_paikka+Z_paikka
    X_kulma:=((X_Angle_a*65536+X_Angle_b)/1000)-180
    Y_kulma:=((Y_Angle_a*65536+Y_Angle_b)/1000)-180
    'Pysty kappaleen muuttujat'
    X_paikka_pysty:=(X_a_pystykpl*65536+X_b_pystykpl)/1000000
    Y_paikka_pysty:=(Y_a_pystykappal*65536+Y_b_pystykpl)/1000000

```



```

'Z_paikka_pysty:=(Z_a_pysty*65536+Z_b_pysty)/1000000'
Z_kulma:=((Z_angle_a*65536+Z_angle_b)/1000)-180
XYZ_pystykp:=X_paikka_pysty+Y_paikka_pysty
'Valitaan kappaleesta vain vinokappale'
'Loop Do_Not_Pick=1 and (X_kulma<15) and (X_kulma>-15) and
(Y_kulma<15) and (Y_kulma>-15)'
X_kulma_rad:=d2r(X_kulma)
Y_kulma_rad:=d2r(Y_kulma)
'Huom! Z kulma tarkoitettu vain pystyssä olevalle kappaleelle'
Z_kulma_rad:=d2r(Z_kulma)
'Suoritetaan poiminta vaaka- tai vinokappaleelle.'
If X_paikka>0
  MoveJ
  Call SubP_rq_close
  Poiminta_apupis
  Lahestymispiste:=p[X_paikka-0.5765,((Y_paikka-0.1417)*-
1),Z_paikka+0.1,0,3.14,0]
  Poimintapiste:=p[(X_paikka-0.5765),((Y_paikka-0.1417)*-
1),Z_paikka+0.05,0,3.14,0]
  If (X_kulma<5) and (X_kulma>-5) and (Y_kulma<5) and (Y_kulma>-5)
    X_kulma_rad:=0
    Y_kulma_rad:=0
  'Lahestymispiste'
  Poimintapiste
  poiminta_pose:=get_actual_tcp_pose()
  X_kiertyma:=p[0.0,0.0,0.0,X_kulma_rad,0,0]
  Y_kiertyma:=p[0.0,0.0,0.0,0,-1*Y_kulma_rad,0]
  X_kaanto:=pose_trans(poiminta_pose,X_kiertyma)
  X_kaanto
  poiminta_pose2:=get_actual_tcp_pose()
  Y_kaanto:=pose_trans(poiminta_pose2,Y_kiertyma)
  Y_kaanto
  ProtectiveS_est:=1

```

```

Waypoint_2
Poiminta_pose3:=get_actual_tcp_pose()
Z_muutos_lahest:=p[0.0,0.0,0.025,0.0,0.0,0.0]
Sormet_sisaan:=pose_trans(Poiminta_pose3,Z_muutos_lahest)
Sormet_sisaan
Call SubP_rq_open_an
ProtectiveS_est:=0
Z_muutos_pois:=p[0.0,0.0,-0.025,0,0,0]
poiminta_pose4:=get_actual_tcp_pose()
Vie_mutteria_hi:=pose_trans(poiminta_pose4,Z_muutos_pois)
Lahestymispiste
Poimintaa_yrit:=1
ElseIf X_paikka_pysty>0 and Mutteri_har_alu $\neq$ 0
'Suoritetaan poiminta pystykappaleelle. Huom! Vaikeuksia tunnistaa valkoinen
mutteri.'
MoveJ
Call SubP_rq_open
Poiminta_apupis
MoveL
Poimintapiste_p:=p[X_paikka_pysty-0.5765,((Y_paikka_pysty-0.1417)*-
1),0.3,0,3.14,0]
Poimintapiste_p
Z_poim_kulma:=d2r(Z_kulma)
Z_kulman_siirto:=p[0.0,0.0,0.0,0,0,Z_poim_kulma]
Poiminta_pose_5:=get_actual_tcp_pose()
Z_trans_pose:=pose_trans(Poiminta_pose_5,Z_kulman_siirto)
Z_trans_pose
ProtectiveS_est:=1
Waypoint_1
Call SubP_rq_close_a
ProtectiveS_est:=0
Poimintaa_yrit:=1
If rq_obj_detect $\neq$ 1

```

MoveJ

Poiminta_apupis

Mutterikaant_yl

Mutterinkaant_a

Waypoint_5

Call SubP_rq_open_an

Mutteri_har_alu:=1

Mutterinkaant_a

Waypoint_6

Call SubP_rq_close_a

Waypoint_7

ProtectiveS_est:=1

MoveL

Waypoint_3

Call SubP_rq_open_an

ProtectiveS_est:=0

MoveL

Waypoint_8

Mutteri_har_alu:=0

If rq_obj_detect \neq 1

MoveJ

Laatikko_lahest

'Palletoidaan mutterit laatikkoon.'

Pallet

Pattern: Square

a1st_Corner_1

a2nd_Corner_1

a3rd_Corner_1

a4th_Corner_1

PalletSequence

Approach_1

PatternPoint_1

Wait: 1.0

Call SubP_rq_close_a

