

Tuotepakkauksen tarkastus teollisuus-robotilla ja konenäöllä

Joel Keski-Kuha

Opinnäytetyö
Marraskuu 2016
Tekniikan ja liikenteen ala
Insinööri (AMK), Automaatiotekniikan tutkinto-ohjelma

Tekijä(t) Keski-Kuha, Joel	Julkaisun laji Opinnäytetyö, AMK	Päivämäärä Marraskuu 2016
	Sivumäärä 50+4	Julkaisun kieli Suomi
		Verkkojulkaisulupa myönnetty: x
Työn nimi Tuotepakkauksen tarkastus teollisuusrobotilla ja konenäöllä		
Tutkinto-ohjelma Automaatiotekniikan tutkinto-ohjelma		
Työn ohjaaja(t) Rantapuska, Seppo Flyktman, Teppo		
Toimeksiantaja(t) Vision Systems Oy Rahkola, Kari		
Tiivistelmä <p>Opinnäytetyön tavoitteena oli tarkastella konenäön ja teollisuusrobotin soveltuvuutta asiakkaan laadunvalvontatarpeisiin. Asiakkaan toivomuksena oli korvata automaatiolla henkilökuntaa sitova laadunvalvonta. Opinnäytetyön tulosten pohjalta tulotisiin mahdollisesti toteuttamaan varsinainen laaduntarkkailulaitteisto asiakkaan tiloihin. Soveltuvuuden varmistamiseksi oli otettava selvää minkälainen laitteisto kykenisi toimimaan asiakkaan toivomalla nopeudella ja silti tuottamaan tarkkoja tuloksia.</p> <p>Soveltuvuuden varmistamiseksi opinnäytetyössä toteutettiin testisolu Jyväskylän ammattikorkeakoulun automaatiolaboratoriotiloihin, jossa oli käytettävissä kaikki tarvittavat laitteistot ja ohjelmistot. Testisolussa käytettiin ABB:n robottia ja Cognex 7200- sarjan mustavalkokameraa. Testisolun laitteistojen ohjelmointiin käytettiin ABB RobotStudio ja Cognex In-Sight- ohjelmia. Käyttöliittymä luotiin käyttämällä RobotStudion Screenmaker-nimistä lisäosaa.</p> <p>Opinnäytetyössä käsitellään käytettyihin laitteisiin liittyvän teorian lisäksi myös muitakin robotti- ja konenäkökameratyyppisiä, sekä pohditaan niiden soveltuvuutta samaan tehtävään. Lisäksi perehdytään itse testisolun toteuttamiseen, tarkistuksen nopeuteen ja tarkkuuteen.</p> <p>Opinnäytetyön tuloksena varmistui, että teollisuusrobotin ja konenäön yhteistyöllä on mahdollista tarkistaa tuotteita asiakkaan tuotantolinjalla nopeasti ja varmasti. Koulun laboratorioon toteutetulla testisolulla ei aivan saavutettu toivottua nopeutta, mutta jatkokehityksellä tavoitteeseen voidaan päästä.</p>		
Avainsanat (asiasanat) ABB, Cognex, In-Sight, konenäkö, RobotStudio, Screenmaker, teollisuusrobotti		
Muut tiedot		

Author(s) Keski-Kuha, Joel	Type of publication Bachelor's thesis	Date November 2016
		Language of publication: Finnish
	Number of pages 50+4	Permission for web publication: x
Title of publication Inspection of product package with industrial robot and machine vision		
Degree programme Automation engineering		
Supervisor(s) Rantapuska, Seppo Flyktman, Teppo		
Assigned by Vision Systems Oy Rahkola, Kari		
Abstract <p>The aim of the thesis was to determine if machine vision and industrial robot solution is a feasible option for the quality control needs of the company's clients. The assigner wished to replace quality control that was tying up personnel with an automated system. To determine if such project was feasible it was necessary to find out if an automated system with fast enough speed and precision could be made. The finalized implementation could then be made with the knowledge gained from the thesis.</p> <p>A test cell was constructed at the automation laboratory of JAMK University of Applied Sciences. JAMK provided all used devices and software licenses. The test cell used ABB robot and Cognex monochrome machine vision camera. The test cell was programmed using ABB RobotStudio and In-Sight programs. Additionally, the user interface was implemented with Screenmaker add-on to RobotStudio.</p> <p>The thesis not only contains theory about the devices used but also about other types of industrial robots and machine vision cameras that could possibly be alternatively used. Design, building and testing of the test cell is also included. Speed and reliability of product inspection were the factors that were used to measure the success of the project.</p> <p>The results of the thesis confirm the feasibility of using machine vision and industrial robot together to provide reliable quality control. The test cell built at the school's automation laboratory did not quite reach the speed requested by the company, however, with additional development it seems reachable.</p>		
Keywords/tags (subjects) ABB, Cognex, industrial robot, In-Sight, machine vision, RobotStudio, Screenmaker		
Miscellaneous		

Sisältö

1	Johdanto	5
1.1	Lähtökohdat.....	5
1.2	Vision Systems Oy	5
2	Robotit	6
2.1	Määritelmä	6
2.2	Historia	6
2.3	Hyödyt ja haitat	7
2.4	Käyttökohteet.....	8
2.4.1	Käsittely	8
2.4.2	Manipulointi.....	8
2.4.3	Laadunvalvonta	9
2.4.4	Lääketeide.....	9
2.5	Robotin osat	9
2.6	Vapausasteet	10
2.7	Kinematiikka	11
2.7.1	Suora kinematiikka	11
2.7.2	Käännteinen kinematiikka	11
2.8	Dynamiikka	12
2.9	Sensorit.....	12
2.9.1	Paikkasensorit	12
2.9.2	Nopeussensorit	14
2.9.3	Kiihtyvyyssensorit.....	14
2.9.4	Kosketussensorit	14
2.9.5	Voimasensorit	14
2.9.6	Lähestymissensorit.....	15
2.9.7	Etäisyysensorit.....	15

	2
2.9.8 Turvallisuussensorit.....	15
2.9.9 Konenäköjärjestelmät	16
2.10 Robottityypit.....	16
2.11 Robottien ominaisuudet	18
2.11.1 Kuormankantokyky	18
2.11.2 Robotin toimintasäde	19
2.11.3 Toistotarkkuus ja Absoluuttinen tarkkuus.....	19
2.12 Koordinaatistot	19
2.13 Ohjelmointitavat.....	20
2.14 ABB robotin liike	21
2.14.1 Liiketyypit.....	21
2.14.2 Paikkapisteet	21
2.14.3 Liikenopeus	22
2.14.4 Paikoitustarkkuus	22
2.14.5 Koordinaatistot ja niiden hierarkia	22
3 Konenäkö	24
3.1 Käyttökohteet.....	24
3.2 Kuvakenno	25
3.3 Optiikka	25
3.4 Valonlähteet	27
3.5 Valaistustekniikat.....	28
3.6 Konenäön perustoimintoja	30
3.7 Tiedonsiirto.....	31
4 Käytetyt laitteistot ja ohjelmistot	32
4.1 ABB IRB-120 teollisuusrobotti	32
4.2 Robotstudio	33
4.3 Cognex 7200 kamera	34

4.4	In-sight Explorer.....	34
5	Tavoitteet.....	35
6	Toteutus.....	36
6.1	Konenäköohjelma.....	39
6.2	Robotin ohjelma.....	41
6.3	Käyttöliittymä.....	44
7	Testaus.....	45
8	Tulokset.....	46
9	Pohdinta.....	47
10	Lähteiden luotettavuus.....	48
	Lähteet.....	50
	Liitteet.....	51

Kuviot

Kuvio 1 (a) Pyörivä inkrementaalinen enkooderi ja (b) lineaarinen inkrementaalinen enkooderi. (Niku 2001, 223.).....	13
Kuvio 2 Resolverin virtapiirikaavio. (Niku 2001, 228.).....	13
Kuvio 3 (a) Voimasensori ja (b) Wheatstonen silta. (Niku 2001, 232).....	15
Kuvio 4 Suorakulmainen manipulaattori. (Graig 2005, 234.).....	16
Kuvio 5 Sylinteri manipulaattori. (Graig 2005, 237.).....	17
Kuvio 6 Napakoordinaatisto manipulaattori. (Graig 2005, 236.).....	17
Kuvio 7 Kiertyvänivelinen manipulaattori. (Graig 2005, 235.).....	18
Kuvio 8 SCARA manipulaattori. (Graig 2005, 235.).....	18
Kuvio 9 Mahdollisia eri asentoja samalle paikkapisteelle. (ABB Robotics 2011, 32.)..	22
Kuvio 10 Koordinaatistojen hierarkia. (ABB Robotics 2011, 27.).....	23
Kuvio 11 Graafinen esitys kameran optiikasta. (Azad 2008, 31.).....	26
Kuvio 12 Eri valonlähteiden vertailutaulukko. (Azad 2008, 19.).....	27
Kuvio 13 Eri valaisusuuntia. (Azad 2008, 21.).....	28
Kuvio 14 Vasemmalla kohtisuoraan ja oikealla sivulta tuleva valaistus. (Azad 2008, 21.).....	29

Kuvio 15 Vasemmallla kuva ilman värisuodatusta ja oikealla punaisen suodattimen kanssa. (Azad 2008, 24.).....	29
Kuvio 16 Polarisaatiosuodattimen vaikutus heijastuksiin. (Azad 2008, 24.).....	30
Kuvio 17 Kuvan kynnystys. (Sonka 2008, 177.)	31
Kuvio 18 Kuvan reunantunnistus. (Sonka 2008, 186.)	31
Kuvio 19 IRB 120 robotin toiminta-alue. (ABB Robotics 2012.)	32
Kuvio 20 Simulaatio käytetystä robotista RobotStudiossa.	33
Kuvio 21 Screenmaker lisäosa.	33
Kuvio 22 Easybuilder näkymä.	34
Kuvio 23 Spreadsheet näkymä.	35
Kuvio 24 Virheellinen etiketti pullossa.....	36
Kuvio 25 Demosolu ensimmäisessä kokoonpanossa.....	37
Kuvio 26 Toisessa kokoonpanossa käytetty metallinen tarttuja.....	38
Kuvio 27 Demosolu toisessa kokoonpanossa.....	39
Kuvio 28 Pakkauksen etupuolen näkymä In-Sight ohjelmassa tarkistuksen aikana. ..	40
Kuvio 29 Pakkauksen takapuolen näkymä In-Sight ohjelmassa tarkistuksen aikana. .	41
Kuvio 30 Robotti noutamassa näytettä.....	42
Kuvio 31 Kamera ottamassa kuvaa pakkauksen etupuolelta.....	43
Kuvio 32 Käyttöliittymä robotin Flexpendant ohjaimessa.	45

1 Johdanto

1.1 Lähtökohdat

Opinnäytetyön toimeksiantajan asiakkaan käyttämä, tuotepakkauksiin printtejä tuostava laite tuotti satunnaisesti viallisesti tulosteita. Laite myös vaati ajoittain säätötoimia tuotannon laadun ylläpitämiseksi. Laitteen yleisimmät tulostusvirheet liittyivät eri värikerrosten virheelliseen kohdistukseen toisiinsa nähden ja tekstitulostuksen tarkkuuden heikentymiseen. Pakkausten laadunvalvonta myös satoi kaksi työntekijää pois muista tuotantotehtävistä. Asiakas toivoi, että järjestelmä kykenisi tarkistamaan jokaisen tuotepakkauksen. Tahtiaika tuotannossa oli yksi sekunnissa. Linjalla käsiteltiin useita erimuotoisia ja värisiä purkkeja. Toteutetun järjestelmän tuli kyetä tunnistamaan nopeasti ja varmasti kaikki halutut virheet pakkauksen etu- ja takapuolelta.

Asiakas toivoi Vision Systems Oy:ltä selvitystyötä, kuinka työvaihe voitaisiin automatisoida käyttäen teollisuusrobotteja ja konenäköä. Vision Systemsillä ei ollut omilla tiloillaan tarvittavia laitteita, joten yritys lähestyi Jyväskylän ammattikorkeakoulua mahdollisella opinnäytetyöprojektilla. Opinnäytetyöprojekti käynnistyi alkuvuodesta 2016.

Työn teoriaosuudessa käsitellään oleelliset asiat liittyen teollisuusrobotteihin ja konenäköön. Teoriaosuuden jälkeen esitellään käytetyt laitteistot ja ohjelmistot, sekä projektin toteutus, testaus ja tulokset. Lopuksi pohditaan vielä muita mahdollisia tapoja toteuttaa tarkistus.

1.2 Vision Systems Oy

Työn toimeksiantajana toimi vuonna 1985 Jyväskylään perustettu Vision Systems Oy joka tuottaa konenäköä tai optoelektroniikkaa hyödyntäviä mittaus-, ohjauk- ja laadunvalvontaratkaisuja. Vuonna 2014 perustettiin tytäryhtiö Vision Development, joka keskittyy ohjelmistojen ja järjestelmien suunnitteluun. Yhtiö myös laajentui Tampereelle kesäkuussa 2015 ollakseen lähempänä asiakkaitaan. Yhtiöllä on tarkoituksena laajentua muillekin paikkakunnille. Yhtiö myy lähinnä omia tuotteitaan mutta tarjoaa myös monia edustustuotteita. Robotiikka on yhtiön viimeisimpiä laajentumisaloja. (Vision Systems 2016)

Toiminnallaan Vision Systems pyrkii parantamaan asiakkaidensa kilpailukykyä ja tuotavuutta tarjoamalla luotettavia ratkaisuja teollisuuden tarpeisiin. Yhtiön toiminta on voimakkaasti kansainvälistä. Viennin osuus liikevaihdosta on noin 40 prosentti ja on kasvussa koko ajan. (Vision Systems 2016)

2 Robotit

2.1 Määritelmä

Näennäisesti samankaltaiset laitteet robotin ja manipulaattorin erottaa toisistaan se, että robottia ohjaa tietokone tai tietokoneen kaltainen laite, kun taas manipulaattoria ohjaa operaattori. Robotin ohjelma on yleensä helposti muokattavissa tarpeen mukaan. Robotti on täten hyvin joustava eikä koko laitetta tarvitse uudelleen suunnitella. Eri mailla on poikkeavia standardeja, joiden mukaan robotti määritellään. Amerikkalaisen standardin mukaan laite pitää olla helposti uudelleenohjelmoitavissa, jotta sitä voitaisiin pitää robottina. Täten kiinteää sekvenssiä tekeviä tai käyttäjän jatkuvasti manuaalisesti operoimia laitteistoja ei pidetä robotteina. (Niku 2001, 2.; Sciavicco 2002, 4.)

Japanissa useammat laitteet luokitellaan roboteiksi. Robottijärjestö JIRA luokittelee robotit kuuteen luokkaan, joista luokka yksi ja kaksi eivät muualla kuulu roboteiksi luokiteltaviin. (Niku 2001, 2.)

- Luokka 1: Operaattorin käyttämä monivapausasteinen laite.
- Luokka 2: Laite joka toistaa samaa työtä ja on vaikeasti muokattavissa muuhun työhön.
- Luokka 3: Sama kuin luokka 2 mutta helpommin muokattavissa.
- Luokka 4: Johdattamalla opetettu robotti, joka toistaa operaattorin sille opettamat liikkeet.
- Luokka 5: Numeerisesti ohjelmoitu robotti joka on ohjelmoitavissa.
- Luokka 6: Älykäs robotti joka ymmärtää ympäristöään ja kykenee suorittamaan tehtävänsä onnistuneesti, vaikka ympäröivät olosuhteet muuttuisivat.

2.2 Historia

Toisen maailmansodan jälkeen teollisuudessa aloitettiin käyttämään ohjelmoitavia automaattisia laitteita tuotannon tehostamiseksi. Samaan aikaan kehitettiin monen

vapausasteen manipulaattoreita radioaktiivisen materiaalin käsittelyyn. Ohjelmoitavuuden ja manipulaattoreiden yhteen liitoksesta syntyivät ensimmäiset yksinkertaiset robotit. Näiden robottien ohjelmat olivat tallennettuna paperille reikinä, josta valosilmät ne lukivat ne sähköisiksi ohjaussignaaleiksi. Teknologian kehittyessä siirryttiin ohjauksessa magneettinauhoihin, muistilaitteisiin ja lopulta tietokoneisiin. (Niku 2001, 4.)

2.3 Hyödyt ja haitat

Robottiikka ja automaatio voivat useissa tilanteissa lisätä työn tehokkuutta, turvallisuutta ja tasalaatuisuutta. Robotit voivat työskennellä ympäristöissä joissa ihmisten oleskelu on hankalaa tai mahdotonta. Robotit voivat myös työskennellä tauotta kellon ympäri laadun heikentymättä ja huomattavasti ihmistä suuremmalla tarkkuudella ja nopeudella. Sensoreilla robotti kykenee myös reagoimaan tarkasti moneen muuttujaan sekä työstämään samanaikaisesti useampaa työkappaletta. Roboteilla toteutettu tuotantoautomaatio on hyvin joustavaa ja helposti muokattavissa erilaisten tuotteiden käsittelyyn. (Niku 2001, 5.; Sciavicco 2002, 4.)

Robotit eivät myöskään tarvitse suojarusteita kuten suojavaatteita tai suojalaseja. Ne tarvitsevat kuitenkin suhteellisen yksinkertaisia työtehtäviä jotka sisältävät mielellään paljon toistoa. (Niku 2001, 20.)

