



TEKNILLINEN TIEDEKUNTA

Konenäkö automaatiojärjestelmässä

Janne Ikonen

KONETEKNIIKAN TUTKINTO-OHJELMA

Kandidaatintyö

Maaliskuu 2022

TIIVISTELMÄ

Konenäkö automaatiojärjestelmässä

Janne Ikonen

Oulun yliopisto, Konetekniikan tutkinto-ohjelma

Kandidaatintyö 2022, 41 s. + 0s liitettä

Työn ohjaaja(t) yliopistolla: Yrjö Louhisalmi

Kandidaatintyön tavoitteena on tutustua kuvankäsittelyyn, erilaisten kuvankäsittelyn sovellutuksiin, sekä niiden mahdollisuuksiin TwinCAT Vision-järjestelmässä. Työssä testataan ja dokumentoidaan Vision-ohjelman ominaisuuksia. Samalla myös pohditaan järjestelmän suorituskykyä.

Asiasanat:Konenäkö, TwinCAT Vision, Kuvankäsittely

ABSTRACT

Machine vision in automation system

Janne Ikonen

University of Oulu, Degree Programme of Mechanical Engineering

Bachelor's thesis 2022, 41 pp. + 0pp. Appendixes

Supervisor(s) at the university: Yrjö Louhisalmi

Bachelor's thesis was to familiarize with the subject of image processing, different image processing applications and their possibilities in TwinCAT Vision system. In thesis Vision software functions are tested and documented. Also, performance of the system is evaluated.

Keywords: Machine vision, TwinCAT Vision, Image Processing

ALKUSANAT

Tein kandidaatintyön osana tekniikan kandidaatin tutkintoa Oulun yliopistossa. Työ koostuu kirjallisuuskatsauksesta sekä lyhyestä käytännön osiosta. Kirjallisuuskatsauksessa käsitellään yleisimpiä koneäön menetelmiä, tutustutaan koneäön markkinoilla oleviin tuotteisiin sekä käydään tarkemmin läpi TwinCAT Vision ohjelmiston ominaisuuksia. Käytännön osiossa käydään läpi järjestelmän käyttöönottoa, sekä tehdään käytännön testejä laitteistolla. Työ pohjautuu kirjallisuuteen, valmistajien internet sivuihin sekä yliopiston laitteistolla tehtyihin kokeisiin.

Kiitän Yrjö Louhisalmea työn ohjauksessa ja avusta laitteiston käytössä. Kiitän myös Oulun yliopistoa työn käyttövaiheessa tarvittavien välineiden lainaamisesta.

Oulu, 03.03.2022

Janne Ikonen

SISÄLLYSLUETTELO

TIIVISTELMÄ

ABSTRACT

ALKUSANAT

SISÄLLYSLUETTELO

| | |
|---|----|
| 1. Johdanto | 7 |
| 2. Konenäkö ja sen yleisimmät menetelmät | 8 |
| 2.1 Esittely | 8 |
| 2.3 Kynnysarvomenetelmä | 10 |
| 2.4 Reunantunnistus | 12 |
| 2.5 Esineen sijainti, erottelu ja määrän laskeminen | 14 |
| 2.6 Optinen merkintunnistus | 15 |
| 2.7 Kuvasta mittaaminen | 17 |
| 2.8 Värien tunnistus..... | 18 |
| 2.9 Viivakoodin lukeminen | 19 |
| 2.10 Syväoppiminen ja konenäkö | 20 |
| 3. Konenäkö ja ohjelmoitava logiikka..... | 22 |
| 3.1 Markkinoilla olevia tuotteita | 22 |
| 3.2 Bechkoff Twincat 3 vision | 24 |
| 4. Twincat Vision käyttöä | 26 |
| 4.1 Järjestelmän käyttöönotto..... | 26 |
| 4.2 Vision mallisovelluksien esittelyä..... | 32 |
| 5. Pohdintaa..... | 38 |
| 6. Lähteet..... | 39 |

1. JOHDANTO

Käsittelen tutkielmassani konenäköä ja sen sovelluksia automaatiassa. Valitsin aiheen sen ajankohtaisuuden vuoksi. Tämä on aihe, jota on kehitetty jo pitkään, mutta tämänhetkisen teknologian myötä konenäköä hyödynnetään kasvavissa määrin. Konenäköä voidaan pitää jatkuvasti kehittyvänä alana ja siksi on tärkeää ymmärtää tämän teknologian mahdollisuudet. Konenäön hyödyntäminen automaatiassa on merkittävä osa-alue siirtyessämme teollisuus 4.0:aan.