```
Exit_1
If cnt_1≠0
'Kun kaikki kuusi mutteria palletoitu, tyhjennetaan laatikko.'
MoveJ
  Laatikonpoim_la
  Call SubP_rq_open_an
  Laatikonpoim_pi
  Call SubP_rq_close_a
  Laatikko_ylos
MoveJ
  Kaatopaikka
Loop 3 times
  MoveL
    Kaatoylos
    Kaatoalas
MoveJ
  Laat_pal_apu1
  Laat_pal_apu2
  Laatikonpalautu
MoveL
  Laatikonpoim_pi
  Call SubP_rq_open_an
  Laatikonpoim_la
MoveJ
  'Hehkutukset'
  Waypoint_12
  Loop 3 times
    Waypoint_13
    Waypoint_12
poiminta_epaonn:=0
var_1:=Advance_gocator
Wait: 0.5
X_paikka:=(X_a*65536+X_b)/1000000
Y_paikka:=(Y_a*65536+Y_b)/1000000
```

```

Z_paikka:=(Z_a*65535+Z_b)/1000000
If XYZ $\neq$ X_paikka+Y_paikka+Z_paikka
  X_paikka:=0
'Pysty kappaleen muuttajat'
X_paikka_pysty:=(X_a_pystykpl*65536+X_b_pystykpl)/1000000
Y_paikka_pysty:=(Y_a_pystykappal*65536+Y_b_pystykpl)/1000000
If XYZ_pystykpl $\neq$ X_paikka_pysty+Y_paikka_pysty
  X_paikka_pysty:=0
Else
  MoveL
  Waypoint_4
  If Poimintaa_yrit $\neq$ 1
    'Oletettavasti kayttajan toimesta johtuva nuhtelu.'
    MoveJ
    Waypoint_14
    Call SubP_rq_close_a
    Loop 3 times
    MoveJ
    Waypoint_14
    Waypoint_15
    Poimintaa_yrit:=0
    X_paikka:=0
    X_paikka_pysty:=0
    poiminta_epaonn:=poiminta_epaonn+1
'Thread_1'
SubP_rq_close_a
  rq_close_and_wait()
SubP_rq_open_an
  rq_open_and_wait()
SubP_rq_activat
  rq_activate_and_wait()
SubP_rq_set_for
  rq_set_force(rq_force)

```

SubP_rq_set_spe

rq_set_speed(rq_speed)

SubP_rq_open

rq_open()

SubP_rq_is_obje

rq_is_object_detected()

SubP_rq_close

rq_close()

LIITE 2

Python-ohjelma kameran koordinaattipisteiden laskemiseksi robotin peruskoordinaattistossa.