Robottien haittoihin lukeutuu, että ne korvaavat ihmistyöntekijöitä ja täten aiheuttaen ekonomisia ongelmia työttömien lisääntyessä. Robottien lisääntyessä yhä enemmän ihmisiä joutuu työttömäksi, jolloin heillä ei ole varaa ostaa niitä tuotteita joita robotit valmistavat. (Niku 2001, 25.)

Roboteilta puuttuu kyky reagoida ongelmiin ja vaaratilanteisiin, jos niitä ei ole otettu huomioon järjestelmän suunnittelu ja ohjelmointivaiheessa. Virheellisesti toimiva robotti voi aiheuttaa omaisuusvahinkoja tai vaaraa ihmisille. Turvajärjestelmiä tarvitaan suojamaan robotin lähellä liikkuvia työntekijöitä. Robottijärjestelmien suunnittelu ja asennus kustannukset ovat myös suhteellisen suuria. Lisäksi robottijärjestelmät tarvitsevat erillisen ohjelmoinnin jokaiseen työtehtävään. (Niku 2001, 6.)

2.4 Käyttökohteet

Robottien teolliset käyttökohteet voidaan karkeasti tiivistää materiaalin käsittelyyn, manipulointiin ja mittaukseen. (Sciavicco 2002, 4.)

2.4.1 Käsittely

Robotin kyky tarttua kappaleisiin ja paikasta toiseen siirtämiseen tekee siitä hyvän kandidaatin mm. materiaalinkäsittelyyn. Materiaalinkäsittelyssä siirrettäviin kappaleisiin ei tehdä muutoksia. (Sciavicco 2002, 4.)

Lastaus ja Purku- robotteja käytetään muiden laitteiden työstökappaleiden asettamiseen ja poistoon. Näissä tapauksissa robotti ei yleensä tee muuta kuin siirrä kappaleita paikasta toiseen. (Niku 2001, 20.)

Asettelu- robotteja käytetään tuotteiden paketointiin, lavaukseen tai hihnalle asetteluun seuraavaa työvaihetta varten. (Niku 2001, 20.)

Vaaralliset ympäristöt, joissa ihminen ei voi tai kykene pitkään selviytymään, ovat otollisia robotin käytölle. Vaikkapa radioaktiivisten materiaalien ollessa kyseessä ihmisten suojaaminen on vaikeaa. Robotit voivat kuitenkin liikkua ja toimia radioaktiivisissa tiloissa hyvin helposti. (Niku 2001, 23.)

2.4.2 Manipulointi

Materiaalin manipuloinnissa robotti muokkaa raaka-aineita kohti lopullista tuotetta. Prosessin aikana kappaleen ulkomuotoa voidaan muokata työstämällä tai siihen voidaan liittää lisää osia. Työn suorittamiseksi robotti kykenee käyttämään monia erilaisia työkaluja. (Sciavicco 2002, 5.)

Hitsaus- robotteja käytetään laajalti teollisuudessa kokoonpanossa. Autoteollisuus käyttää robotteja enimmäkseen hitsaukseen, jossa ne liittävät autonomiasta yhteen. Hitsausrobotit ovat yleensä suuria. (Niku 2001, 20.)

Maalaus- robotteja käytetään myös paljon autoteollisuudessa autojen korien maalaamiseen. Ihmisille tarpeeksi tuuletetun tilan järjestäminen maalausta varten on yleensä vaikeaa, joten robotit soveltuvat tähän hyvin toistuvaan työhön erinomaisesti. (Niku 2001, 21.)

Tuotteiden valmistuksessa robotti muokkaa jotain esinettä esimerkiksi poraamalla, liimaamalla tai leikkaamalla. Elektroniikkateollisuus käyttää paljon robotteja komponenttien asetteluun piirilevyille. (Niku 2001, 22.)

2.4.3 Laadunvalvonta

Materiaalin käsittelyn ja manipuloinnin lisäksi robotteja käytetään tuotteiden laadunvalvontaan. Yleensä tällöin robotin parina on jokin tarkkailulaite jota robotti käyttää kappaleen tarkastamiseen. Tällaisia robotteja käytetään laajasti muun muassa piirilevyjen tarkistamiseen, jolloin tarkasteltava kappale joko hyväksytään tai hylätään. (Niku 2001, 21.)

2.4.4 Lääketiede

Teollisuuden lisäksi robotteja käytetään myös lääketieteessä, jossa ne toimivat kirurgin apuna leikkaussalissa. Robotin avulla kirurgiseen työhön saadaan huomattavasti lisää tarkkuutta, jolloin robotti suorittaa kaikki mekaaniset työvaiheet kuten leikkauksen ja poraamisen. Robotin avulla voidaan myös suorittaa mikrokirurgisia leikkauksia, jotka olisivat muuten liian pieniä suoraan ihmisen suoritettavaksi. Leikkaukset voidaan myös tarvittaessa toteuttaa etäyhteyttä käyttämällä, jolloin operoivan kirurgin ei tällöin tarvitse olla potilaan kanssa samassa tilassa. (Niku 2001, 23-24.)

2.5 Robotin osat

Manipulaattori on robotin liikkuva osa, joka koostuu varsista, nivelistä ja alustasta. Lisäksi tarvitaan muita osia, että manipulaattorivarsi voidaan luokitella robotiksi. (Niku 2001, 6.)

Työkalu sijaitsee robotin varren päässä ja jonka kärjen paikasta robotin ohjelmisto yleensä pitää lukua. Tavallisin työkalu on tarttuja. Muita mahdollisia työkaluja ovat esimerkiksi hitsauspistooli, maaliruisku tai liimasuutin. Robotin tarttujassa yleisiä toimintaperiaatteita ovat avautuva ja sulkeutuva liike, magneettinen vetovoima tai alipaineella toimiva imu. Robotin toiminnan kannalta tarttuja on järjestelmää suunniteltaessa yksi tärkeimpiä osa alueita. (Suomen robotiikkayhdistys Ry 1999, 60.)

Yleensä robotin valmistaja toimittaa robotin mukana vain yksinkertaisen tarttujan. Työkalu on yleensä helppo vaihtaa, joten käyttökohteeseen erityisesti suunniteltu ja toteutettu työkalu on yleensä helppo asentaa. (Niku 2001, 7.)

Toimilaitteet toimivat robotin voimanlähteinä. Useimmiten käytetään sähkömoottoreita kuten servo- tai askelmoottorit. Toimilaitteissa täytyy olla tarpeeksi voimaa, että robotti kykenee liikkumaan kuorman kanssa riittävän nopeasti ja tarkasti. Tarvittaessa suurempia voimamääriä saavutetaan käyttämällä pneumatiikkaa tai hydrauliiikkaa. (Niku 2001, 7.)

Sensorit keräävät robotin toiminnalle tärkeää tietoa robotin asennosta ja ympäristön tilasta. Liikkuakseen järkevästi robotin on tiedettävä missä asennossa jokainen sen liikkuvista nivelistä on. Jokainen nivel on varustettu sensorilla, joka lähettää robotin kontrollerille jatkuvaa tietoa paljonko nivel on liikkunut. Robottiin on myös voitu kytkeä ulkoista tilaa havainnoivia sensoreita kuten konenäkökameroita. (Niku 2001, 7.)

Kontrolleri on robotin aivot ja on vastuussa robotin liikkeistä. Kontrolleri pitää sensordatan avulla lukua robotin nivelten asennoista. Nivelkulmien ja robotin varresta tallennetun tiedon perusteella kontrollerin prosessori kykenee laskemaan robotin työkalun tarkan sijainnin. Liikekäskeyn saatuaan kontrolleri pystyy päättämään miten paljon sen pitää liikuttaa kutakin niveltä päästäkseen uuteen haluttuun asentoon. Nykyaikaisissa roboteissa laitteen nopeutta ja voimaa voidaan myös kontrollerin kautta muokata. (Niku 2001, 7.)

2.6 Vapausasteet

Robotin vapausasteilla tarkoitetaan sitä, kuinka monen akselin suuntaisesti robotti voi liikkua. Kolmiulotteiseen tilassa olevan kappaleen sijainti voidaan ilmoittaa x-, y- ja z-koordinaatteina. Täten kolmen vapausasteen robotti kykenee siirtämään kappaleita kaikkiin pisteisiin kolmiulotteisessa tilassa, mutta voi lähestyä kappaleita vain yhdestä suunnasta. Esimerkiksi nosturi on kolmen vapausasteen laite. Paikkatiedon lisäksi kiinteillä kappaleilla voi olla lukematon määrä eri asentoja. Jotta kappaleen sijainti ja asento voitaisiin täysin kuvailla, on käytettävä kuutta eri asentotietoa. Täten jos robotissa on vähemmän kuin kuusi niveltä ovat sen mahdollisuudet manipuloida kappaleita rajalliset. (Niku 2001, 8.)

Teollisuudessa kuitenkin käytetään paljon alemman kuin kuuden vapausasteen robotteja. Joissain tapauksissa liikettä kaikilla akseleilla ei tarvita kappaleiden siirtämiseen, joten tällöin voidaan käyttää yksinkertaisempaa ja halvempaa robottia. Yksinkertaisempien robottien ohjelmointi on myös helpompaa ja niitä käytetäänkin laajasti teollisuudessa. Esimerkiksi piirilevyn kokoonpanoroboteissa on useimmiten 3.5 vapausastetta. Tuolloin tarvitaan kaksi akselia liikuttamiseen xy-tasossa, yksi komponentin suunnan kääntämiseen, sekä puoli vapausastetta komponentin nostoon ja laskuun. Puolta vapausastetta käytetään, kun nivelellä on rajalliset asentovaihtoehdot. Tällaiset robotit ovat kuitenkin varsin rajoittuneita eivätkä sovellu muunlaiseen työhön. (Niku 2001, 10.)

2.7 Kinematiikka

Kinematiikkalaskuissa käytetään tietoa robotin tukivarsien pituuksista, nivelten asennoista ja työkalun koosta. Useimmissa roboteissa työkalu on vaihdettavissa, jolloin työkalun dimensiot tietenkin vaikuttavat siihen missä robotin pää sijaitsee. Robotin paikka voidaan laskea käyttämällä suoraa tai käänteistä kinematiikkaa. (Niku 2001, 29.)

2.7.1 Suora kinematiikka

Suorassa kinematiikassa robotin työkalun paikka voidaan laskea käyttämällä tietoa robotin nivelten asennoista ja sen kiinteiden varsien pituuksista. Tällä menetelmällä ei kuitenkaan voida selvittää missä asennossa nivelten tulisi olla, että robotin työkalu olisi tietyssä pisteessä ja asennossa. Tämän selvittämiseksi joudutaan käyttämään käänteistä kinematiikkaa. (Niku 2001, 53.)

2.7.2 Käänteinen kinematiikka

Käänteisessä kinematiikassa lasketaan missä asennoissa robotin nivelten on oltava, jotta robotin työkalu on tietyssä sijainnissa ja asennossa. Laskenta tapahtuu siis käänteisesti verrattuna suoraan kinematiikkaan. Käänteinen kinematiikka on yleensä käytännöllisempää robotiikassa, sillä robotin kontrolleri tarvitsee näitä yhtälöitä tarvittavien nivelkulmien laskemiseksi, kun ohjelmassa robotin halutaan siirtyvän tiettyyn pisteeseen. (Niku 2001, 53.)

2.8 Dynamiikka

Kappaleen liikuttamiseksi on siihen kohdistettava voimaa. Robotin liikkeeseen tarvittava voima saadaan johdettua seuraavasta kaavasta.

$$F = m \cdot a, \text{ jossa } F = \text{Voima (N)}, m = \text{massa (kg)} \text{ ja } a = \text{kiikhtyvyyys } \left(\frac{\text{m}}{\text{s}^2}\right)$$

Jotta robotti kykenee liikkumaan halutulla nopeudella, on sen toimilaitteiden tuotettava juuri oikea määrä energiaa. Robotin kontrolleri laskee täten kullekin liikeradalle ja nopeudelle tarvittavat voimamäärät. Laskuissa liikkuvan massan inertia on yleensä tärkeämpi tieto kontrollerille kuin itse kappaleiden paino. Kuitenkin robotin fyysisen rakenteen monimutkaisuuden takia Newtonilaisen fysiikan käyttö robotin dynamiikan laskemiseen voi olla vaikeaa. Vaihtoehtona voidaan soveltaa Lagrangen mekaniikkaa, jota on robottien tapauksissa useimmiten helpompi käyttää. (Niku 2001, 119-120.)

2.9 Sensorit

Robottiikassa sensoreita käytetään robotin asennon selvittämiseen sekä ulkoiseen ympäristön havainnointiin. Robotin kontrolleri kykenee resolverien ja enkooderien lähettämistä tiedoista päättelemään missä asennossa kukin robotin nivelistä on. Ulkoiset sensorit voivat esimerkiksi toimia robotin näkönä tai tuntoaistina. Sensoreita valittaessa on hyvä ottaa huomioon niiden soveltuvuus tarpeeseen. Joissain tapauksissa samaan tarkoitukseen voi soveltua useampikin sensorityyppi. Tällöin paremmin soveltuva voidaan valita esimerkiksi hinnan, koon, resoluution tai tarkkuuden mukaan. Alla robotiikassa useimmiten käytettyjä sensorityyppejä. (Niku 2001, 219-220.)

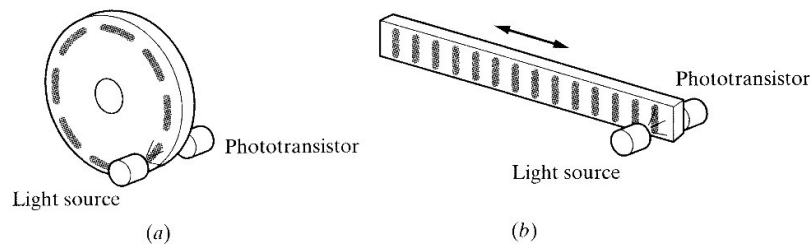
2.9.1 Paikkasensorit

Potentiometrit (Potentiometer) muuttavat paikkatiedon muuttuvaksi jänniteviestiksi liukuvastuksen avulla. Potentiometrejä käytetään usein palauttamaan tietoa nivelten asennoista. Niissä esiintyvät ongelmat lineaarisuuden ja resoluution kanssa kuitenkin rajoittavat niiden käyttöä. (Graig 2005, 252.; Niku 2001, 222.)

Enkooderit (Encoder) lähettävät signaalin aina tietyn lyhyen välimatkan välein. Enkooderi sisältää valonlähteen, enkooderikiekon ja valosensorin. Liike havainnoidaan,

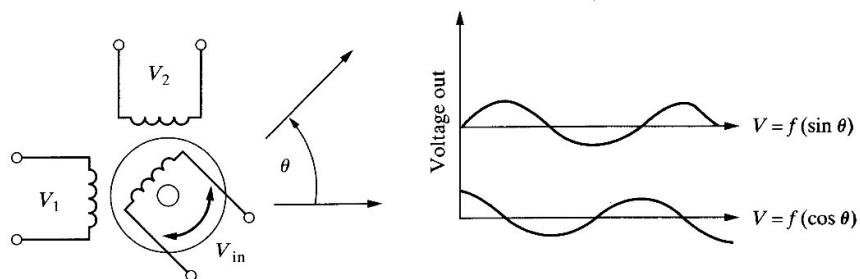
kun valo pääsee tietyin väliajoin enkooderikiekossa olevien reikien läpi valosensorille. Liikkeen nopeus voidaan määrittää valosensorilta tulevien signaalien tiheydestä. Enkooderi voidaan toteuttaa myös lineaarisena. (Niku 2001, 223.)

Enkoodereita on saatavilla inkrementaalisisena sekä absoluuttisina. Inkrementtianturilla signaaleja pitää laskea, asentoa ei siis tiedetä absoluuttisella tarkkuudella. Absoluuttisen enkooderin asento tiedetään aina tarkasti, sillä niissä käytetään useampaa sensoria ja reikäriviä samassa kiekossa. Kullakin asennolla on uniikki binäärinen koodinsa. Haittapuolena absoluuttisissa enkoodereissa on tarve useammalle digitaaliselle sisääntulolle. (Niku 2001, 223.)



Kuvio 1 (a) Pyörivä inkrementaalinen enkooderi ja (b) lineaarinen inkrementaalinen enkooderi. (Niku 2001, 223.)

Resolverit (Resolver) mittaavat kiertyvää liikettä. Kiertyvään akseliin on liitetty ensiökäämi, johon on johdettu jännite. Tällöin muodostuu magneettikenttä joka aiheuttaa jännitteen toisiokäämeihin. Resolverit toimivat siis muuntajien tavoin. Toisiokäämejä on kaksi ja ne ovat 90° kulmassa toisiinsa nähden. Niihin muodostuneiden jännitteiden vaihekulmista voidaan päätellä missä asennossa akseli on. Jännitteet muuttuvat riippuen missä kulmassa toisiokäämit ovat ensiökäämiin nähden. (Niku 2001, 228.)



Kuvio 2 Resolverin virtapiirikaavio. (Niku 2001, 228.)

2.9.2 Nopeussensorit

Jos robotissa on käytössä enkooderi, ei tarvita erillistä nopeussensoria. Silloin voidaan laskea laitteen nopeus signaalin pulssinopeudesta. Riippumatta käytetystä paikkasensoryypistä nopeus voidaan aina helposti laskea paikkatiedon muutosnopeudesta. Nopeuden määrittämiseen voidaan vaihtoehtoisesti myös käyttää takometriä. Takometrit ovat nopeussensoreita jotka toimivat generaattorin tavoin. Tällöin takometrin tuottama jännite on suoraan verrannollinen sen pyörimisnopeuteen. (Niku 2001, 230.)