Konenäkö on aiheena laaja ja jotta tutkielman laajuus ei kasvaisi liian suureksi, kirjallisuuskatsauksessa käsitellään konenäön menetelmiä vain pintapuolisesti. Tämä kuitenkin on antaa kuvan siitä, miten konenäköä voidaan hyödyntää, sen rajoitteista sekä sen vaatimuksista. Käyttövaiheessa rajallisen laajuuden vuoksi työssä rajoitutaan TwinCAT Vision ohjelman käyttöönottoon sekä muutamaa demon käyttöön. Uskon tämän osion olevan hyödyllistä tietoa uusille käyttäjille, sillä internetistä saatava tieto on tällä hetkellä vielä hyvin rajallista ja vaikeasti löydettävää.

2. KONENÄKÖ JA SEN YLEISIMMÄT MENETELMÄT

2.1 Esittely

Konenäkö on kameroiden hyödyntämistä automaatio-sovellutuksissa. Sen tarkoituksena on luoda digitaalinen mallinnus todellisesta maailmasta. Konenäkö ei rajoitu pelkästään teolliseen maailmaan, vaan sitä on käytössä myös muun muassa autoissa, lentokoneissa, videovalvonnassa ja sään ennustamisessa. Konenäkö on jatkuvasti kehittyvää ja yhä kasvavissa määrin hyödynnettyä teknologiaa. (Jain et al. 1995)

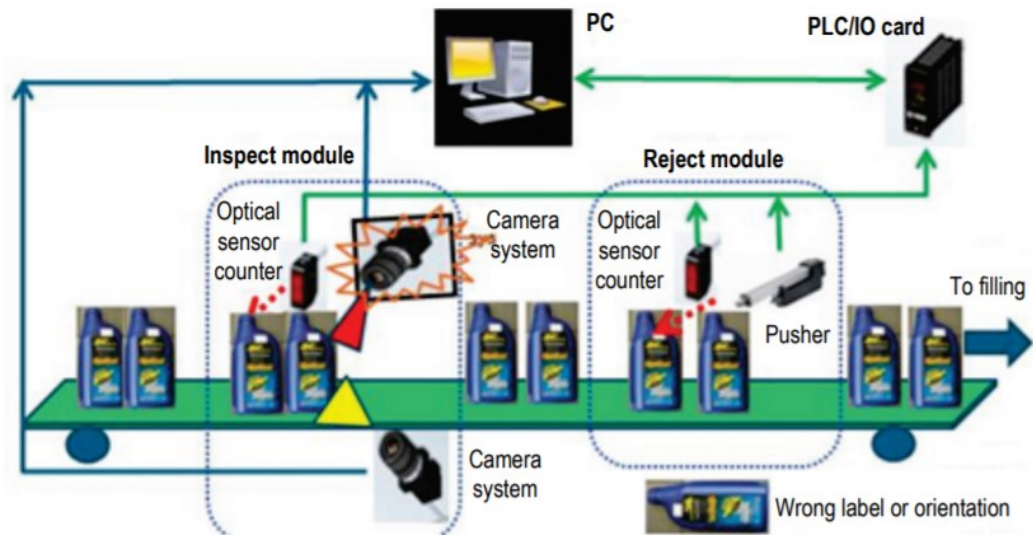
Konenäköä hyödynnetään teollisuudessa tyypillisesti tuotteiden, kuten autojen, prosessoreiden, ruuan ja lääkkeiden tarkastamiseen. Konenäön kehittyminen on mahdollistanut tarkastusprosessien täydellisen automatisoinnin, jonka takia näiden tehokkuus ja tarkkuus ovat nousseet. (Labudzki et al. 2014)

Tyypillinen konenäköön perustuva tarkastusjärjestelmä sisältää seuraavat elementit:

- Kamera (mustavalko tai väri), kuvan digitalisoija ja mahdolliset erilaiset linssit
- Tietokone tai prosessori
- I/O-laitteisto
- Valaistus kameralle
- Kuvankäsittelyohjelma
- Anturi kuvattavan esineen tunnistamiseksi kohdalla
- Logiikka, jonka mukaan laitteisto poistaa viallisen esineen

Yllä mainitut elementit mahdollistavat konenäön käyttämisen tuotteen tarkastamiseen. Ensimmäisenä tuote saapuu kuvauspaikalle, jolloin lähestymisanturi lähettää käskyn kameralle kuvauksesta. Myös valaistus on usein järkevää asettaa samaan käskyyn, jolloin valot ovat käytössä vain kuvaushetkellä. Seuraavaksi kuva muutetaan digitaaliseen muotoon tietokoneen konenäköohjelmalle. Ensin ohjelmassa kuva käsitellään erilaisilla kuvankäsittelyn menetelmillä. Kuvan käsittely mahdollistaa kuvan tarkastamisen sekä parantaa sen tuloksia. Kun kuva on käsitelty ja analysoitu, tietokone käskyyttää laitteistoa

I/O-laitteiston kautta. Toimenpiteet määräytyvät ohjelmoidun logiikan mukaan, kuva 1. (Labudzki et al. 2014)



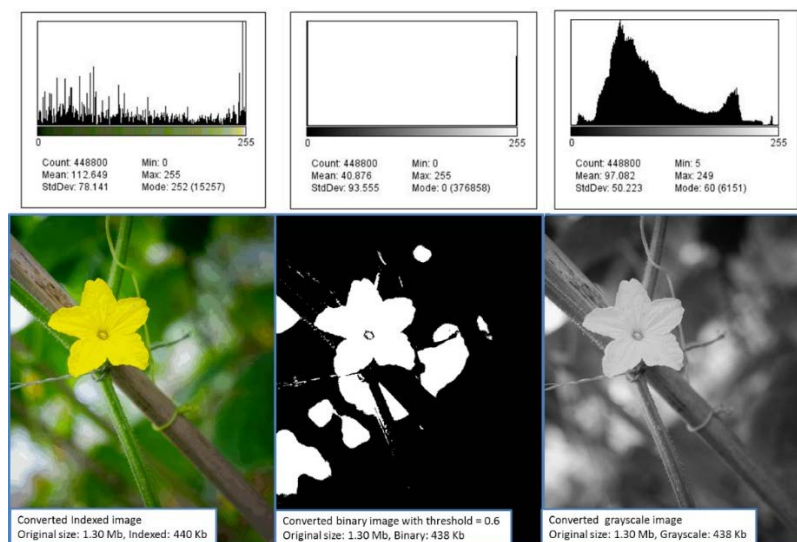
Kuva 1. Kuvassa yksinkertainen järjestely öljypurkkien tarkastamiseen (Labudzki et al. 2014).

Käsittelyn tarkoituksena on parantaa kuvan laatua, jotta niiden analysointi olisi mahdollista ja että niistä saatu data olisi mahdollisimman tarkkaa. Yleensä samassa prosessissa käytetään useampaa menetelmää samanaikaisesti. Seuraavissa osioissa käydään läpi tyypillisiä kuvankäsittelyn menetelmiä, joita konenäkö käyttää hyödykseen. Tarkoituksena on kuitenkin esitellä aiheita sekä niiden toimintaperiaatteita vain pintapuolin, eikä näissä tulla tarkastelemaan laskennallista perustaa.

2.3 Kynnysarvomenetelmä

Useissa kuvankäsittelyn sovellutuksissa kuvassa täytyy erotella esineet taustasta. Käytännössä tummat esineet vaaleaa taustaa vasten muutetaan täysin mustiksi ja tausta valkoiseksi (tai toisinpäin), jolloin saadaan aikaiseksi kaksivärinen eli binäärikuva, kuva 2. Tätä toimenpidettä kutsutaan kynnysarvomenetelmäksi (eng. Thresholding), eli asetetun kynnysarvon mukaan kuvan muokkaamista tietokoneelle helpommin luettavaksi. (Davies 2012 kpl. 4)