```
# -*- coding: latin-1 -*-
```

```
import math
```

```
# Akselien kiertymien erotusten kysely rhoo = x-akselin suhteen (pitch), theeta = y-akselin (yaw) ja fii = Z-akselin (roll) suhteen
```

```
kysely_onnistunut = 0
```

```
while kysely_onnistunut == 0:
```

```
    try:
```

```
        rhoo = float(input("rhoo ? (asteina) "))
```

```
        theeta = float(input("theeta ? "))
```

```
        fii = float(input("fii ? "))
```

```
    except ValueError:
```

```
        print("Virheellinen luku!")
```

```
        continue
```

```
    except NameError:
```

```
        print("Kirjoita vain lukuasteina!")
```

```
        continue
```

```
    kysely_onnistunut = 1
```

```
# Muunnetaan radiaaneiksi
```

```
rhoo_rad = math.radians(rhoo)
```

```
theeta_rad = math.radians(theeta)
```

```
fii_rad = math.radians(fii)
```

```
def cos(kulma_rad):
```

```
    return math.cos(kulma_rad)
```

```

def sin(kulma_rad):
    return math.sin(kulma_rad)
# Kiertymä matriisin laskeminen, johtaminen:  $R_{xyz} = X_r * Y_r * Z_r$ 
R_xyz = [cos(fii_rad)*cos(theeta_rad), cos(fii_rad)*sin(theeta_rad)*sin(rhoo_rad) -
sin(theeta_rad)*cos(rhoo_rad), cos(fii_rad)*sin(theeta_rad)*cos(rhoo_rad) +
sin(fii_rad)*sin(rhoo_rad),
sin(fii_rad)*cos(theeta_rad), sin(fii_rad)*sin(theeta_rad)*sin(rhoo_rad) +
cos(fii_rad)*cos(rhoo_rad), sin(fii_rad)*sin(theeta_rad)*cos(fii_rad) -
cos(fii_rad)*sin(rhoo_rad),
-sin(theeta_rad), cos(theeta_rad)*sin(rhoo_rad), cos(theeta_rad)*sin(rhoo_rad)]
laskin = 0
for alkio in R_xyz:
    print(alkio)
    laskin += 1
    if laskin % 3 == 0:
        print "\n"

# Paikkamuutosten lisääminen ja homogeenisen translaatiomatriisin luominen

kysely_onnistunut2 = 0
while kysely_onnistunut2 == 0:

    try:
        delta_X = float(input("X-koordinaattien erotus "))
        delta_Y = float(input("Y-koordinaattien erotus "))
        delta_Z = float(input("Z-koordinaattien erotus "))

    except ValueError:
        print("Virheellinen luku!")
        continue
    except NameError:
        print("Kirjoita vain lukuarvoina!")
        continue
    kysely_onnistunut2 = 1

```



```

X = []
Y = []

X_rivi1 = []
X_rivi2 = []
X_rivi3 = []
X_rivi4 = [0,0,0,1]
alkio = []
laskin = 0
for alkio in R_xyz:
    laskin += 1
    if laskin < 4:
        X_rivi1.append(alkio)
    elif laskin == 4:
        X_rivi1.append(delta_X)
        X_rivi2.append(alkio)
    if laskin < 7 and laskin > 4:
        X_rivi2.append(alkio)
    elif laskin == 7:
        X_rivi2.append(delta_Y)
        X_rivi3.append(alkio)
    if laskin < 10 and laskin > 7:
        X_rivi3.append(alkio)
    if laskin == 9:
        X_rivi3.append(delta_Z)

laskin = 0
X.append(X_rivi1)
X.append(X_rivi2)
X.append(X_rivi3)
X.append(X_rivi4)

for rivi in X:

```

```

laskin += 1
print rivi
if laskin % 3 == 0:
    print "\n"

# Kysytään kuvakoordinaatiston pistettä -> näytetään base-koordinaatistossa

kysely_onnistunut3 = 0
while kysely_onnistunut3 == 0:

    try:
        Kamera_X = float(input("Kameran antamat X-koordinaatit "))
        Kamera_Y = float(input("Kameran antamat Y-koordinaatit "))
        Kamera_Z = float(input("Kameran antamat Z-koordinaatit "))

    except ValueError:
        print("Virheellinen luku!")
        continue
    except NameError:
        print("Kirjoita vain lukuarvoina!")
        continue

    kysely_onnistunut3 = 1
# 4 X 1-matriisi
#Y.append(Kamera_X)
#Y.append([Kamera_Y)
#Y.append(Kamera_Z)
#Y.append([1])
Y = [Kamera_X, Kamera_Y, Kamera_Z, 1]