2.9.3 Kiihtyvyyssensorit

Kiihtyvyyssensoreilla mitataan kappaleen kiihtyvyyttä. Teollisuusroboteissa niitä käytetään hyvin harvakseltaan. Niitä käytetään lähinnä tarkempien lineaaristen toimilaitteiden kanssa. (Niku 2001, 230.)

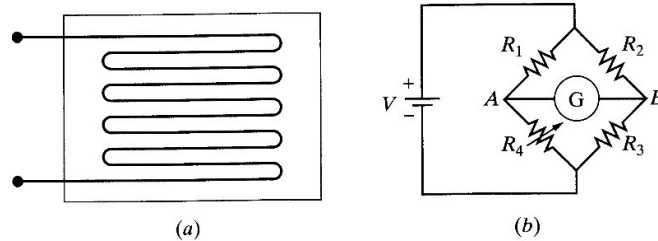
2.9.4 Kosketussensorit

Kosketussensoreilla voidaan havaita, onko robotti kosketuksissa jonkun kappaleen kanssa. Yksinkertaisimmillaan se voidaan toteuttaa käyttämällä mikrokytkintä tai valosensoria valonlähteen kanssa. Mikrokytkimet ovat pieniä kytkimiä joita käytetään laajasti esimerkiksi robotin turvajärjestelmissä tai kontaktin varmistamisessa. Mikrokytkimet ovat kestäviä ja halpoja. Valosensorit ovat vastuksia jotka reagoivat valon määrään. Valosensorin vastus laskee valon määrän lisääntyessä, joten valon määrä voidaan laskea mittaamalla sensorin läpi kulkevasta virrasta. Tarvittaessa tarkempaa havainnointia järjestelmissä voidaan käyttää voimasensoreita. (Niku 2001, 230-234.)

2.9.5 Voimasensorit

Voimasensoreita käytetään robottien tarttujissa silloin kun on tarve rajoittaa käsiteltävään kappaleeseen kohdistuvaa puristusvoimaa. Voimasensorit ovat vastuksia joiden vastusarvo muuttuu suhteessa niihin kohdistuvaan voimaan. Vastusarvon lukemiseen käytetään usein Wheatstonen siltaa. Lämpötilan muutokset aiheuttavat voimasensoreissa mittavirheen, ellei sitä ole jollakin tapaa kytkennässä kompensoitu.

Kompensointiin voidaan käyttää toista kuormaamatonta voimasensoria, joka on kytketty Wheatstonen siltaan jonkin vastuksen tilalle. (Niku 2001, 231.; Sciavico 2002, 312)



Kuvio 3 (a) Voimasensori ja (b) Wheatstonen silta. (Niku 2001, 232)

2.9.6 Lähestymissensorit

Lähestymisantureita käytetään selvittämään, onko kappaleita tietyllä etäisyydellä. Anturi lähettää signaalin, kun kappale on havaittu anturille ominaisella etäisyydellä. Niiden käyttö liittyy usein robotin navigointiin. Lähestymisanturien toiminta voi perustua magnetismiin, valoon tai ultraääneen. Magnetismilla toimivat anturit reagoivat muutoksiin magnaattikentässä. Valon perustuvat sisältävät lähettimen ja vastaanottimen. Tällöin seurataan joko valon heijastumista tai peittymistä. Ultraäänessä anturi lähettää ultraäänipulssin joka heijastuu takaisin kohteesta. (Niku 2001, 236.)

2.9.7 Etäisyysensensorit

Etäisyysensoreilla mitataan etäisyyttä kappaleisiin tai kartoitetaan ympäristöä. Ne useimmiten perustuvat valoon tai ultraääneen ja niiden heijastumiseen mitattavasta kohteesta. Etäisyys on mahdollista laskea lähetetyn ja palaavan heijastuneen pulssin välillä kuluneesta ajasta. Tällöin tulee tietää nopeus, jolla pulssi kulkee ilmassa. Ultraäänellä tämä nopeus on äänennopeus ja valoa käytettäessä valonnopeus ilmassa. (Niku 2001, 238–341.)

2.9.8 Turvallisuussensorit

Turvallisuussensorien tehtävä on estää robottien aiheuttamia tapaturmia. Usein käytetään samoja sensoreita mitä muussakin robotin anturoinnissa käytetään. Turvakäytössä olevien sensorien ero normaaleihin on yleensä se, että niissä vikaantumisen ai-

heuttamat riskit on minimoitu. Laser valoverhoon tai konenäköön perustuvat ratkaisut ovat yleistymässä, sillä ne sallivat laajan alueen valvonnan yhdellä sensorilla. (Suomen robotiikkayhdistys Ry 1999, 105.)

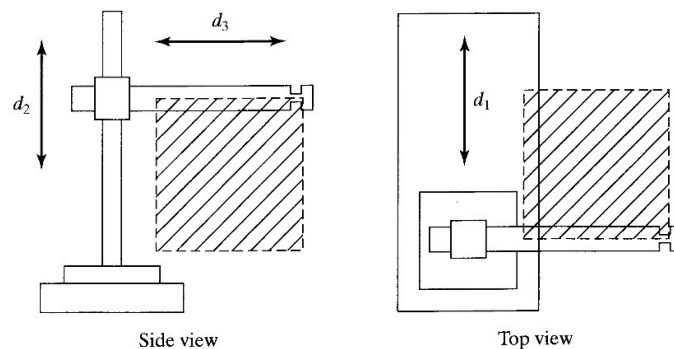
2.9.9 Konenäköjärjestelmät

Konenäköjärjestelmät ovat monipuolisimpia sensoreita mitä robotiikassa käytetään. Konenäköä käsitellään tarkemmin omassa kappaleessaan 3. (Niku 2001, 341.)

2.10 Robottityypit

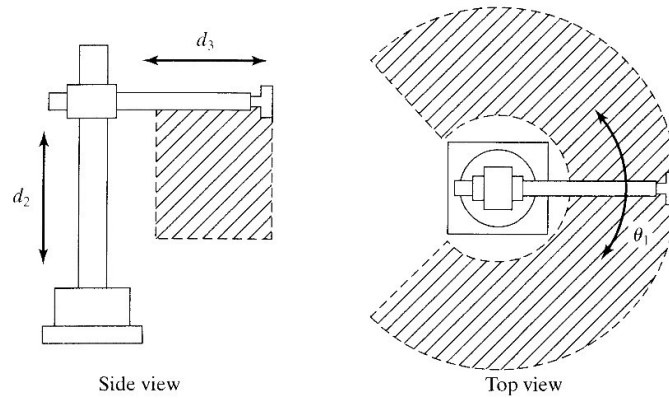
Robottien manipulaattoreissa käytetään erityyppisiä nivellyksiä kuten lineaarisia, kääntyviä, liukuvia ja palloniveliä. Yleisimpiä ovat lineaariset ja kääntyvät nivelet, kun taas pallonivelet ovat harvinaisempia. Manipulaattorit luokitellaan niiden kolmen ensimmäisen käyttämän nivellystyyppin mukaan. Manipulaattorin päässä on myös usein jonkinlainen kääntyvä ranne ja tarttuja. Alla yleisesti käytettyjä luokituksia. (Graig 2005, 234.)

Suorakulmaisesti (Cartesian) toimivissa manipulaattoreissa on kolme lineaarisesti liikkuvaa nivelvartta. Tällainen rakenne on erittäin jäykkä ja voi täten liikuttaa suuria-kin kuormia. (Graig 2005, 234.)



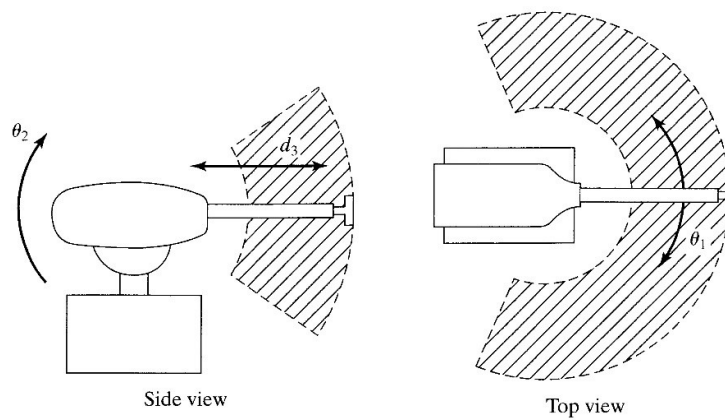
Kuvio 4 Suorakulmainen manipulaattori. (Graig 2005, 234.)

Sylinterityyppisissä (Cylindrical) manipulaattoreissa on jalustassa yksi kiertyvä ja varressa kaksi lineaarisesti liikkuvaa niveltä. (Graig 2005, 236.)



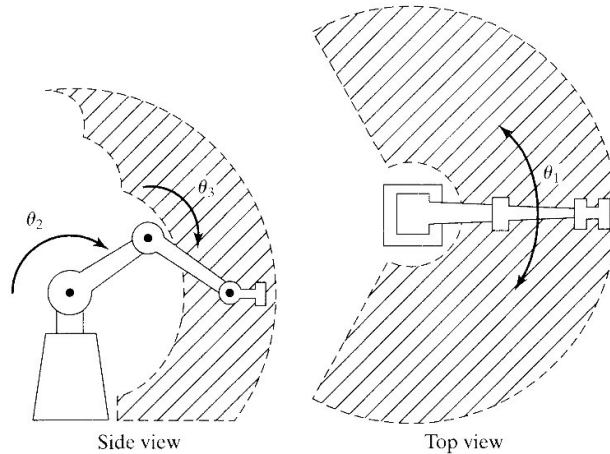
Kuvio 5 Sylinteri manipulaattori. (Graig 2005, 237.)

Napakoordinaatisto (Spherical) manipulaattoreissa on kaksi kiertyvää niveltä ja yksi lineaarinen nivel. (Graig 2005, 236.)



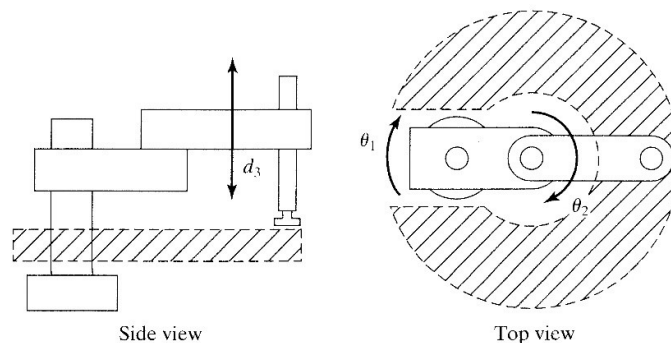
Kuvio 6 Napakoordinaatisto manipulaattori. (Graig 2005, 236.)

Kiertyvänivelisissä (Articulated) manipulaattoreissa on kolme kiertyvää niveltä ja ne muistuttavat eniten ihmisen kättä toiminnaltaan. Niiden etuihin kuuluu myös pieni tilantarve työalueelta. Ne ovat kenties myös yleisin käytetty manipulaattorityyppi. (Graig 2005, 235.)



Kuvio 7 Kiertyvänivelinen manipulaattori. (Graig 2005, 235.)

SCARA manipulaattori sisältää kaksi samassa tasossa kääntyvää niveltä ja yhden lineaarisen nivelen pystytasossa liikkumiseen. Kahden ensimmäisen nivelen ei myöskään tarvitse kantaa painoa, joten tällä nivelasetelmällä manipulaattori on erittäin nopea liikkeissään. (Graig 2005, 235.; Niku 2001, 11.)



Kuvio 8 SCARA manipulaattori. (Graig 2005, 235.)

2.11 Robottien ominaisuudet

2.11.1 Kuormankantokyky

Kuormankantokyky vaikuttaa robotin kykyyn käsitellä eripainoisia kappaleita. Robotti saattaa olla kykenevä käsittelemään painavampia kappaleita kuin sen valmistaja ilmoittaa, mutta sen ylittyessä robotti ei välttämättä enää kykene liikkumaan tarkasti ja yhtä nopeasti kuin normaalisti. (Niku 2001, 15.)

2.11.2 Robotin toimintasäde

Toimintasäde kertoo robotin maksimi työalueen. Useimpiin paikkoihin työalueella robotti kykenee saapumaan useammassa asennossa. Siirryttäessä lähemmäksi robotin toiminnan rajoja mahdolliset asennot alkavat olla rajoitetumpia. Toimintasäde riippuu robotin varsien pituuksista ja käytetyistä niveltyypeistä. (Niku 2001, 15.)

Robotin toimintakyky rajoittuu äärialueilla silloin kuin se menettää vapausasteitaan. Tämä tapahtuu silloin kun nivel on saavuttanut ääriasentonsa eikä kykene liikkumaan enempää. Tästä syystä robotin kurottaessa kauas sen kyky asetella kappaleita eri asentoihin alkaa rajoittua huomattavasti. (Niku 2001, 82.)

2.11.3 Toistotarkkuus ja Absoluuttinen tarkkuus

Toistotarkkuus koostuu robotin tilastollisesta tarkkuudesta saapua opetettuun pisteeseen useamman kerran. Robotti voidaan esimerkiksi ajaa tiettyyn pisteeseen 100 kertaa. Riippuen useista tekijöistä robotti ei välttämättä saavu joka kerta aivan absoluuttisesti samaan pisteeseen. Näistä tekijöistä riippumatta robotti aina saavuttaa kohdepisteen tietyn etäisyyden sisään, jolloin muodostuvan pistepilven säde kertoo robotin toistotarkkuuden. Robotin tarkkuus liikkeen toistettavuudessa on yleensä tärkeämpi tieto kuin sen absoluuttinen tarkkuus. (Niku 2001, 16.; Suomen robotiikkayhdistys Ry 1999, 14.)

Absoluuttinen tarkkuus kertoo robotin kyvystä saapua sen koordinaatistossa määrättyyn pisteeseen. Absoluuttinen tarkkuus on usein monia kertoja toistotarkkuutta heikompi, sillä robotin tarkkuus myös riippuu sen asennosta. Mitä kauemmaksi robotti kurkottaa sitä heikommaksi sen tarkkuus yleensä muuttuu. (Niku 2001, 16.; Suomen robotiikkayhdistys Ry 1999, 14.)

2.12 Koordinaatistot

Robotin asentoja voidaan kuvata käyttämällä erilaisia koordinaatistoja. Käytetty koordinaatisto vaikuttaa tapaan jolla robotti liikkuu. (Niku 2001, 12.)

Maailmakoordinaatisto on x y z-akselinen koordinaatisto, joka on sidottu robotin jalustaan. Robotin liike on aina samansuuntainen riippumatta mihin suuntaan sen varsi osoittaa. (Niku 2001, 13.)

Nivelkoordinaatistoa käytettäessä paikkakoordinaatit on ilmoitettu nivelten asentoina. Tällöin liikekäsken aikana robotin jokainen nivel liikkuu itsenäisesti. Robotin käyttämät niveltyypit vaikuttavat siihen millaista rataa pitkin tarttuja liikkuu kohdepisteiden välillä. (Niku 2001, 13.)

Työkalukoordinaatisto on sidottu robotin varren päähän ja sitä käytettäessä robotti liikkuu työkalun asennon mukaan. Esimerkiksi työkalun x-akselin osoittaessa alaspäin liikekäsky akselin positiiviseen suuntaan liikuttaa työkalua alaspäin sen sijaan että se liikkuisi maailmakoordinaatiston x-akselin osoittamaan suuntaan eli ylöspäin. (Niku 2001, 13.)

2.13 Ohjelmointitavat

Opettamalla ohjelmoinnissa robotin nivelet asetetaan haluttuun asentoon käsiohjainta käyttämällä. Haluttu paikka ja asento tallennetaan robotin muistiin. Paikkapisteitä voidaan jälkeinpäin kutsua ohjelmassa, jolloin robotti liikkuu opittujen pisteiden välillä. (Niku 2001, 13.)

Ennen tietokoneiden yleistymistä kaikki varhaiset robotit ohjelmointiin käyttäen tätä metodologiaa. Robottien liikkeet olivat lähinnä yksinkertaisia kokoonpano sekvenssejä. Modernien robottien ohjelmointi keskittyy tietokoneella tehtyyn työhön. Näiden robottien käsiohjaimet lähestyvät ominaisuuksiltaan tietokoneterminaalia. (Graig 2005, 340.)

Johdattamalla ohjelmoinnissa operaattori liikuttaa robotin vartta käsin haluttua liikerataa pitkin. Samalla robotin nivelten asennot tallennetaan muistiin tasaisin väliajoin. Tallennettu liikerata voidaan siten halutessa toistaa. (Niku 2001, 14.)

Robottien ohjelmointikielet keskittyvät lähinnä niiden ongelmien ratkaisuihin mitä robottien ohjelmointiin liittyy. Tietokoneiden yleistyttyä robottien ohjelmointi siirtyi enemmän tietokoneilla tehdyksi. Lähes aina robotti voidaan kuitenkin edelleen ohjel-

moida vain käyttämällä robotin käsiohjainta. Tällöin ei vain voida käyttää ohjelmistojen tarjoamia kehittyneempiä menetelmiä. Esimerkkejä vain roboteille tarkoitetuista ohjelmointikielistä ovat ABB Robotics:in RAPID, IBM:n AML ja GMF Robotics:in KAREL. Robotin paikkaa, asentoa ja työskentelykoordinaatistoa kuvaavat tietotyypit ovat robottien ohjelmointikielille hyvin tyypillisiä. Niitä ei esiinny lainkaan tavanomaisissa ohjelmointikielissä. (Graig 2005, 341–342.)