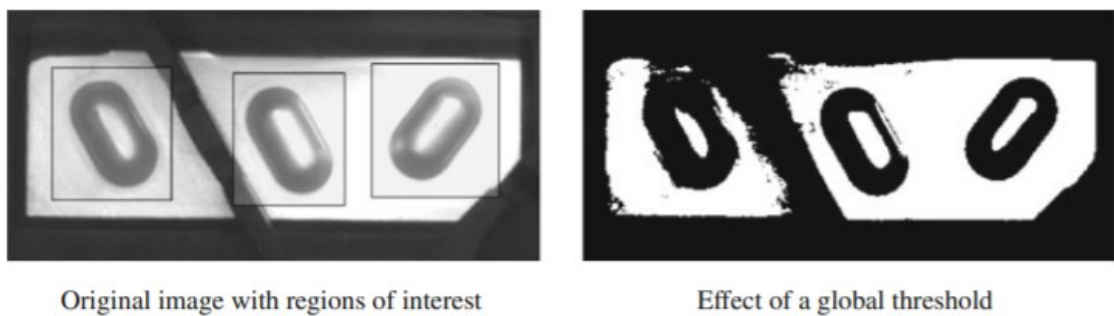
Binäärikuvan luominen aloitetaan harmaasävykuvasta, joka sisältää 256 eri harmaan sävyä. Seuraavaksi valitaan kynnysarvo, jonka mukaan pikseleitä muutetaan joko mustiksi tai valkeiksi. Esimerkiksi vaalealla taustalla olevan tumman esineen kaikki asetetun raja-arvon suuruiset tai ylittävät pikselit muutetaan mustiksi ja alittavat valkeiksi. (Demant et al. 2013 s. 84)



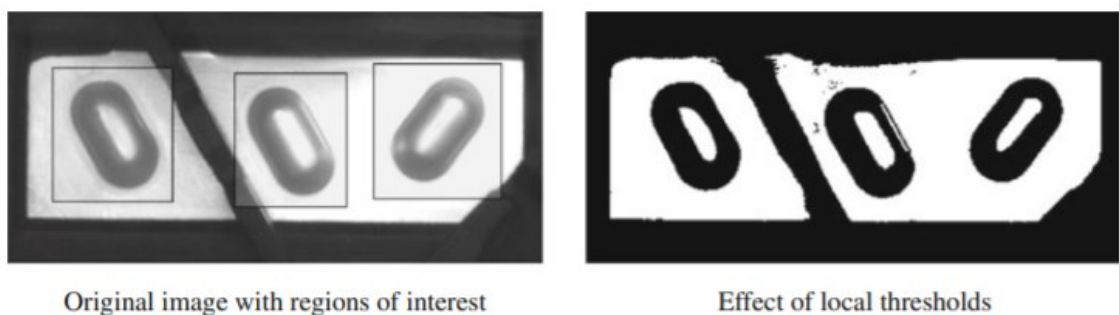
Kuva2. Rgb-, binääri- ja harmaasävyskaala -versiot samasta kuvasta (imagingknowledgebin.blogspot.com. 2022).

Yksinkertaisin tapa binäärikuvan luomiseen on käyttää samaa kynnysarvoa kaikille kuvan pikseleille. Tämä kuitenkin saa helposti aikaan virheitä ja kohinaa kuvassa, johtuen vaihtelevista kirkkauden tasoista. Globaaliin kynnysarvoon yleisiä virheiden aiheuttajia ovat vaihtelevat valaistuksen tasot kuva-alueella sekä väritykseltään epätasainen tausta.

Jotta yhtä globaalia kynnysarvoa voitaisiin käyttää luotettavasti, täytyisi kuvaolosuhteet olla tarkkaan mietityt ja toistettavat. Tästä syystä onkin usein järkevää käyttää kirkkauden mukaan muuttuvaa kynnysarvoa, joka lasketaan alueittain. Kuvassa 3 muunnokseen on käytetty yhtä yleistä kynnysarvoa, kun taas kuvassa 4 alueellisia. Globaalin arvon tapauksessa rajat alkavat sekoittumaan taustaan, kun taas alueellisilla esineiden rajat pysyvät selkeinä. (Demant et al. 2013 s. 85-86)



Kuva 3. Kuvassa käytetty yleistä kynnysarvoa (Demant et al. 2013).