# Matriisi muunnoksen suorittaminen
tulos = [0,0,0,0,]
# iteroidaan X:n rivit
for i in range(len(X)):
    # iteroidaan Y:n rivit

```

```
for k in range(len(Y)):

    tulos[int(i)] += X[int(i)][int(k)] * Y[int(k)]
laskin = 0
for t in tulos:
    laskin += 1
    if laskin == 1:
        print "Robotin X-koordinaatti: ", t
    elif laskin == 2:
        print "Robotin Y-koordinaatti: ", t
    elif laskin == 3:
        print "Robotin Z-koordinaatti: ", t
    elif laskin == 4:
        print t
```


GOCATOR 2380 A -PIKAKÄYTTÖOHJEET WINDOWS 7, 8 JA 10



1. KÄYTTÖLIITTYMÄN AVAAMINEN

Esivaatimukset:

- varmista että Gocator on liitetty kiinni tietokoneen ethernet-porttiin
- varmista että sensorin power-valo on päällä
- varmista että olet lukenut ja ymmärtänyt turvallisuusohjeet

1.1 Muuta tietokoneen verkkoasetuksia

Avaa ohjauspaneeli -> Verkko ja Internet -> verkko- ja jakamiskeskus -> Muuta sovittimen asetuksia.

Klikkaa oikealla painikkeella haluttua verkkoyhteyttä -> ominaisuudet.

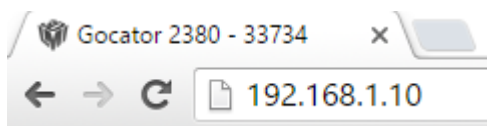
Valitse verkkoasetusten sivu -> Valitse listalta Internet Protocol Version 4 (TCP/IPv4) -> ominaisuudet.

Valitse kohta "Käytä seuraavaa IP-osoitetta".

Anna IP-osoitteeksi: ____ . ____ . ____ . ____ ja aliverkon peitteeksi: 255.255.255.0

-> OK.

1.2 Avaa selain ja aseta sensorin IP-osoite osoitepalkkiin -> Enter



Gocator-sensorissa oletuksena olevat verkkoasetukset:

Asetus	Oletusarvo
DHCP	Estetty
IP-osoite	192.168.1.10