2.14 ABB robotin liike

ABB robotin liikekäskyt koostuvat halutusta liiketyypistä, kohdepisteestä, liikenopeudesta, paikoitustiedosta, käytetystä työkalusta sekä koordinaatistosta jonka mukaan liike tehdään. (ABB Robotics 2011, 25.)

2.14.1 Liiketyypit

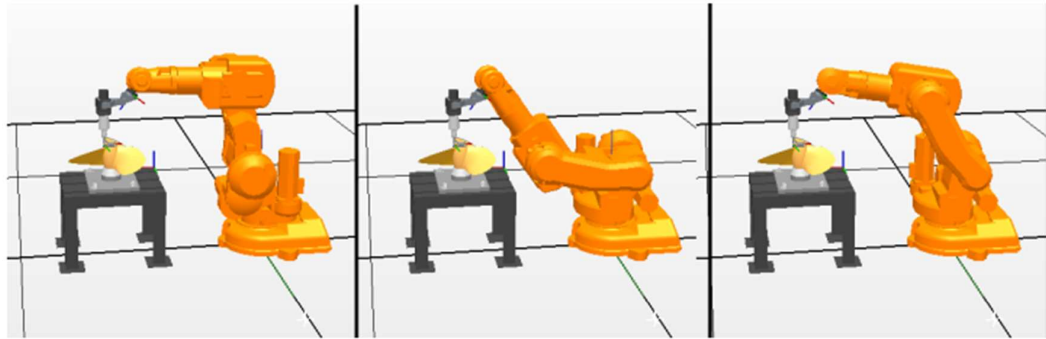
Lineaariliikkettä käytetään, kun robotin työkalun keskipisteen halutaan liikkuvan lineaarisesti paikkapisteiden välillä. (ABB Robotics 2010, 264.)

Nivelliikeettä käytettäessä robotti liikkuu nopeasti haluttuun asentoon epälineaarista reittiä pitkin. Kaikki robotin nivelet saapuvat kohdeasentoon samanaikaisesti. Nivelliikettä kannattaa käyttää, kun tarvetta lineaariselle liikkeelle ei ole. (ABB Robotics 2010, 253.)

Ympyräliikkeessä robotti liikuttaa työkalua ympyräliikkeessä paikkapisteiden välillä. Komennossa tulee antaa kohdepisteen lisäksi välipiste, jonka kautta ympyrän kaari kulkee. (ABB Robotics 2010, 236.)

2.14.2 Paikkapisteet

Paikkapisteet on määritetty sijainteina halutussa koordinaatistossa, sekä orientaationa johon robotti asettaa työkalunsa. Robotin kontrollerin laskiessa tarvittavia nivelkulmia kullekin paikkapisteelle saadaan usein ratkaisuksi useita mahdollisia asentoja. Eri vaihtoehtojen erottamiseksi toisistaan on kaikissa paikkapisteissä ilmoitettu robotin nivelille kvadrantti eli neljännes joissa niiden tulisi sijaita. (ABB Robotics 2011, 25–32.)



Kuvio 9 Mahdollisia eri asentoja samalle paikkapisteelle. (ABB Robotics 2011, 32.)

2.14.3 Liikenopeus

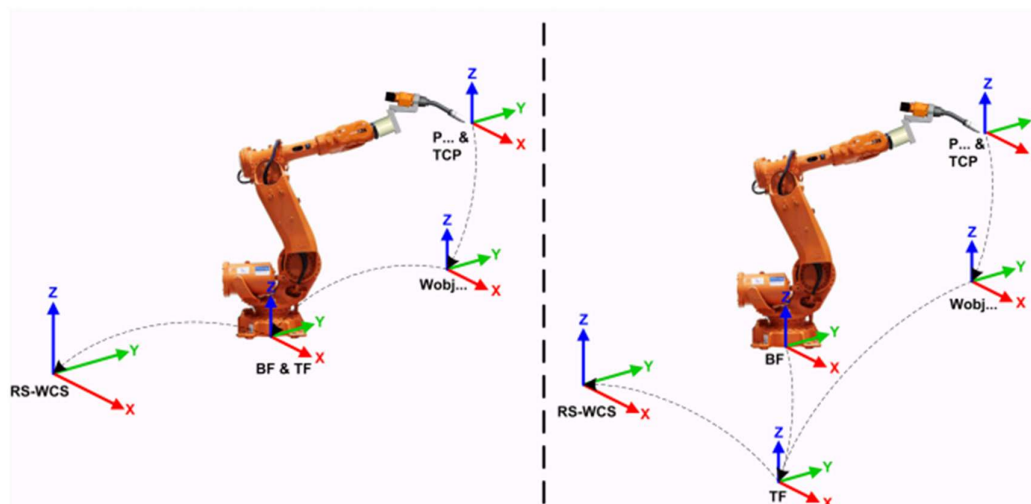
Määritetty liikenopeus vaikuttaa nopeuteen, jolla robotti liikuttaa työkalua. Nopeus voidaan määrittää työkalun liikenopeutena tai nivelten kääntymisnopeutena. Ohjelmisto sisältää esimääritettyjä nopeustietoja, joita voidaan käyttää itsemäärittämisen sijaan. (ABB Robotics 2010, 1185–1186.)

2.14.4 Paikoitustarkkuus

Liikekäslyn paikoitustarkkuus vaikuttaa siihen kuinka tarkasti robotin tulee kulkea kohdepisteen kautta ennen kuin se jatkaa seuraavaan pisteeseen. Käyttämällä fine paikoituskomentoa robotti kulkee tarkasti kohdepisteeseen. Jos ei ole tarvetta kulkea tarkasti pisteen kautta voidaan ilmoittaa pisteelle säde, jonka kautta robotin tulee kulkea. Tällöin robotti voi oikaista ja liike on sulavampi. (ABB Robotics 2010, 1232.)

2.14.5 Koordinaatistot ja niiden hierarkia

ABB robotin käyttämät koordinaatistot ovat kaikki hierarkkisessa suhteessa toisiinsa perustuen maailmakoordinaatistoon. Jokainen koordinaatisto siis sijaitsee jonkun toisen koordinaatiston sisällä. (ABB Robotics 2011, 26.)



Kuvio 10 Koordinaatistojen hierarkia. (ABB Robotics 2011, 27.)

RS-WCS on maailmakoordinaatisto, jonka sisään kuuluu koko robottisolu, vaikka siihen kuuluisi useampikin robotti. Kaikki muut koordinaatistot perustuvat tähän koordinaatistoon. (ABB Robotics 2011, 26.)

Base Frame (BF) koordinaatisto on aina liitetty robotin jalustaan. Kullakin robotilla on oma BF. (ABB Robotics 2011, 26.)

Task Frame (TF) koordinaatisto kuvaa robotin kontrollerin käyttämää maailmakoordinaatistoa jonka sisällä työ suoritetaan. Sijaitsee useimmiten samassa sijainnissa, kun BF. (ABB Robotics 2011, 26.)

Workobject (Wobj) kuvaa robotilla työstettävän kappaleen koordinaatisto. Jos sitä ei ole määritetty robotti käyttää vakiota joka on sama kuin BF. Workobjekteja voidaan käyttää koordinoitussa liikkeessä siten että robotti löytää kohdepisteensä käsiteltävässä kappaleessa, vaikka se olisi liitetty toiseen robottiin. (ABB Robotics 2011,31.)

Tool Center Point Coordinate (TCP) eli työkalun keskipiste, jota robotti siirtelee ohjelmoituihin pisteisiin. (ABB Robotics 2011, 26.)

P on robotin paikoitus käyttäjän valitsemassa Wobj koordinaatistossa. (ABB Robotics 2011, 26.)

3 Konenäkö

Konenäkö perustuu kameroihin jotka ovat silmän toimintaa imitoivia laitteita. Niiden toiminnan kannalta keskeisiä asioita ovat valo, linssi ja kuvakenno. Kuva muodostuu, kun tarkasteltavasta kohteesta heijastunut valo kulkee kameran linssin läpi ja muodostaa kuvattavasta kohteesta projektion kuvakennolle. Konenäön etuihin voidaan lukea sen väsymättömyys, nopeus ja tasalaatuisuus. (Cognex 2016, 5; Matarić 2007, 109.)

Teollisuudessa käytetään 1D-, 2D- ja 3D-kameroihin. 1D-kamerat ovat viivakameroita, jotka kuvaavat vain yhden viivan kerrallaan. Viivakamerat soveltuvat paikkoihin joissa tarkkailtava tuote on jatkuvassa liikkeessä, kuten esimerkiksi paperiteollisuudessa. Niitä voidaan myös käyttää sylinterimäisten kappaleiden tarkastamiseen pyöräyttämällä tarkasteltava kappale kameran edessä. 2D-kamerat ovat kaikista konenäkökameroista eniten käytettyjä. Ne ottavat perinteisiä kaksiulotteisia kuvia. 3D-kamerajärjestelmät sisältävät useimmiten useamman kameran tai laser-etäisyysensorin, jotta niillä saadaan muodostettua 3D-kuva. 3D-kameroilla voidaan tarkemmin syöttää esimerkiksi roboteille tarkkaa tietoa työstettävien kappaleiden asennoista tai niillä voidaan tarkastella kappaleiden 3D-profiileja virheiden varalta. (Cognex 2016, 17–20.)

3.1 Käyttökohteet

Konenäöllä tehtävät työt voidaan jakaa ohjaukseen, tunnistukseen, mittaukseen ja tarkastukseen. Konenäkökameroita valmistava Cognex ehdottaa konenäön käyttökohteiksi mm. seuraavia. (Cognex 2016, 7-10.)

- Osien asennon selvitys robotilla käsittelyä varten.
- Tuotteiden tarkistus virheiden varalta.
- Tuotteiden viivakoodien tunnistus ja etikettien tekstien tarkistus.
- Osien tunnistus niiden värin, muodon tai koon avulla.
- Tuotteiden laaduntarkkailu mittaamalla mikroskooppisia etäisyyksiä esimerkiksi auton syytystulppien kärkivälin tarkistuksessa.
- Elintarvikepakkausten eheysvalvonta.

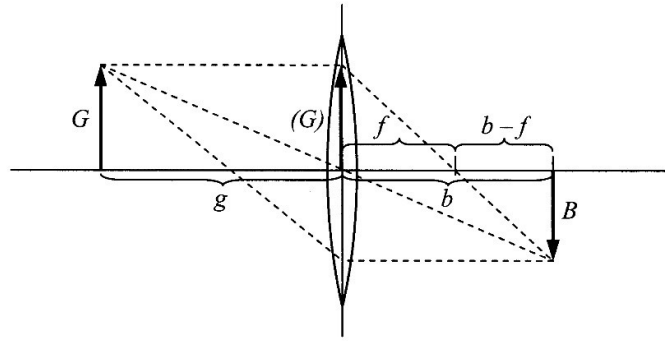
3.2 Kuvakenno

Kuvakenno on piistä valmistettu levy, jossa on tuhansia valoherkkiä pisteitä. Kennot perustuvat CCD (charge coupled device) tai CMOS (complementary metal oxide semiconductor) tekniikoihin. Yksi piste muodostaa pikselin, joka riippuen valon määrästä varautuvat sähköisesti. Luettaessa varukset muunnetaan digitaaliseen muotoon. Yksittäisistä pikseleistä yhteen kokoamalla saadaan kokonainen kuva. Esimerkkinä 800 × 600 resoluution kamerassa on 600 kertaa 800 pisteen kuvapisteriviä. Kyseisessä kennossa olisi siis kuvapisteitä yhteensä 480 000. Vertailun vuoksi ihmissilmä sisältää noin 126 miljoonaa valoherkkää solua. (Matarić 2007, 109.; Niku 2001, 14.)

Jotta kameralla voitaisiin ottaa mustavalkokuvien sijaan värikuvia, on käytettävä monimutkaisempia keinoja. Ensimmäisenä vaihtoehtona voidaan kullekin värille käyttää omaa värifilterillä varustettua kuvakennoa. Näin saadaan hyvin korkealaatuisia kuvia mutta kamerasta tulee tällöin hyvin kallis. Toisena vaihtoehtona voidaan käyttää Foveonin periaatetta jossa valoherkät pikselit ovat kolmella eri syvyydellä piilevyyssä. Pikselit reagoivat tällöin valon eri aallonpituuksiin niiden syvyyden mukaan. Kolmantena vaihtoehtona on Bayer kuvio, jossa kunkin pikselin päälle on asetettu värisuodin. Yhdistelemällä viereisten pikselien tietoja interpoloimalla voidaan koota kokonainen väripikseli. (Azad 2008, 46.)

3.3 Optiikka

Polttoväli (focal distance) valitaan riippuen kameran kennon ja kuvattavan alueen koosta, sekä etäisyydestä kameran linssistä. Tyypillisesti myydään tietyillä polttoväleillä varustettuja linsejä, joissa $f = 4.2, 6, 8, 9, 12, 16, 25, 35, 50, \dots$ [mm]. Oikealla linssin polttovälin valinnalla varmistetaan siitä, että kuvattava kohde mahtuu kokonaan kuvaan. (Azad 2008, 29–31.)



Kuvio 11 Graafinen esitys kameran optiikasta. (Azad 2008, 31.)

Kameran optiikassa eri arvot ovat suhteessa toisiinsa seuraavan kaavan kautta.

$$f = \frac{g \cdot B}{G + B}$$

Kuviossa 11 ja kaavassa f = polttoväli, g = kuvattavan alueen etäisyys kameran linsistä, G = kuvattavan alueen koko ja B = kameran kennon koko. (Azad 2008, 30.)

Aukko (aperture) voi olla kiinteä tai säädettävissä. Aukon suurus kuvaa kuinka paljon valoa linssi päästää kennolle. Aukko siis kuvaa linssin tehokkuutta kerätä valoa. Aukon kaava on $k = f/d$, jossa f = polttoväli ja d = linssin halkaisija. Pienempi arvo on siis parempi. Säädettävä aukko voi myös olla moottoriohjattu. (Azad 2008, 31.)

Tarkennus (focus) tapahtuu muuttamalla linssin ja kuvakennon välistä etäisyyttä. Tarkennettaessa kaukaiseen kohteeseen kennon etäisyys linsistä b lähestyy linssin polttoväliä f . Tarkennus voidaan aukonsäädön tavoin automatisoida ohjatulla moottorilla. (Azad 2008, 32.)

Resoluutio (resolution) kertoo kuinka tarkkaa kuvaa kamera tuottaa. Resoluutio vaikuttaa täten myös kameran mittaustarkkuuteen. Seuraavalla kaavalla voidaan laskea, mitä pituutta kuvassa yksi kameran pikseli vastaa.

$$R(\text{spatiaalinen}) = G \cdot \frac{1}{R(\text{kamera})}$$

Kaavassa G = kuva-alueen koko ja R = resoluutio. (Azad 2008, 36.)

3.4 Valonlähteet

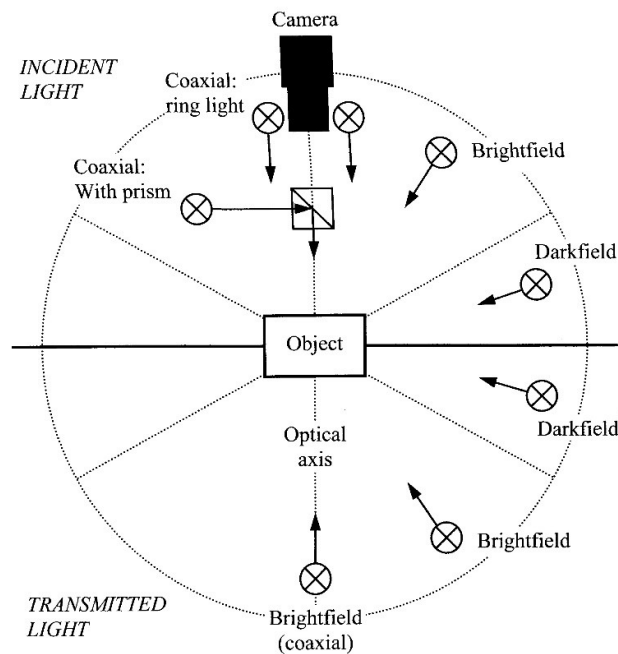
Konenäköjärjestelmät tarvitsevat valonlähteen. Kameran tarvitsema valonlähteinä voi toimia hehkulamppu, halogeenilamppu, kaasupurkauslamppu, neonlamppu, led, laserdiodi tai auringonvalo. Kaikilla näistä on omat hyvät ja huonot puolensa. Kokonaisuutena näistä parhaimmat ominaisuudet ovat led valoilla. Ne ovat pieniä, kestävät tuhansia tunteja, sekä niiden hyötysuhde on hyvä mutta haittapuolena niiden valoteho on heikohko. Tarvittaessa paljon valotehoa kaasupurkauslamppu on paras valinta. Kuviossa 12 on taulukkona eri valonlähteiden ominaisuuksia. (Azad 2008, 18.)