Kuvat 4. Kuvassa käytetty alueellisia kynnysarvoja (Demant et al. 2013).

Kynnysarvon määrittelyssä käytetään harmaan sävyn histogrammia, ts. pylvästaulukkoa harmaan sävyistä sekä niiden esiintymistajuudesta kuvassa. Histogrammin muodostetaan analysoimalla jokainen pikseli erikseen ja lisäämällä pylvästaulukoon kyseisen harmaansävyn kohdalle. (Demant et al. 2013 s. 87)

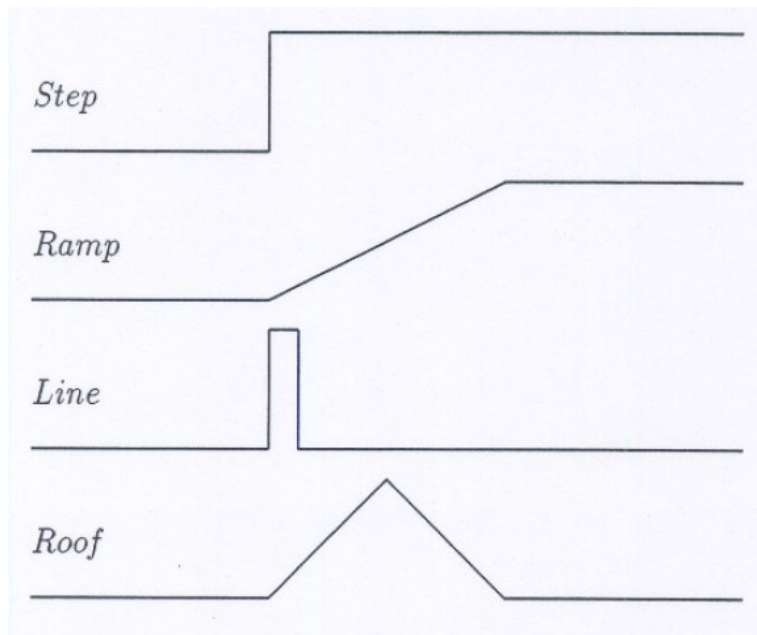
Kynnysarvomenetelmällä saadaan usein hyvä arvio esineiden rajoista. Kynnysarvomenetelmän haasteena on kuitenkin, oikean raja-arvon löytäminen. Harvoin todellisen elämän sovellutuksissa voidaan käyttää yhtä asetettua globaalia arvoa, vaan

vaihtelevissa olosuhteissa, kuten tehdasympäristössä on järkevämpää käyttää erilaisia automaattisia menetelmiä alueellisten kynnsarvojen löytämiseksi. Yleisiä kuvan laatua heikentäviä tekijöitä ovat kirkkaista esineistä aiheutuvat heijastukset sekä esineiden aiheuttamat varjot. Hyvän erottelun varmistamiseksi oikeanlaisen valaistuksen ja taustan valinta on tärkeää. Monessa sovellutuksessa vaihtoehtoinen tapa kynnsarvomenetelmälle on reunan etsintä, jota käsittelemme seuraavassa luvussa. (Davies 2012 kpl. 4)

2.4 Reunantunnistus

Reunantunnistus on vaihtoehtoinen menetelmä esineiden erotteluun kynnsarvomenetelmälle. Monesti reunan tunnistus soveltuu paremmin, sillä kynnsarvomenetelmä on usein häiriöherkkä ja vaatii paljon laskennallista tehoa paikallisten kynnsarvojen laskemiseksi. Reunantunnistuksessa yleisesti periaatteena on löytää tarpeeksi suuri intensiteetin muutos, joka merkkää reunan sijaintia, ts. reunan kohta määräytyy kohtiin, joissa kuvan kirkkaus muuttuu terävästi. Toisin sanoen verrataan valitun pikselin ympärillä olevia intensiteettejä toisiinsa, ja jos ero ylittää asetetun kynnsarvon, tämä merkkää reunan paikkaa. (Davies 2012 kpl. 5)