Aliverkon peite	255.255.255.0
Gateway	0.0.0.0

1.3 Valitse kieli (Language) ja paina Login:



HUOM!

”Ylläpitäjä” (Administrator) on oletuksena tyhjä ja sitä *ei* tarvitse muuttaa kirjautuakseen sisään.

2. SKANNAAMINEN

2.1 Valitse Scan-sivu vasemman yläkulman valintapalkista:

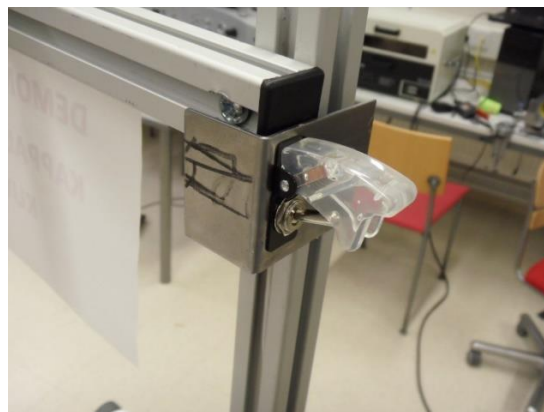


- 2.2 Varmista että oikealta yläkulmasta löytyvä Replay-tila on pois päältä:**
(liukukytкин on vasemmalla).



Replay-liukukytкин

- 2.3 Aseta laserin turvakytкин sallivaan tilaan**
(sensorin STATUS kohtaan syttyy LASER-ledi ks. sivu 6, YLEISNÄKYMÄ)



- 2.4 Paina Start-painiketta:**
(sensori käynnistää laserin)

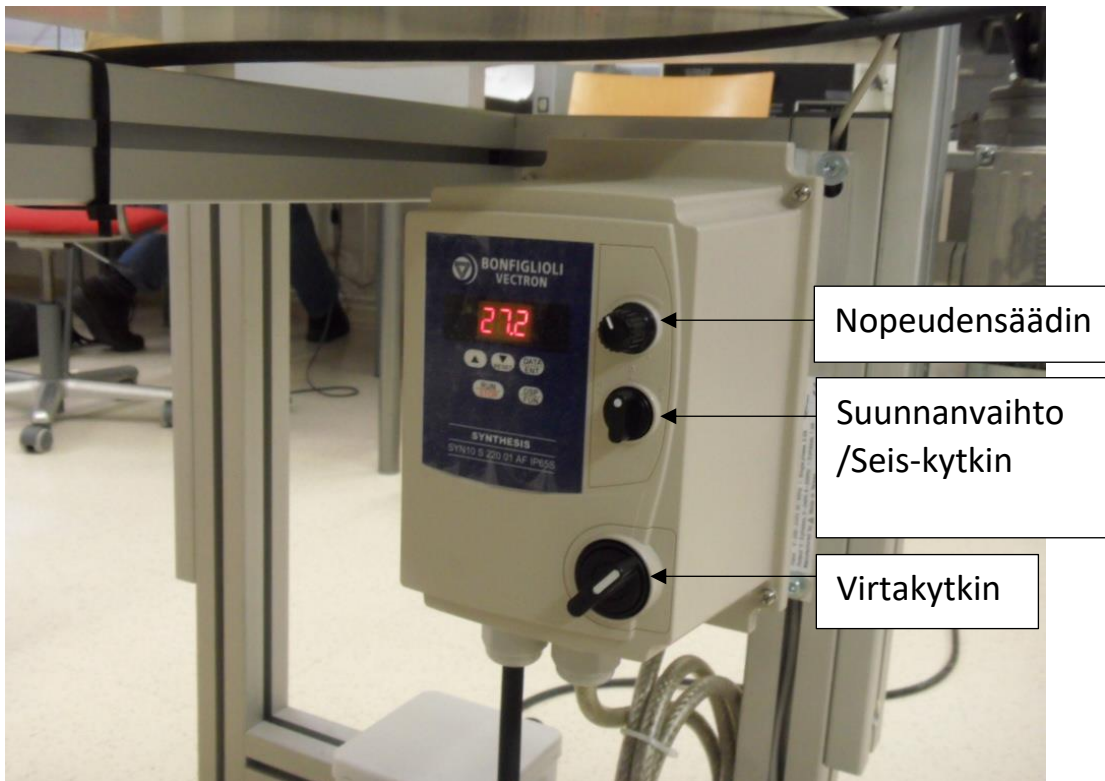


Start-painike

HUOM! Älä katso suoraan kohti laseria!

2.5 Lämpivalaise kohde laserilla.

Aseta valaistava kappale liukuhihnan toiseen päähän. Ohjaa liukuhihnaa taajuusmuuntajan säätimistä (liukuhihna liikkuu, kun virtakytkin on I- asennossa ja suunnanvaihtokytkin on joko R- tai F- asennossa -> Ks. kuva). Pysäytä liukuhihna kun kappale on kokonaan valaistu (-> suunnanvaihto O- asentoon).

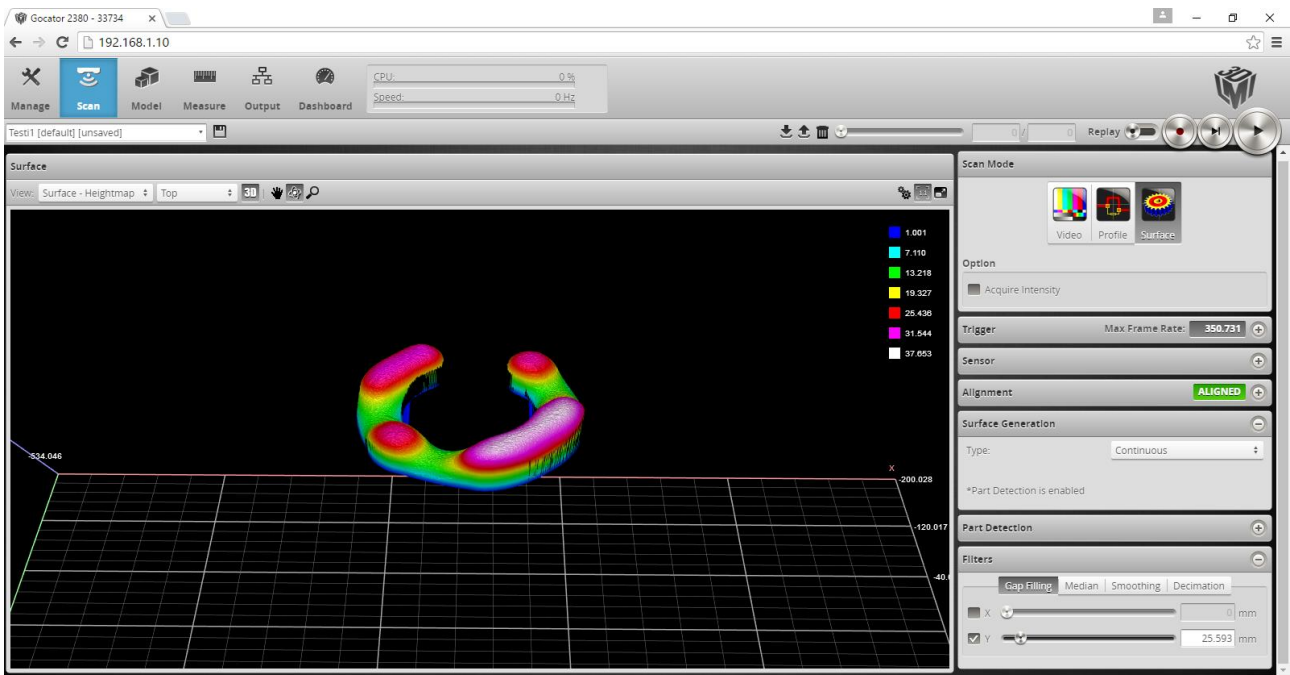


2.6 Paina Stop-painiketta lopettaaksesi skannaaminen ja tarkastele kohdetta:

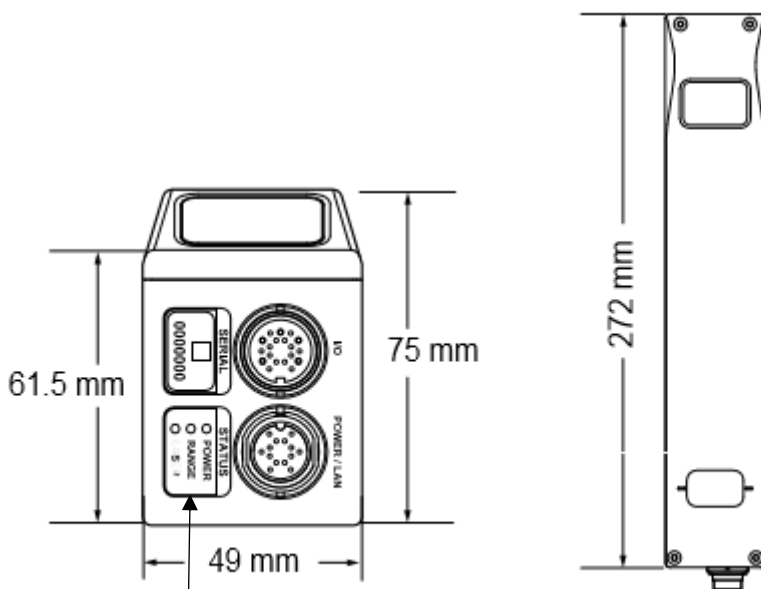


(Ks. tarkemmat ohjeet Gocator-manuaalista.)

ESIMERKKI SKANNAUKSESTA



YLEISNÄKYMÄ



STATUS-tila:
 POWER- LED
 RANGE- LED
 LASER- LED

Kuva1. Gocator 2380 Yleisnäkymä (LMI Technologies 2016).

TURVALLISUUSOHJEET