Characteristics	Size	Cost, relating to Φ_v	Efficacy η_v , approx.	Maximum luminous flux	Suitability for diffuse illumination	Suitability for directed illumination	Suitability for usage with lenses	Usage as strobe, with sync	Aging effects	Operating hours approx..	Comments
Light source											
Spiral-wound filament	0	++	1.9-2.6 %	-	-	0	0	-	-	1 000	
Halogen	+	+	2.3-5.1 %	+	-	0	+	-	-	3 000	Great heat generation, therefore frequently used with fiber optics.
Gas discharge (HLL, HSL ...)	+	-	15-27 %	++	-	0	+	+	-	6 000	Great heat generation, therefore frequently used with fiber optics.
Neon, luminescent material	-	+	6.6-15 %	+	++	--	--	-	+	7 500	Almost solely used with high frequency PSU.
Light emitting diode	++	0	5-20 %	-	0	++	++	++	++	50 000	Different colors, also available as IR and UV, small size (\rightarrow array configurations).
Laser diode	+	-	7-12 %	-	--	++	++	++	0	10 000	Usage as "structured light".
Daylight	++	++	n/a	\pm Weather? Daytime?	++	0 with lens	+	with mechanical shutter	n/a	++	

Kuvio 12 Eri valonlähteiden vertailutaulukko. (Azad 2008, 19.)

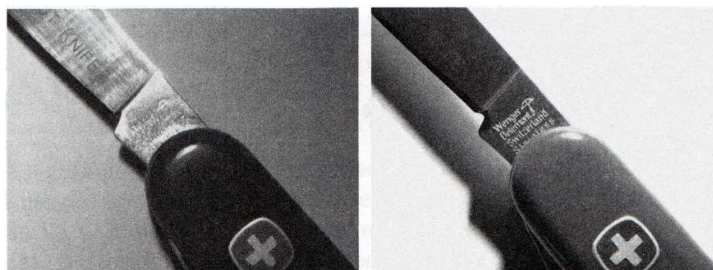
3.5 Valaistustekniikat

Valon sijainnilla suhteessa kameraan voidaan vaikuttaa siihen, millainen kuva kohteesta välittyy kameralle. Valo voidaan asettaa eteen, taakse tai sivulle. Sijoittelulla valitaan sen mukaan mitä kohteesta halutaan nähdä. Kohtisuorassa valaistuksessa (brightfield) valo heijastuu kohteesta suoraan kameraan. Tällöin myös tasaiset pinnat näkyvät hyvin. Jos tarkoituksena on tarkastella kohteen siluettia niin siinä tapauksessa valo kannattaa asettaa kohteen taakse. (Azad 2008, 20.; Cognex 2016b, 5-6)



Kuvio 13 Eri valaisusuuntia. (Azad 2008, 21.)

Sivusta tuleva valo eli pimeäkenttä (darkfield) taas tuo hyvin esiin pinnan syvyysvaihtelut. Kuviossa 14 näkyy kohtisuoran ja pimeäkenttä valaistuksen ero. Jos valo kulkee samansuuntaisesti kameras kanssa, sitä kutsutaan myös aksiaaliseksi (coaxial). (Azad 2008, 21.; Cognex 2016, 13–14.)



Kuvio 14 Vasemmalla kohtisuoraan ja oikealla sivulta tuleva valaistus. (Azad 2008, 21.)

Valodiffusoreja käytetään, jos kuvattava kohde on hyvin heijastava. Tällöin voidaan esimerkiksi käyttää Ulbricht kupolia, jonka sisäpinta on päällystetty hyvin heijastavalla materiaalilla. Usein voidaan myös käyttää kameran linssin ympärillä olevaa diffusoidua pyöreää led valoa. Diffusoidulla valolla vältetään varjojen muodostumiselta ja valaisu on hyvin tasainen. (Azad 2008, 22–23.)

Laservaloa käytettäessä voidaan luoda viiva tai kuvio esineen pintaan, jolloin kuvion vääristymästä voidaan päätellä kappaleen fyysisiä mittoja. (Azad 2008, 23.; Cognex 2016, 13)

Kameran erottelukykyä voidaan myös tarvittaessa parantaa käyttämällä valosuodattimia, jolloin kaikki muut kuin suodattimen läpi päästämät värisävyt näyttävät tummilta. Kuviossa 15 on käytössä punainen valosuodatin, jolloin kamera ei näe beigen väristä tekstiä. (Azad 2008, 23.)



Kuvio 15 Vasemmalla kuva ilman värisuodatusta ja oikealla punaisen suodattimen kanssa. (Azad 2008, 24.)

Valon polarisointisuodattimia soveltuvat vähentämään valon heijastumista aiheuttavaa häiriötä. Tällöin suodatin asennetaan kameran linssin eteen ja valonlähteeseen asetetaan toinen suodin, mutta 90-astetta eri asentoon käännettynä. Polarisoituilla suodattimilla saadaan erinomaisia tuloksia muovien ja sellofaanien kanssa. Polarisoitimetodi ei kuitenkaan toimi metallien kanssa. (Azad 2008, 24.)



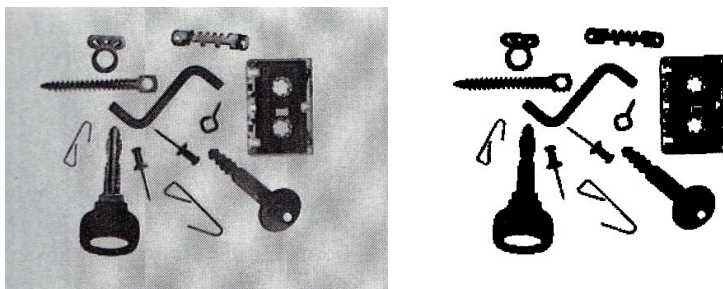
Kuvio 16 Polarisaatio-suodattimen vaikutus heijastuksiin. (Azad 2008, 24.)

Konenäössä on myös otettava huomioon ylimääräiset valot, jotka usein häiritsevät mittauksia. Auringonvalo ja kattolamput voivat pahimmillaan täysin estää konenäköä toimimasta. Ongelma voidaan kiertää käyttämällä valotiiviitä kotelaita tai käyttämällä spektrifilttereitä, jotka estävät näkyvän valon pääsyn kameraan. Kameroissa käytetyt kuvakennot reagoivat myös ihmissilmälle näkymättömiin IR- ja UV-aallonpituuksiin. (Azad 2008, 24–25.)

3.6 Konenäön perustoimintoja

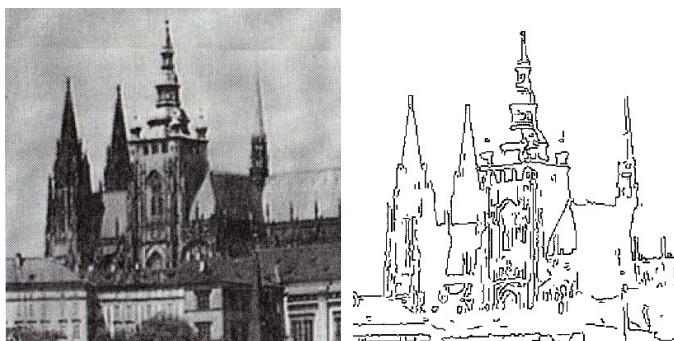
Kuvan segmentaatio osiin on konenäössä yksi tärkeimpiä työkaluja. Segmentaatiossa tarkoituksena on tuoda esiin kappaleita, joita halutaan mitata. Useimmiten tämä tarkoittaa esineiden erittelyä tasavärisestä taustasta kontrastierojen avulla. Alla yleisimpiä käytettyjä segmentaatiotapoja. (Sonka 2008, 175.)

Kynnistäminen (thresholding) on yksinkertainen tapa, jolla harmaasävykuva saadaan segmentoitua. Erottelu tapahtuu syöttämällä tarkoitukseen sopiva harmaasävyarvo, jonka mukaan kuva jaetaan valkoiseen taustaan ja mustiin alueisiin jotka kuvaavat havaittavia kappaleita. Jakavan arvon oikea valinta on tärkeää kynnistyksen toiminnalle, sillä väärin määritetty arvo aiheuttaa virheellisiä havaintoja. Kynnystämässä on myös usein tärkeää, että havaittavat kappaleet eivät kosketa toisiaan. Laskennallisesti kynnistäminen on erittäin kevyt toiminto. (Sonka 2008, 176.)



Kuvio 17 Kuvan kynnystys. (Sonka 2008, 177.)

Reunantunnistus (edge recognition) on yksi yleisimpiä konenäössä käytettyjä metodeja. Reunaksi kuvassa määritellään paikka kuvassa, jossa pikselien kirkkaus muuttuu huomattavasti viereisiin pikseleihin nähden. Kameran kuvassa kuitenkin yleensä esiintyy jonkin verran kohinaa joka aiheuttaa virheellisiä havaintoja. Kohinaa voidaan vähentää käyttämällä kohinansuodatusta. (Matarić 2007, 110.)



Kuvio 18 Kuvan reunantunnistus. (Sonka 2008, 186.)

Hahmontunnistus (pattern recognition) on metodi, jossa kuvasta etsitään ennalta määrättyä kuviota. Yleensä yhdestä kuvasta leikataan haettava kohde malliksi. Mallia vastaavaa kuviota voidaan siten hakea muista kuvista. Haku tapahtuu useimmiten vertailemalla harmaasävyarvoja. (Sonka 2008, 238.)

3.7 Tiedonsiirto

Analoginen tiedonsiirto on vanha tekniikka, joka kuitenkin on vielä käytössä joissain järjestelmissä. Näissä tapauksissa järjestelmän päivitys digitaaliseksi tulisi useimmiten maksamaan huomattavasti, joten päivitys tehdään vasta kun se on ehdottoman välttämätöntä. Analogiset järjestelmät ovat halpoja ja yksinkertaisia mutta niiden tarkkuus ei pärjää digitaalisille järjestelmille. (Azad 2008, 51.)

USB eli Universal Serial Bus on lähinnä käytössä web-kameroissa, mutta joidenkin valmistajien konenäkökameroissa on tämä liitännämahdollisuus. (Azad 2008, 51.)

Firewire tai vaihtoehtoisesti IEEE 1394 on laajalle levinnyt tiedonsiirto standardi. Firewiren kanssa voidaan käyttää pitkiä kaapeleita ja laitteita on mahdollista ketjuttaa. (Azad 2008, 52.)

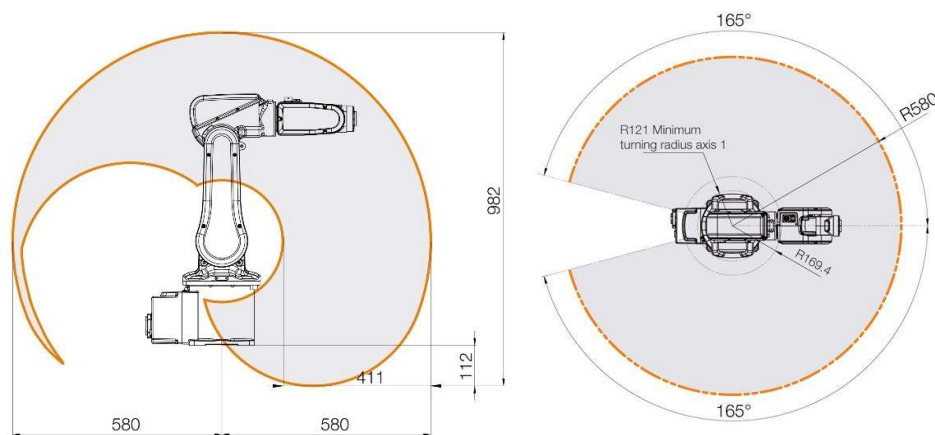
Ethernet yhteyden edut ovat halpa hinta ja standardisoidut liittimet. Tietoa voidaan myös siirtää suuria etäisyyksiä ja laitteiden määrä samassa verkossa ei ole rajoitettu. Ethernetin tapauksessa verkossa liittyvän datan määrää on rajoitettava, ettei verkko tukkeudu. (Azad 2008, 53.)

Suora I/O on nopea ja helppo tekniikka, jossa kamera kytketään suoraan kiinni ohjelmoitavaan logiikkaan käyttämällä sen digitaalisia ulostuloja. (Cognex 2016, 16.)

4 Käytetyt laitteistot ja ohjelmistot

4.1 ABB IRB-120 teollisuusrobotti

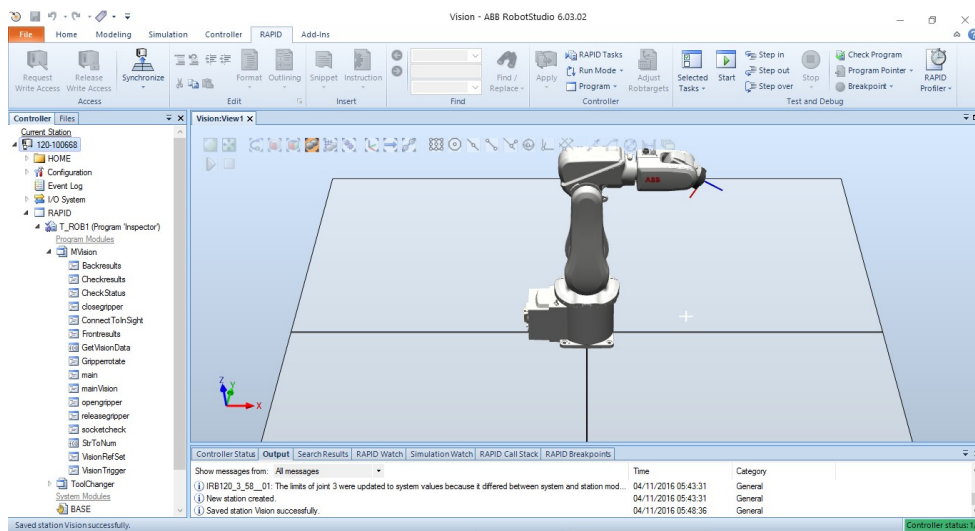
IRB 120 on ABB:n pienin valmistama teollisuusrobotti. Painoa robotilla on vain 25 kg ja se voidaan asentaa mihin asentoon tahansa. Robotissa on kuusi kääntyvää niveltä ja sen maksimikuorma on 3 kg. IRB 120 kykenee toimimaan 580 mm etäisyydellä ja -lulistastaan 0.01 mm toistotarkkuudella. Robotista on myös saatavilla IRB 120T variantti, jossa akselit 4, 5 ja 6 ovat huomattavasti nopeampia. (ABB Robotics 2012.)



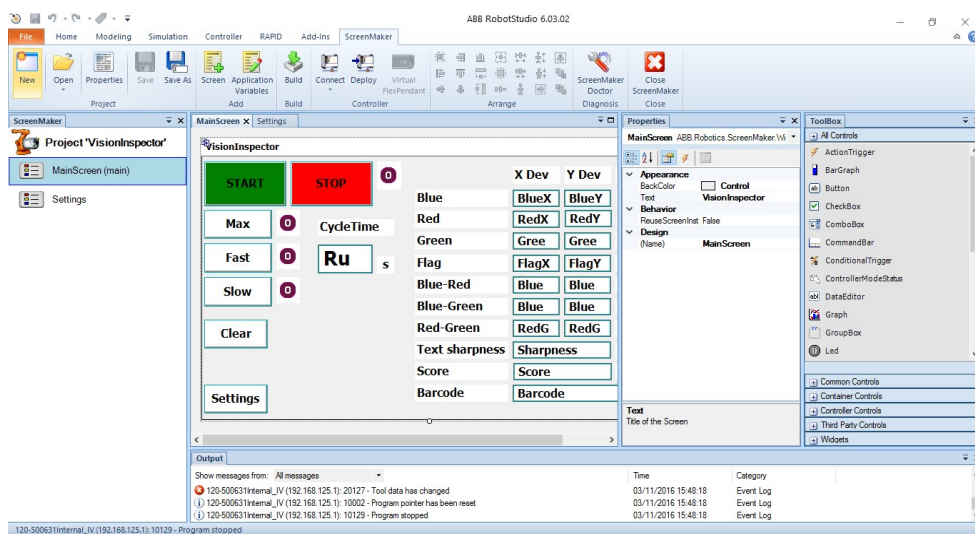
Kuvio 19 IRB 120 robotin toiminta-alue. (ABB Robotics 2012.)

4.2 Robotstudio

Robotin ohjelmointiin käytettiin ABB Robotstudio- ohjelmistoa. Robotin ohjaimessa näkyvään käyttöliittymään käytettiin Robotstudion Screenmaker lisäosaa. Robotstudiossa on myös mahdollista simuloida robottien toimintaa laajasti. Työstettävistä kappaleista voidaan tuoda 3D-mallit ohjelmaan, jossa voidaan hyvin helposti luoda robotille kappaleen reunoja seurailevat liikeradat. Työskentelyä voi tehdä tarpeen mukaan online- ja offline-tiloissa. Kuviossa 20 näkymä, jossa käytetty robotti esiintyy simuloituna ohjelmointiin. Kuviossa 21 esiintyy ohjelmointinäkymä Screenmakeriin jossa käyttöliittymä luotiin.



Kuvio 20 Simulaatio käytetystä robotista RobotStudiassa.



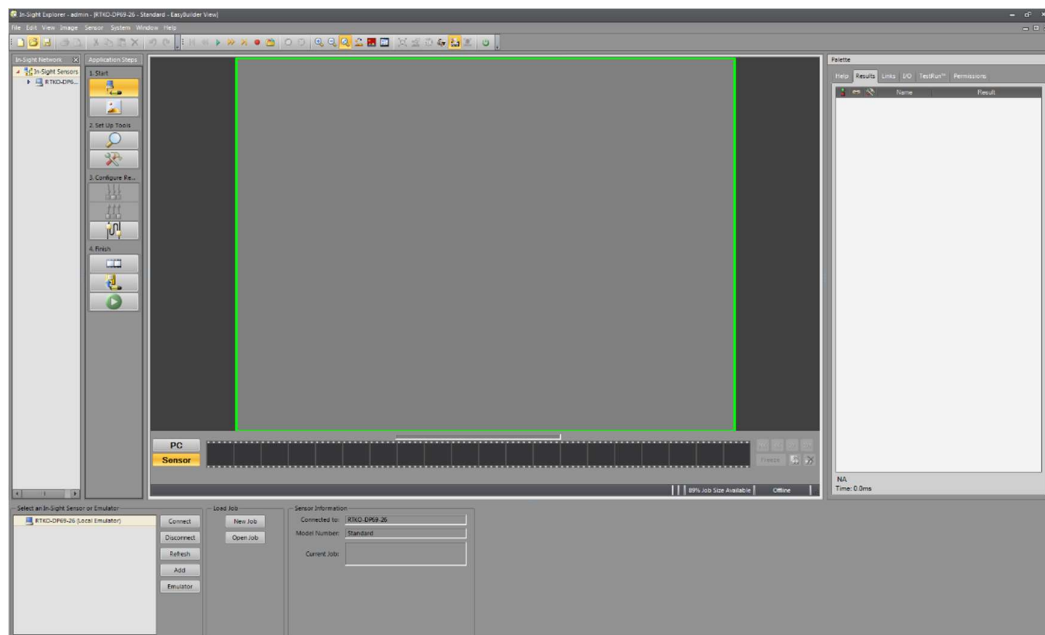
Kuvio 21 Screenmaker lisäosa.

4.3 Cognex 7200 kamera

Cognex 7200 on mustavalko, 800 x 600 resoluutiolla, automaattitarkennuksella ja integroidulla valolla varustettu konenäkökamera. Kamera kykenee myös suoraan ohjaamaan ulkoisia lisävaloja. Maksimi kuvanopeus kamerassa on 100kuvaa sekunnissa. Tuettuna on Easybuilder ja spreadsheet ohjelmointitilat. (Cognex 2013. 1-2.)

4.4 In-sight Explorer

Kameran ohjelmointityökalu oli In-sight Explorer, jossa konenäköohjelma luodaan käyttämällä Easybuilder tai spreadsheet ohjelmointitilaa. Easybuilderilla (Kuvio 22) työskentely on nopeaa, kun taas spreadsheet tilassa (Kuvio 23) ohjelmoijalla on tarkempi kontrolli ohjelman toiminnasta. Ohjelma sisältää laajan valikoiman erilaisia työkaluja kameran ottamien kuvien analysoimiseksi. Ohjelmalla on myös mahdollista tehdä konenäköprojekteja offline-tilassa ilman kameraa. Offline-tilassa työskentely tapahtuu simuloidusti tarkastelutilanteesta esitallennettujen kuvien parissa.



Kuvio 22 Easybuilder näkymä.

The screenshot shows a spreadsheet application window titled 'In-Sight Explorer - Admin - PTKG-0969-26 - Standard - Spreadsheet View'. The spreadsheet contains data for a sensor configuration. The columns are labeled A through S, and the rows are numbered 1 through 42. The data includes various numerical values and text labels such as 'Trigger', 'Exposure Control', and 'Light Control'. A 'Palette' is visible on the right side of the window, listing various functions and tools like 'Vision Tools', 'Image', 'Edge', 'Flaw Detection', 'Histogram', 'ID', 'Image', 'Superimpose', 'ROI OC/OCR', 'Pattern Match', 'Geometry', 'Mathematics', 'Text', 'Coordinate Transforms', 'Input/Output', 'Closed Data Storage', 'Vision Data Access', and 'Structure'.

Row	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	0.000	0.000	0.000	1.000	0.000														Overall
2	Trigger	Trigger Delay	Trigger Interval (ms)	Exposure (ms)	Auto-Expos	Max Expos	Target Brightness												1.00
3	0.000	0.000	0.000	0.000	0.000	0.000	0.000												1.00
4	1.000	0.000	1.000	1.000	0.000														0.000
5	0.000	0.000	0.000	0.000	0.000														0.000
6	488	Exposure Control	None	1.000	0.000														0.000
7	1.000	1.000	1.000	1.000	1.000														1.000
8	1.000	1.000	1.000	1.000	1.000														1.000
9	88.000	100.000	320.000	440.000	0.000														0.000
10	88.000	100.000	320.000	440.000	0.000														0.000
11					0.750	End													0.750

Kuvio 23 Spreadsheet näkymä.

5 Tavoitteet

Työn tarkoituksena oli toteuttaa koulun laitteistoilla toimiva konseptin todistava demo tuotepakkausten tarkistuksessa. Työn pohjalta mahdollisesti toteutetaan Vision Systemsin asiakkaan tiloihin lopullinen järjestelmä.

Asiakas oli toivonut, että työssä toteutettu järjestelmä kykenisi tarkistamaan jokaisen tuotepakkauksen. Tällöin vaadittava nopeus olisi noin yksi pakkaus sekunnissa. Asiakkaan mukaan linjalla käsitellään viittä erilaista tuotepurkkia. Eri värejä on useita. Ohjelmaan tulisi olla helppo lisätä lisää testattavia tuotteita. Järjestelmän tulisi kyetä tunnistamaan vialliset tuotteet ja tilanteet jolloin etikettitulostuslaite tarvitsee uudelleenviirittämistä operaattorin toimesta. Tarkistuksen tuli siis olla nopea sekä varmatoiminen. Pakkauksen etikettien värikerrosten tulostuminen väriin paikkoihin johtui oletettavasti siitä, että pakkaus pääsee liukumaan painokoneen otteessa. Kuviossa 24 on esimerkki viallisesta etupuolen etiketistä, jossa vihreä ja punainen värikerros on useita millimetrejä väärässä paikassa.



Kuvio 24 Virheellinen etiketti pullossa.

6 Toteutus

Työ suoritettiin Jyväskylän ammattikorkeakoulun automaatiolaboratoriotiloissa käyttäen ABB IRB-120 teollisuusrobottia ja Cognex 7200-sarjan mustavalkokameraa. Laitteistojen ohjelmointiin käytettiin koulun lisenssejä Robotstudio ja In-sight ohjelmistoihin. Työskentelyssä käytettiin vesiputousmallia, jossa edettiin askel kerrallaan ja tarpeen tullen palattiin askel taaksepäin. Käytettyjä askeleita olivat:

1. Vaatimusten määrittely
2. Suunnittelu
3. Implementointi
4. Testaus

Kehitystyö jakautui karkeasti kameran, robotin ja käyttöliittymän ohjelmointiin. Pääasiallisesti ohjelmointi tapahtui Robotstudioissa käyttäen RAPID ohjelmointikieltä. Laitteet laitettiin kommunikoimaan keskenään ethernet verkon yli. Käyttöliittymä päätettiin laittaa teollisuusrobotin ohjaimeen, jolloin erillistä tietokonetta ja näyttöä tarvittiin laitteiston käyttämiseen.

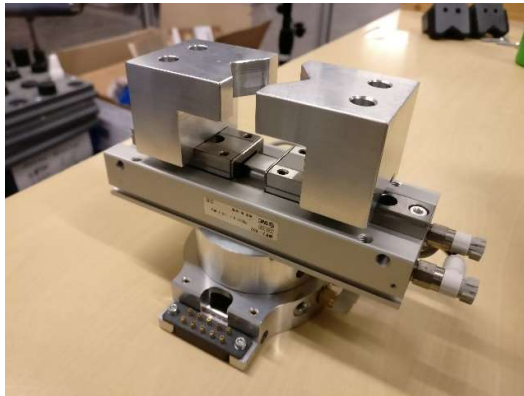
Työn toteutus käsitti konenäköohjelman teon kameralle, robotin ohjelman, laitteiston suunnittelun sekä tarttujan rakentamisen. Työ oli voimakkaasti ohjelmointipainotteinen. Eniten aikaa suunnittelussa meni laitteiden manuaalien lukemiseen ja robotin RAPID ohjelmointikielen sisäistämiseen. Keskeisiä haasteita toimivien ratkaisujen löytäminen, robotin ja kameran välisen rajapinnan luonti sekä luotettavan mittauksen saavuttaminen.

Kuviossa 25 on demosolun ensimmäinen versio siinä kokoonpanossa missä testit lopulta suoritettiin. Oikealla ABB IRB-120 teollisuusrobotti. Vasemmalla mustakeltainen Cognex 7200 kamera. Kameran takana sijaitsee diffuusiokupolilla varustettu punainen ledivalo kameran ottaman kuvan kontrastin parantamiseksi. Etuoikealla on neljä tuotepakkausta valmiina testattavaksi. Robotin mukana tullut tarttuja ei soveltunut pulloihin tarttumiseen kovin hyvin, joten tarttujan leuat korvattiin muovista sahatuilla uusilla leuoilla.

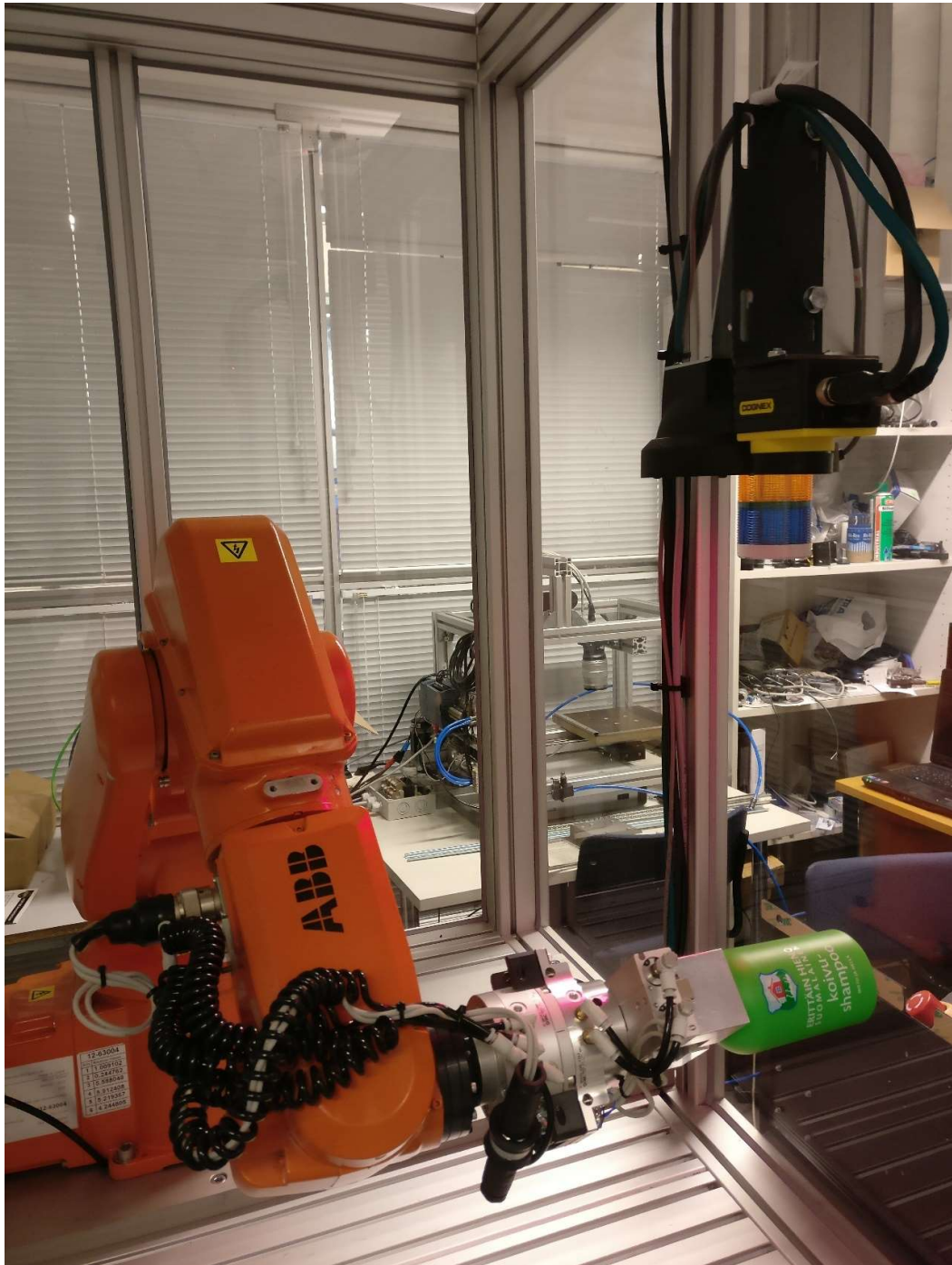


Kuvio 25 Demosolu ensimmäisessä kokoonpanossa.

Robottisolusta toteutettiin Vision Systemsin toiveesta messuesittelyä varten toinen versio, joka sijoitettiin ABB:n toimittamaan suojakytkimin varusteltuun kaappiin. Kuviossa 27 nähtävillä olevassa kokoonpanossa käytettiin toista vastaavaa IRB-120 ro-
bottia. Erona ensimmäiseen kokoonpanoon oli, että tarttujan muoviset leuat oli kor-
vattu metallisilla ja kuvaustaso oli pystyasennossa. Muuten varustelu oli sama. Ohjel-
mistoon ja käyttöliittymään tehtiin myös muutamia muutoksia. Robotin liikese-
venssi muutettiin ahtaaseen kaappiin paremmin soveltuvaksi ja siihen lisättiin yli-
määräinen hitaampi toimintatila messuesittelyä ajatellen. Käyttöliittymään tehtiin
tarvittavat muutokset, jotta liikenopeuksia voitiin nopeasti vaihtaa.



Kuvio 26 Toisessa kokoonpanossa käytetty metallinen tarttuja.



Kuvio 27 Demosolu toisessa kokoonpanossa.

6.1 Konenäköohjelma

Kameralle luotiin ohjelma, jolla tunnistetaan halutut asiat tuotepakkauksesta. Tarkistetut asiat olivat värikerrosten sijaintien tarkistukset pullon etu ja takapuolelta, sekä takapuolelta lisäksi viivakoodin luku ja ylimmän tekstirivin tarkkuuden tarkistus.

Koska käytössä oli mustavalkokamera ei värejä voitu tunnistaa suoraan. Sen sijaan käytetyin työkalu oli hahmontunnistus, jolla etsittiin kullakin värillä tulostettuja kuviota. Ohjelma luotiin käyttämällä Easybuilder ohjelmointitilaa. Tällöin In-sight Explorer luo automaattisesti kaikki tagit muuttujille. Tagejä käytetään, kun robotin kontrolleri hakee kameran mittaustietoa ethernet verkon kautta. Easybuilder tilaa käytettäessä kameran ohjelmaa voidaan myös muokata helposti ilman että robotin koodia täytyy muokata.

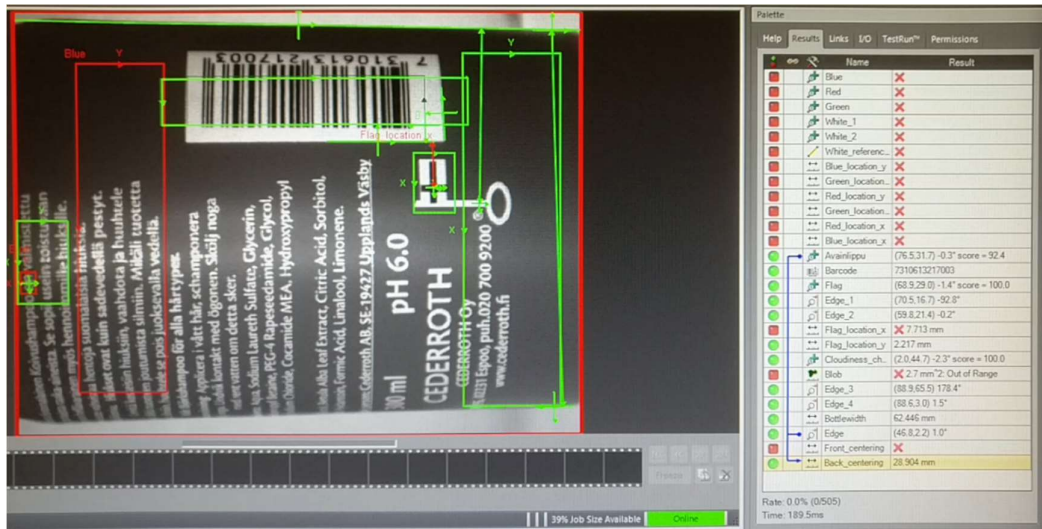
Pakkauksen tarkastelu alkoi etsimällä etupuolen logossa olevaa talon ikkunaa tai takapuolen suomilipun ristiä käyttäen hahmontunnistusta. Hakemisen nopeuttamiseksi kaikki hahmontunnistuksen olivat rajoitettu niiden oletteluihin esiintymisalueisiin. Kohteet eivät useimmiten olleet oikeassa asennossa ensimmäisessä kuvassa jolloin robotti käänsi ne suoraan kameraa kohti. Konenäköohjelma teki vain mittauksia. Hyväksytyt/hylätyt päätökset suoritettiin robotin ohjelmassa.

Etupuolella värikerrosten sijainnit paikallistettiin luomalla kiintopisteet logon värikerrosten osiin. Punaisen värin kiintopiste sijaitsee savupiipussa, vihreän pienissä joutsenissa, sinisen ikkunassa ja valkoisen logon alla olevassa tekstissä. Näiden kiintopisteiden etäisyydet mitattiin x- ja y-akseleilla. Ohjelma myös ottaa huomioon tarkasteltavan purkin asennon kuvassa, joten pieni heiluminen robotin otteessa ei aiheuttanut mittausvirheitä.