LASER-TURVALLISUUS

”Luokka 3B

Luokka 3B käsittää laserit, joiden säteily ylittää luokan 3R emissiorajat. Jatkuvatoimisen luokan 3B laserin suurin sallittu säteilyteho on 500 mW. Luokkaan 3B kuuluvan laserin suora ja peiliheijastunut säde on aina vaarallinen silmille. Luokan 3B laser voi aiheuttaa pieniä ihovaurioita. Muun muassa tutkimuksessa käytetään usein luokan 3B lasereita.” (STUK, 2015.)



Kuva2. Laserturvallisuusluokka 3B:n varoitusmerkkejä (STUKLEX 2015).

SÄHKÖTURVALLISUUS

Sensorit tulee olla maadoitettu

Kaikki sensorit tulee kytkeä kotelonsa kautta maahan. Kaikki sensorit pitäisi asentaa maadoitettuun kehykseen, käyttäen sähköisesti johtavaa laitteistoa, varmistaakseen että sensorin kotelointi on yhdistetty maahan. Käytä yleismittaria tarkistaaksesi sensorin ja maadoituksen jatkuvuus, ja varmistaaksesi toimiva yhteys.

Minimoi systeemin ja sensorin maadoituksen potentiaaliero

On huolehdittava jännitepotentiaalminimoimisesta anturin ja systeemin (maapotentiaalreferenssi I/O – signaaleille) maadoituksen välillä. Tämä jännitepotentiaali voidaan määrittää mittaamalla jännitettä Analog out:in ja systeemin maadoituksen välillä. Sallittu maksimi jännitepotentiaali on 12V, mutta suositellaan pitämään alle 10V:ssa, sarja- ja enkooderi-yhteyksien vaurioitumisen välttämiseksi.

Käytä sopivaa virransyöttöä

Käytä sopivaa virransyöttöä +24 - +48V DC. Gocator-sensoreiden virransyötön tulisi olla eristetty syöttö, varustettuna käynnistysvirran suojaimella tai sen pitäisi kestää korkeaa kapasitiivista kuormaa.