Kuvio 28 Pakkauksen etupuolen näkymä In-Sight ohjelmassa tarkistuksen aikana.

Pakkauksen takapuolelta tarkistettiin sinisen värikerroksen asento suhteessa valkoiseen värikerrokseen. Valkoisen kerroksen kiintopiste oli tällöin avainlipun ala-osassa ja sinisen kerroksen lipun ristissä. Saatujen tietojen mukaan pakkauksia valmistavan painokoneen painojälki alkoi satunnaisin väliajoin heiketä. Heikentynyt laatu aiheutti tekstin sumentumista eritoten ylimmällä tekstirivillä ja täten vei enemmän tilaa. Tekstin tarkkuus tarkistettiin mittaamalla tältä ylimmältä riviltä valitun sanan pinta-ala. Lisäksi pakkauksen takapuolelta luettiin viivakoodi.



Kuvio 29 Pakkauksen takapuolen näkymä In-Sight ohjelmassa tarkistuksen aikana.

6.2 Robotin ohjelma

Robotin ohjelmasykli alkoi käyttöliittymässä valitun näytteen noutamisella. Yhteensä valittavia näytteitä oli robotin työalueella neljä kerrallaan. Nopeampaa nivelliikettä käytettiin aina kun tarvetta lineaariselle liikkeelle ei ollut. Koordinaatistona käytettiin robotin maailmakoordinaatistoa. Näytettä robotti lähestyi ylhäältä käsin ja tarttui niihin tarttujallaan. Seuraavaksi robotti siirsi näytteen kameras eteen määritettyyn tarkistuspisteeseen. Tarkistuspisteessä robotti pyöritti tarkasteltavan tuotepakkauksen kameralle. Liikkeen aikana robotin kontrolleri myös valmisteli verkon yli yhteyden kokenäkökameraan sekä aloitti ajanoton tarkistuksen kestosta. Jotta tietoa saatiin liikuteltua robotin ja kameras välillä oli käytettävä jonkinlaista rajapintaa.

ABB myy roboteillensa ohjelmistolisäosaa jonka avulla Robotstudioissa voidaan suoraan ohjelmoida Cognexin kameroita. Koulun robottiin tätä ominaisuutta ei kuitenkaan ollut hankittu, joten rajapinta suunniteltiin ja ohjelmoitiin itse. Robotin kontrollerin ja kameran välinen rajapinta toteutettiin Cognexin esimerkkikoodin pohjalta. Rajapintakoodin kautta voitiin kutsua suoraan tagien nimeltä kameran mittausarvoja, jolloin haluttu arvo siirtyi automaattisesti verkon yli kameran muistista robotin muistiin. Esimerkkikoodissa viittaukset olivat kiinteitä kameran koodin spreadsheet näkyvän soluihin, mikä olisi johtanut siihen, että kaikki viittaukset olisi pitänyt päivittää käsin joka kerta kun kameran ohjelmaa muokattiin.



Kuvio 30 Robotti noutamassa näytettä.

Kun tuotepakkaus oli saapunut kameran eteen, lähetti robotti kameralle kuvanotto-käskyn verkon yli. Jos kamera löysi joko pakkauksen etupuolelta logon talon ikkunan tai takapuolelta avainlipun. Jos kumpaakaan ei löytynyt, käännettiin purkkia 90 ° myötä päivään ja otettiin uusi kuva. Tätä toistettiin enintään kolme kertaa ennen kuin pakkaus julistettiin suoraan hylätyksi. Kohteen löydyttyä se käännettiin suoraan kameraa kohti ja ensimmäiseltä puoliskolta otettiin varsinainen mittaus tieto talteen.

Seuraavaksi robotti käänsi pakkauksen toisen puolen suoraan oikeaan asentoon kameraa kohti. Eroa etu ja takapuolen mittauskohdilla oli aina noin 150 ° joten kun toisen tarkka sijainti saatiin selvitettyä, niin voitiin toisen sijainti laskea. Toiselta puolelta otettiin myös mittaustieto talteen robotin muistiin. Käyttöliittymässä näkyvän tarkistusajan otto myös pysäytettiin robotti lähdettyä palauttamaan pakkausta hakuun paikkaan. Robotin siirtyessä myös laskettiin tarkistetulle pakkaukselle pisteytys vertaamalla toleranssien sallimissa rajoissa mitattuja arvoja referenssiarvoihin.

Normaalijonon lisäksi pakkauksille voitiin myös ajaa kalibrointi ajo, jossa robotin muistiin tallennetaan vertailuun tarvittut referenssiarvot.



Kuvio 31 Kamera ottamassa kuvaa pakkauksen etupuolelta.

Pisteytyksessä maksimi oli kahdeksan ja sen alle jääneet pakkaukset hylättiin. Pisteytykseen vaikuttavia asioita olivat seuraavat:

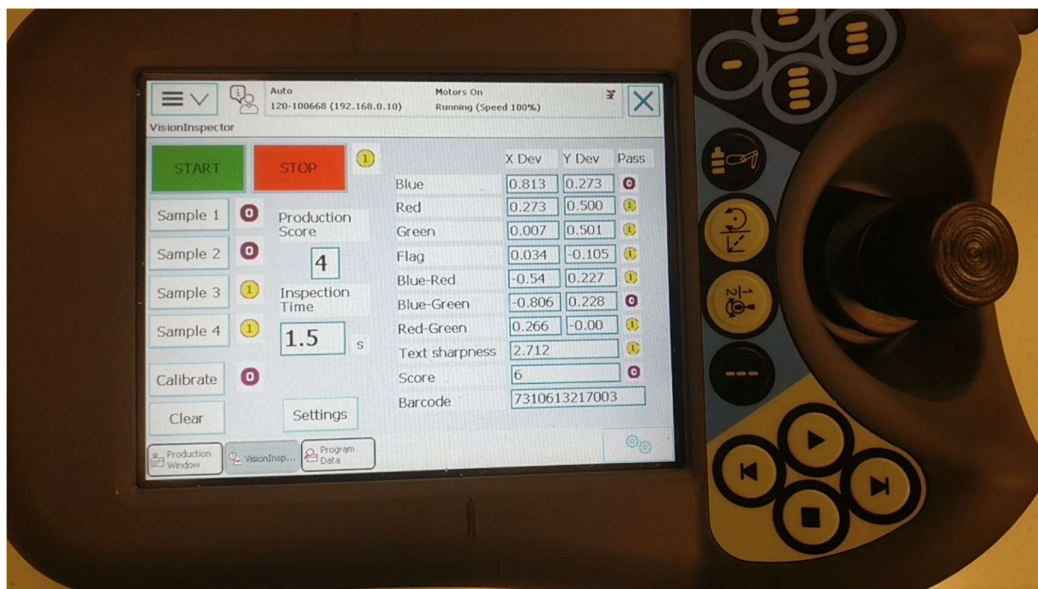
1. Sinisen värikerroksen sijainti referenssiin nähden.
2. Punaisen värikerroksen sijainti referenssiin nähden.
3. Vihreän värikerroksen sijainti referenssiin nähden.
4. Sinisen ja punaisen sijainti toisiinsa nähden.
5. Sinisen ja vihreän sijainti toisiinsa nähden.
6. Punaisen ja vihreän sijainti toisiinsa nähden.
7. Avainlipun ristien sijainti referenssiin nähden.
8. Takapuolen ylimmästä tekstirivistä poimitun sanan pinta-ala.

Värikerrosten siirtymät toisiinsa nähden lisättiin ohjelmaan, kun testauksessa todettiin, että järjestelmä hyväksyi pakkauksia, jotka olivat visuaalisesti tarkasteltaessa selvästi viallisia. Jos värikerrokset olivat liukuneet vastakkaisiin suuntiin, olisi tällöin hyväksytty maksimissaan kaksi kertaa toleranssin mukaisia virheitä. Kun kerrosten välinen vertailu otettiin mukaan, saatiin tämä ongelma poistettua.

Ohjelmassa myös luettiin pakkauksen viivakoodi ja se näytettiin käyttöliittymässä. Jos näytteitä olisi ollut useampaa laatua, sitä olisi voitu käyttää tunnistamaan pakkaustyyppi ja siten ottaa automaattisesti kullekin pakkaustypille oikeat vertausarvot käyttöön.

6.3 Käyttöliittymä

Käyttöliittymä luotiin Robotstudion Screenmaker lisäosalla kuviossa 32 näkyvään robotin ohjaimeen. Käyttöliittymä koostuu päänäkymästä ja asetussivusta. sen toiminta perustui tietokenttien linkittämiseen suoraan robotin muistipaikkoihin. Käyttöliittymästä pystyi valitsemaan näytteet, jotka robotti haki testattavaksi. Asetussivulta pystyi määrittämään testissä sallitut virhetoleranssit. Keskelle käyttöliittymän päänäkymää laitettiin tuotannon pisteytys, joka liikkui välillä 0-5. Kunnollisista pakkauksista annettiin yksi piste ja viallisista vähennettiin yksi. Pisteiden mennessä nolnaan annettiin virhetila mikä johti robotti pysähtymiseen. Tällä simuloitiin tilannetta, jossa tarvittaisiin operaattorin toimia liian monen virheellisen pakkauksen vuoksi. Näytössä esitettiin myös aika, joka meni pakkauksen tarkistukseen. Tätä käytettiin tarkistuksen tehokkuuden toteamiseen. Oikealla esitetään tarkka erittely pakkauksen pisteytyksestä. Pakkaus on hyväksytty, jos se sai täydet kahdeksan pistettä. Keltainen pallo kunkin rivin lopussa kuvasi hyväksytyä mittaustulosta.



Kuvio 32 Käyttöliittymä robotin Flexpendant ohjaimessa.

7 Testaus

Järjestelmän jokainen osa testattiin toimivaksi erikseen niiden toiminnalle sopivalla tavalla. Ohjelmistopuolella testaus tapahtui useimmiten syöttämällä funktioille tunnettuja arvoja ja observoimalla tulokset saatiin todennettua niiden toiminta. Esimerkiksi tiedonsiirtorajapinta testattiin vertaamalla käsin kameran ja robotin muistissa olevien arvojen vastaavuus.

Liikkeiden toimintaa observoitiin silmämääräisesti ja hidastetun videokuvan avulla. Videokuvasta voitiin selkeästi päätellä kohdat tarkistussekvenssissä, joita oli vielä mahdollista parantaa. Liikkeet testattiin ensin hitaalla ajolla ja sen jälkeen täydellä nopeudella, jotta mahdollisilta törmäyksiltä voitaisiin välttyä.

Mittausvirhe testattiin suorittamalla kalibrointi ja sen jälkeen testaamalla sama pullo useaan kertaan. Ilman kääntelyä mittaukselliset tulokset olivat odotetusti aina samat, joten parempien tulosten saamiseksi pulloa käännettiin käsin robotin otteessa satunnaisesti otoksien välillä.

Kokonaisuuden toiminta voitiin todeta, kun järjestelmä näytti odotettuja lopputuloksia kuviossa 30 näkyvässä käyttöliittymässä. Molemmille toteutetuille kokoonpanoille suoritettiin lopputestaus ajamalla useampi kierros täydellä nopeudella ja kirjaamalla

ylös käyttöliittymässä näkyvät tulokset. Tulosten vaihteluista pystyttiin päättelemään tarkistuksen toimivuus.

8 Tulokset

Syksyyn mennessä saatiin toteutettua kaksi testisolua, joilla molemmilla onnistuneesti tunnistettiin kaikki vaaditut asiat tuotepakkauksista. Ensimmäisellä kokoonpanolla saavutettu tarkistusnopeus oli yleensä 1.5-2 s luokkaa riippuen missä asennossa pullo on robotin tarttuessa siihen. Toisessa testikokoonpanossa testausaika oli hieman nopeampi 1.4-2 s eli hieman nopeampi johtuen paremmasta tarttujasta. Keskimääräinen testausaika oli molemmissa noin 1.8 s. Mitattu aika sisältää konenäkötar kasteluun sekä pullon pyörytykseen kuluneen ajan. Näytteen noutamiseen kulunut aika jätettiin pois laskuista niiden tapauskohtaisuuden vuoksi. Liikkeeseen kulunut aika riippuu lopullisessa toteutuksessa käytetyistä liike-etäisyyksistä jotka eivät testauksen aikana olleet tiedossa. Testikokoonpanoilla ei päästy toivottuun yhden sekunnin tahtiaikaan. Tämä kuitenkin oli odotettavissa, sillä käytettyihin laitteistoihin ei pystynyt käytännössä vaikuttamaan. Saavutettua tarkistusaikaa voidaan pitää onnistuneena jatkokehitystä ajatellen, jolloin yhden sekunnin tahtiaika pitäisi olla saavutettavissa.

Ensimmäisessä testauskokoonpanossa huomattiin, että käytetyn prototyyppitarttujan pehmusteet painuivat sisään pitkissä testausjaksoissa. Siitä johtuen pullojen asennot eivät enää aina testaussarjan loppupäässä olleet kunnollisia, mikä taas johti ylimääräisiin mittausvirheisiin. Toisessa testikokoonpanossa muoviset tarttujan leuat korvattiin metallisilla, joten siinä kyseistä ongelmaa ei enää ollut. Tarttujan optimointia kuhunkin työtehtävään voidaan pitää erittäin tärkeänä osana robottijärjestelmien suunnittelua.

Kameran konenäköohjelma saatiin toimimaan luotettavasti. Ohjelma sietä hyvin tarkastelukohteen heilumista. Toiminnan edellytykset olivat, että pullon keskiosa on suurin piirtein keskellä kuvaa ja että molemmat pullon kyljet näkyivät kuvassa. Kameran mittaustarkkuus käytetyllä kuvaus etäisyydellä oli laskennallisesti 0.1 mm. Käytännössä saavutettiin noin 0.3 mm mittaustarkkuus testikokoonpanossa yksi ja 0.2 mm tarkkuus testikokoonpanossa kaksi. Yhden kuvan käsittelyyn kameralta meni

noin 200 ms ja niitä otettiin yleensä 3-4 kpl riippuen pullon asennosta robotin tarttujassa. Suurimmat ongelmat liittyivät valon voimakkaaseen heijastumiseen pakkaus-ten etikettien tulostamiseen käytetystä musteesta. Heijastumat aiheuttivat satunnaisia häiriöitä hahmontunnistuksen toimintaan, jolloin myöskään etäisyysmittaukset eivät toimineet.

Opinnäytetyöprojektin päättyessä Vision Systemsin asiakas ei ollut vielä tehnyt päätöstä automatisointiprojektin jatkamisesta. Sovellusta esiteltiin myös Tekniikka 2016 – messuilla Jyväskylän Paviljongissa hiukan muunneltuna.

9 Pohdinta

Opinnäytetyön aihe oli mielenkiintoinen. Sain vapaasti kehittää ratkaisua virheiden tunnistamiseksi. Opin työn aikana paljon eri robottityypeistä ja konenäköratkaisuista. Laitteiden käytön opetteluun meni huomattavan paljon aikaa. Lopulta käyttämäni lähestymistapa ongelman ratkaisemiseksi oli mielestäni onnistunut. Kuitenkin olisin tahtonut testata useampaa kameramallia. Ainakin viivakamera olisi mielestäni ollut hyvä testata. Cognexin viivakameraa ei kuitenkaan ollut koululla saatavilla.

Teollisuusrobotteja valmistajat mainostavat paljon ohjelmistoillansa offline työskentelyn mahdollisuuksista. Vaikka Robotstudioissa on mahdollista simuloida robotteja laajasti, jouduin suurimman osan ajasta testaamaan toimintaa oikeilla laitteilla. Simulointi ei ollut mahdollista lähinnä koska kameran ja robotin yhteistyö oli toteutuksessa hyvin toisistaan riippuvaista.

Työssä toteutettu laitteisto on vain yksi monista tavoista, jolla tuotteiden konenäkö-tarkistus voitaisiin toteuttaa. Vaihtoehtoisesti voitaisiin käyttää useampaa kameraa. Jos pullot tulevat aina ulos painokoneesta samassa asennossa voitaisiin molemmat puolet kuvata suoraan yhdellä kertaa. Tällöin kuvaamisen menisi aikaa vain noin 200 ms jolloin manipulointiin jäisi jopa 800 ms.

Toinen vaihtoehto olisi käyttää viivakameraa, jolloin pyöryttämällä pullo kameralle saataisiin suoraan kuvattua koko pullon pinta tasaisesti. Viivakameralla saataisiin myös korkeamman resoluution kuva pullon pinnasta. IRB-120T variantti robotista on nopeampi 6-akseliltaan. Sen nopeus kyseisellä akselilla on 600 °/s jolloin pyörytyk-

seen menisi 600 ms ja kuvan analysointiin oletettavasti 200 ms. Yhdellä viivakameralla tarkistus ei siis ole välttämättä tarpeeksi nopea. Kahdella viivakameralla ja IRB-120T robotilla konenäkö tarkistuksen nopeus olisi 300 ms, jolloin manipulointiin ja konenäkö ohjelman ajoon jäisi 700 ms. Robotin käyttö pullojen tarkistuksessa ei ole myöskään välttämätöntä, mutta silloin menetettäisiin robotiikan mahdollistama järjestelmän helppo muokattavuus.