Katso Gocator-manuaalin kohdasta *Gocator 2300 & 2880 I/O Connector* sivulta 409 Gocator 2300-sarjan liittimien esittelyyn.

Käsittele virtalaitteita varoen

Johtimiin ei tulisi koskea kun sensori on käynnissä, koska se voi johtaa käyttäjän saamaan sähköiskuun tai laitteiston vaurioitumiseen.

(käännetty Gocator-manuaalista s.11-16)

Tekijä: Kalle Jantunen

LÄHTEET

LMI Technologies, 2015. Viitattu 21.3.2016. http://downloads.lmi3d.com/system/files/15159-4.4.4.40_MANUAL_User_Gocator-2100-2300-2880-Series.pdf

LMI Technologies, 2015. Viitattu 21.3.2016.

http://downloads.lmi3d.com/system/files/Gocator/documents/Gocator%202300%20Series/15173-1.11_Manual_Quickstart_Gocator-2100A-2300A-Series_WEB.pdf

LMI Technologies, 2015. Viitattu 21.3.2016.

http://downloads.lmi3d.com/system/files/Gocator/documents/Gocator%202300%20Series/DATA_SHEET_Gocator_2300_WEB_EN.pdf

STUK. Viitattu 21.3.2016. <http://www.stuk.fi/aiheet/laserit/laserluokat>

STUKLEX. Viitattu 21.3.2016. <http://plus.edilex.fi/stuklex/fi/lainsaadanto/saannosto/ST9-4>

International standard IEC 60825-1 (2001 - 08) consolidated edition, Safety of laser products – Part1: Equipment classification, requirements and user's guide.

Technical report 60825-10, Safety of laser products –Part 10. Application guidelines and explanatory notes to IEC60825-1.

Laser Notice No. 50, FDA and CDRH <http://www.fda.gov/cdrh/rad-health.html>