10 Lähteiden luotettavuus

ABB ja Cognex ovat suuria toimijoita aloillaan, joten heidän suosituksensa laitteiden käytöstä voidaan pitää luotettavina. Yhtiöt kuitenkin keskittyvät myymään omia tuotteitaan jolloin parhaan ratkaisun valinta jää suunnittelijan vastuulle.

Computer Vision Principles and Practice teoksen laatijoista kaksi työskentelee Karlsruhen Teknillisessä Instituutissa. Azad, P. on tohtori ja Dillmann, R. toimii samassa koulussa professorina. Molemmilla on useita julkaisuja. Teosta voidaan pitää luotettavana.

Introduction to Robotics, Mechanics and Control laatijalla Craig, J. on lukuisia alan julkaisuja. Asiantuntemusta voidaan pitää luotettavana.

The Robotics Primer kirjan tekijä Matarić, M. toimii professorina ja robottivuorovaiutuslaboratorion johtajana Etelä Kalifornian Yliopistossa. Laajasti julkaisuja robotiikan alalta, joten asiantuntemus on luotettavaa.

Introduction to Robotics, Analysis, Systems, Applications kirjan tehnyt Niku, S. työskentelee professorina Kalifornian Polyteknisessä Yliopistossa, joten pätevyyttä voidaan pitää luotettavana.

Modelling and Control of Robot Manipulators kirjan laatijoista Sciavicco, L. toimii robotiikan professorina Rooman kolmannessa Yliopistossa. Hän on tehnyt useita julkaisuja. Kirjan toinen tekijä Siciliano, B. toimii professorina Napolin Yliopistossa. Hänelläkin on useita tieteellisiä julkaisuja. Teosta voidaan siis pitää luotettavana lähteenä.

Image Processing, Analysis and Machine Vision kirjan tekijä Sonka, M. työskentelee Iowan Yliopistossa professorina. Julkaisuja hänellä on useita. Teoksen toinen tekijä

Hlavac, V. työskentelee professorina Prahan Teknisessä Yliopistossa. Kolmas kirjoittaja Boyle, R. toimii professorina Leedsin Yliopistossa. Kirjaa voidaan pitää luotettavana tietolähteenä.

Lähteet

ABB Robotics 2010. Technical reference manual, RAPID Instructions, Functions and Data types. Viitattu 10.10.2016.

[https://library.e.abb.com/public/688894b98123f87bc1257cc50044e809/Technical%20reference%20manual RAPID 3HAC16581-1_revJ_en.pdf](https://library.e.abb.com/public/688894b98123f87bc1257cc50044e809/Technical%20reference%20manual%20RAPID%203HAC16581-1_revJ_en.pdf)

ABB Robotics 2011. Operating manual, RobotStudio. Viitattu 10.10.2016.

https://library.e.abb.com/public/244a8a5c10ef8875c1257b4b0052193c/3HAC032104-001_revD_en.pdf

ABB Robotics 2012. IRB 120 Industrial Robot. Viitattu 10.10.2016.

https://library.e.abb.com/public/3bd625bab3c7cae1c1257a0800495fac/ROB0149EN_D_LR.pdf

Azad, P., Gockel, T., Dillmann, R. 2008. Computer Vision Principles and Practice. Iso-Britannia: Elektor International Media BV.

Cognex 2013. In-sight 7000 Series Datasheet. Viitattu 10.10.2016.

<http://www.cognex.com/downloads/LiteratureDelivery.aspx?id=10358>

Cognex 2016. Introduction to Machine Vision Viitattu 10.10.2016.

<http://www.cognex.com/ExploreLearn/FindOutMore/WhitePapersArticles/WhitePaperArticle.aspx?id=17152>

Craig, J. 2005. Introduction to Robotics, Mechanics and Control. USA: Pearson Education International, Inc.

Matarić, M. 2007. The Robotics Primer. USA: Massachusetts Institute of Technology.

Niku, S. 2001. Introduction to Robotics, Analysis, Systems, Applications. USA: Prentice Hall.

Sciavicco, L., Siciliano, B. 2000. Modelling and Control of Robot Manipulators. Iso-Britannia: Springer-Verlag London Ltd.

Sonka, M., Hlavac, V., Boyle, R. 2008. Image Processing, Analysis and Machine Vision. USA: Cengage Learning.

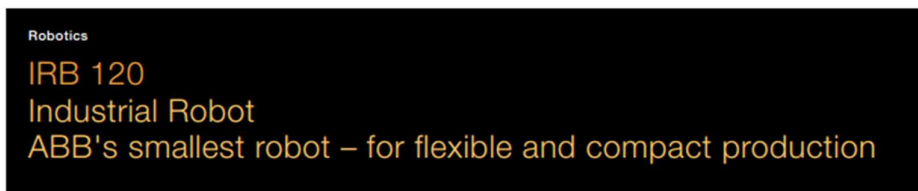
Suomen Robotiikkayhdistys Ry. 1999. Robotiikka. Suomi: Talentum Oyj.

Vision Systems Oy. Vision-konserni. Viitattu 24.8.2016.

http://www.visionsystems.fi/yrittys/vision_konserni/

Liitteet

Liite 1. ABB IRB-120 Esite



The IRB 120 robot is the latest addition to ABB's new fourth-generation of robotic technology and ABB's smallest robot ever produced. Ideal for material handling and assembly applications, the new IRB 120 robot provides an agile, compact and lightweight solution with superior control and path accuracy.

Compact and lightweight

As the smallest robot from ABB, the IRB 120 offers all the functionality and expertise of the ABB range in a much smaller package. Its reduced weight of only 25kg and compact design enables it to be mounted virtually anywhere, whether it is inside a cell, on top of a machine or close to other robots on the production line.

Multipurpose

Ideal for a wide range of industries including the electronic, food and beverage, machinery, solar, pharmaceutical, medical and research sectors, the IRB 120 joins ABB's fourth-generation of new robotic technology.

A white finish Clean Room ISO class 5 version enhances this versatility by making it suitable for environment with stringent cleanliness standard.

The six-axis robot handles a payload of up to 3kg (4kg with its wrist down) and with a reach of 580 mm, the robot is able to carry out a series of operations using flexible rather than hard automated solutions. The IRB 120 is the perfect building block to design cost effective applications – especially when space is at a premium.

Easy to integrate

Weighing in at only 25kg, this robot arm is truly the most portable and easy to integrate on the market. It can be mounted at any angle without any restriction. The smooth surfaces are easy to clean and the cables for air and customer signals are internally routed, all the way from the foot to the wrist, ensuring that integration is effortless.

Optimized working range

In addition to a horizontal reach of 580 mm, the robot has best in class stroke and the ability to reach 112 mm below its base. Furthermore, the IRB 120 has a very compact turning radius, which is enabled by the robots symmetric architecture, without offset on axis 2. This ensures the robot can be mounted close to other equipment and the slim wrist enables the arm to reach closer to its application.



Fast, accurate and agile

Designed with a light, aluminum structure, the powerful compact motors ensure the robot is enabled with a fast acceleration, and can deliver accuracy and agility in any application. Using the IRB 120T variant, cycle-times can be reduced up to 25% where the work piece needs extensive re-orientation and axis 4, 5 and 6 are predominantly used. This faster versions is well suited for pick and packing applications and guided operations together with PickMaster 3™.

IRC5 Compact controller – optimised for small robots

ABB's new IRC5 Compact controller takes the capabilities of the extremely powerful IRC5 controller and presents them in a truly compact format. The new Compact controller brings accuracy and motion control to applications, which previously had been exclusive to large installations.

In addition to space saving benefits, the new controller also enables easy commissioning through one phase power input, external connectors for all signals and a built-in expandable 16 in, 16 out, I/O system.

RobotStudio for offline programming enables manufacturers to simulate a production cell to find the optimal position for the robot, and provide offline programming to prevent costly downtime and delays to production.

Reduced footprint

For applications where floor space is crucial, the combination of the new compact, lightweight architecture of the IRB 120 with the new IRC5 Compact controller introduces a significantly reduced footprint.

IRB 120

Specification			
Variants	Reach	Payload	Armload
IRB 120-3/0.6	580 mm	3 kg (4kg)*	0.3 kg

Features	
Integrated signal supply	10 signals on wrist
Integrated air supply	4 air on wrist (5 bar)
Position repeatability	0.01 mm
Robot mounting	Any angle
Degree of protection	IP30
Controllers	IRC5 Compact / IRC5 Single cabinet

Movement			
Axis movements	Working range	Maximum speed	
		IRB 120	IRB 120T
Axis 1 Rotation	+165° to -165°	250 °/s	250 °/s
Axis 2 Arm	+110° to -110°	250 °/s	250 °/s
Axis 3 Arm	+70° to -110°	250 °/s	250 °/s
Axis 4 Wrist	+160° to -160°	320 °/s	420 °/s
Axis 5 Bend	+120° to -120°	320 °/s	590 °/s
Axis 6 Turn	+400° to -400°	420 °/s	600 °/s

Performance		
	IRB 120	IRB 120T
1 kg picking cycle		
25 x 300 x 25 mm	0.58 s	0.52 s
25 x 300 x 25 with	0.92 s	0.69 s
180° axis 6 reorientation		
Acceleration time 0-1 m/s	0.07 s	0.07 s

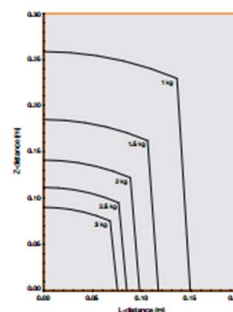
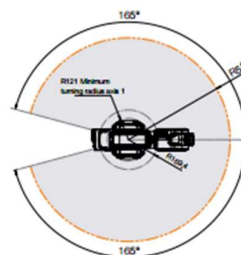
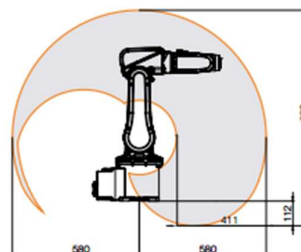
Electrical connections	
Supply voltage	200–600 V, 50/60 Hz
Rated power	
Transformer rating	3.0 kVA
Power consumption	0.25 kW

Physical	
Dimension robot base	180 x 180 mm
Dimension robot height	700 mm
Weight	25 kg

Environment	
Ambient temperature for Robot manipulator:	
During operation	+5°C (41°F) to +45°C (122°F)
Relative transportation and storage	-25°C (-13°F) to +55°C (131°F)
For short periods	up to +70°C (158°F)
Relative humidity	Max 95%
Options	Clean Room ISO class 5 (certified by IPA)**
Noise level	Max 70 dB (A)
Safety	Safety and emergency stops 2-channel safety circuits supervision 3-position enabling device
Emission	EMC/EMI-shielded

* With vertical wrist
** ISO class 4 can be reached under certain conditions
Data and dimensions may be changed without notice

Working range at wrist center & load diagram



© Copyright ABB Robotics. ROB0149EN_LD_May 2012

www.abb.com/robotics

Power and productivity
for a better world™

ABB



In-Sight 7000 Series

INTEGRATED VISION SYSTEM | SMART CAMERA

- Self-contained vision systems
- Flexible, integrated lighting and lens options
- Autofocus feature and fast image capture
- Unmatched ease-of-use
- Powerful Cognex vision tool library

You will find so many applications for the award-winning In-Sight® 7000 series of vision systems. These self-contained smart cameras feature autofocus, fast image capture, integrated lighting and lens with powerful vision tools for inspection, color and OCR models, and more. They also have the capability to power and control a range of external lighting—all in a compact, industrial IP67 package measuring 75mm x 55mm x 47mm.



Flexible lighting

The integrated, field-replaceable lighting options (red, blue, green, white and infrared) give you total flexibility. Unlike most vision systems, the In-Sight 7000 additionally has the capability to power and control external lighting directly eliminating the need for external power supplies which occupy valuable machine space.

Autofocus and lens options

You can easily set and save the optimal focus values associated with each job on your line. The autofocus feature simplifies setup for situations requiring regular part changes or projects that require the system to be placed in hard-to-reach spaces. In addition, integrated field-replaceable lenses, like C-mount, allow you to further customize each system for specific applications.

Fast image capture

The In-Sight 7000 delivers the highest acquisition speeds of all In-Sight products at over 100 frames per second. This rate provides reliable 100% automated inspection of products on the fastest production lines.



Benefits of all In-Sight vision systems

Intuitive software options

The In-Sight Explorer spreadsheet view provides a robust, flexible and efficient way to configure the vision tools for your application. The flexible In-Sight Explorer EasyBuilder® user environment makes applications easy to set up and deploy for any user.
www.cognex.com/insightexplorer

Smart tools

Patented, industry-leading tools such as PatMax®, OCRMax™, color, ID, blob, calibration, edge, and flexible flaw detection are just a click away.
www.cognex.com/visiontools

System validation

TestRun validation enables you to refine, test and verify inspection systems automatically.
www.cognex.com/testrun

Built-in communications

The Cognex Connect™ suite offers the widest range of built-in communication protocols that interface to any PLC, robot or HMI on your factory network.
www.cognex.com/connect

Complete visualization

The VisionView® operator interface is ideal for monitoring and controlling systems on the factory floor—available on four platforms.
www.cognex.com/visionview

Expansive hardware portfolio

Depending on your environment, mounting options, performance speed requirements, image acquisition type or application specific needs, there is an In-Sight vision system model for you.
www.cognex.com/visionsystems

In-Sight 7000 Series

Model ¹	Speed Rating ²	Acquisition ³ (fps)	Resolution		User Interface		Supported Tools ⁴							Part Number			
			800 x 600	1280 x 1024	EasyBuilder	Spreadsheet	Base Tools	Essential Tools	Extended Tools	ID Tools	Color Tools	PatMax Available	OCR Only				
7010	2x	102	•		•		B						C			IS7010-01	
7020	2x	102	•		•		B	E								IS7020-01	
7050	2x	102	•		•	•	B	E								IS7050-01	
7200	6x	102	•		•	•	B	E	X		I	C	P			IS7200-01, IS7200-11 (PatMax)	
7400	12x	102	•		•	•	B	E	X		I	C	P			IS7400-01, IS7400-11 (PatMax)	
7402	12x	60		•	•	•	B	E	X		I	C	P			IS7402-01, IS7402-11 (PatMax)	
OCR Models																	
7230	6x	102	•		•											O	IS7230-01
7430	12x	102	•		•											O	IS7430-01
7432	12x	60		•	•											O	IS7432-01
ID Models																	
7210	6x	102	•		•	•										I	IS7210-01
7410	12x	102	•		•	•										I	IS7410-01
7412	12x	60		•	•	•										I	IS7412-01

Notes:

- In-Sight 7000 has four high speed input and four high speed output built in. I/O modules are available for additional I/O.
- Speed rating compared to In-Sight Micro 1020 model and does not include image acquisition rate. The 5604 and 5614 models have acquisition speed rated in lines per second.
- Acquisition rate is based on minimum exposure, and a full image frame capture.
- Supported Tools:
 - B** Base tool set includes brightness, contrast, pattern, edge, point-to-point geometry, distance, angle, plot and blob tools.
 - E** Essential Tool Set includes blob, edge, curve and line finding, histogram and geometry tools, image filters, pattern matching, and standard calibration.
 - X** Extended Tool Set includes non-linear calibration and caliper tool. PatMax option available.
 - I** ID tool set includes: 1-D/2-D barcode reading and verification, text reading and verification (OCR/OCV) and image filters.
 - C** Color tool set includes MatchColor, ExtractColor, color histogram, color to greyscale filters and color to greyscale distance filter (7010 model with color identification only).
 - P** Includes PatMax, Cognex patented geometric pattern matching technology.
 - O** OCR model, EasyBuilder only, OCRMax, location tools and a subset of the Base tool set.



**Entry-Level Model:
In-Sight 7010**

In-Sight 7010 is an entry-level smart camera vision system developed specifically for inspection tasks where vision sensors are too limited and other entry-level vision systems fall short of your needs. www.cognex.com/IS7010

Americas

United States, East +1 508 650 3000
 United States, West +1 650 969 8412
 United States, South +1 615 844 6158
 United States, Detroit +1 248 668 5100
 United States, Chicago +1 630 649 6300
 Canada +1 905 634 2726
 Mexico +52 81 5030 7258
 Central America +52 81 5030 7258
 South America +1 909 247 0445
 Brazil +55 47 8804 0140

Europe

Austria +43 1 23060 3430
 Belgium +32 2 8080 692
 France +33 1 4777 1550
 Germany +49 721 6639 0
 Hungary +36 1 501 0650
 Ireland +353 1 825 4420
 Italy +39 02 6747 1200
 Netherlands +31 208 080 377
 Poland +48 71 776 0752
 Spain +34 93 445 67 78
 Sweden +46 21 14 55 88
 Switzerland +41 71 313 06 05
 Turkey +90 212 371 8561
 United Kingdom +44 1327 856 040

Asia

China +86 21 5050 9922
 India +9120 4014 7840
 Japan +81 3 5977 5400
 Korea +82 2 539 9047
 Singapore +65 632 55 700
 Taiwan +886 3 578 0060

© Copyright 2013, Cognex Corporation. All information in this document is subject to change without notice. All Rights Reserved. In-Sight, VisionView, EasyBuilder, PatMax, Cognex, Cognex.com and the Cognex logo are registered trademarks and Cognex, Cognex, OCRMax and Make It Right are trademarks of Cognex Corporation. All other trademarks are property of their respective owners. Lit. No. 0705-1304

COGNEX
www.cognex.com

Corporate Headquarters
One Vision Drive Natick, MA 01760 USA
Tel: +1 508 650 3000 Fax: +1 508 650 3344

**MAKE IT
RIGHT**