Ideaalisessa tapauksessa intensiteetin muutokset ovat joko askelepäjätkuvuuksia tai viivaepäjätkuvuuksia. Askelepäjätkuvuuden tapauksessa intensiteetin arvo muuttuu reunan kohdalla äkillisesti ja pysyy sellaisena. Viivaepäjätkuvuudessa intensiteetti muuttuu äkillisesti, mutta palaa takaisin alkuperäiseen arvoonsa lyhyen matkan päästä. Todellisuudessa nämä äkilliset muutokset ovat harvinaisia ja usein ovat pyöristyneet kohinan takia mittaussignaaleihin päästyään. Niinpä äkillisistä askelepäjätkuvuuksista tulee ramppiepäjätkuvuuksia ja viivaepäjätkuvuuksista tulee kattoepäjätkuvuuksia, kuva 5. Kohinaa ja siitä aiheutuvia ongelmia koitetaan vähentää erilaisilla suodattimilla. (Jain et al. 1995 kpl. 5)

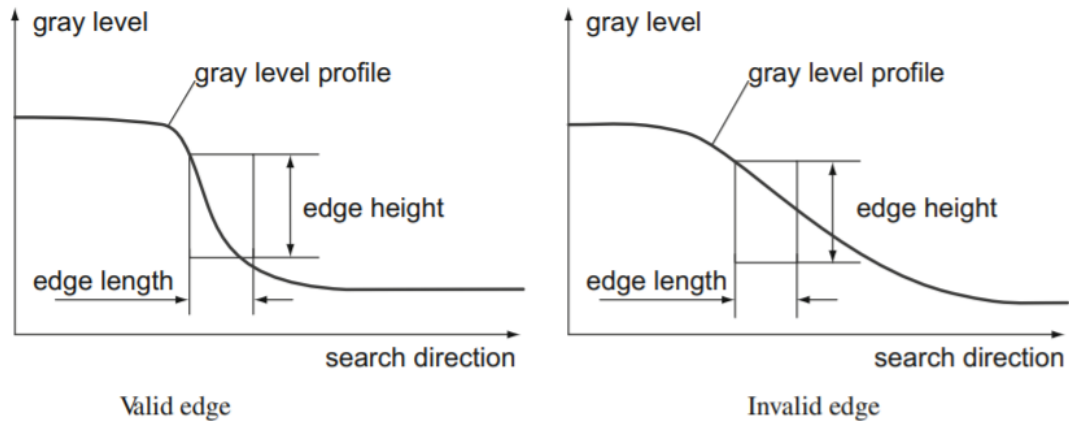


Kuva 5. Ideaaliset epäjatkuvuudet ja niiden vastineet (Jain et al. 1995).

Jotta kuvasta löytyisi mahdollisimman tarkasti vain haluttuja reunoja ja jotta virheellisiä reunoja ei pääsisi mukaan, täytyy etsinnän reunaehtoja määrittellä. Reunaehtojen asettaminen nopeuttaa myös operaatiota vähentämällä laskennallista työmäärää. Kun kaikki halutut määrittelyehdot tuodaan yhteen, puhutaan reunamallista. (Demant et al. 2013 s. 107)

Reunanetsinnässä kuljetaan ennalta määrättyä reittiä, jonka mukaan harmaasävykuvan tasoja mitataan. Tämän tiedetyn suunnan myötä voimme käyttää reunan suuntaa reunaehtojen määrittelyssä. Jos reitillä kuljetaan vaaleasta tummaan, on kyseessä laskeva reuna. Kun tummasta siirrytään vaaleaan, puhutaan nousevasta reunasta. (Demant et al. 2013 s. 107)

Toinen reunaa määrittelevä tekijä on reunan korkeus ts. minimi reunankorkeus. Tällöin reunan vaatimuksena on riittävän korkea nousu tai lasku harmaasävytasolla. Tähän samaan kuuluu myös minimi reunan pituus, eli vähimmäispituus, jolla minimi reunankorkeus täytyy tapahtua, kuva 6. (Demant et al. 2013 s. 107)



Kuva 6. Kuvassa vasemmalla reunamallin ehdot täyttävä tason lasku, oikealla lasku on liian loiva täyttääkseen reunan kriteereitä. Kuvaajissa y-akselilla on harmaan sävyn taso ja x-akselilla siirtymä reitillä. (Demant et al. 2013)

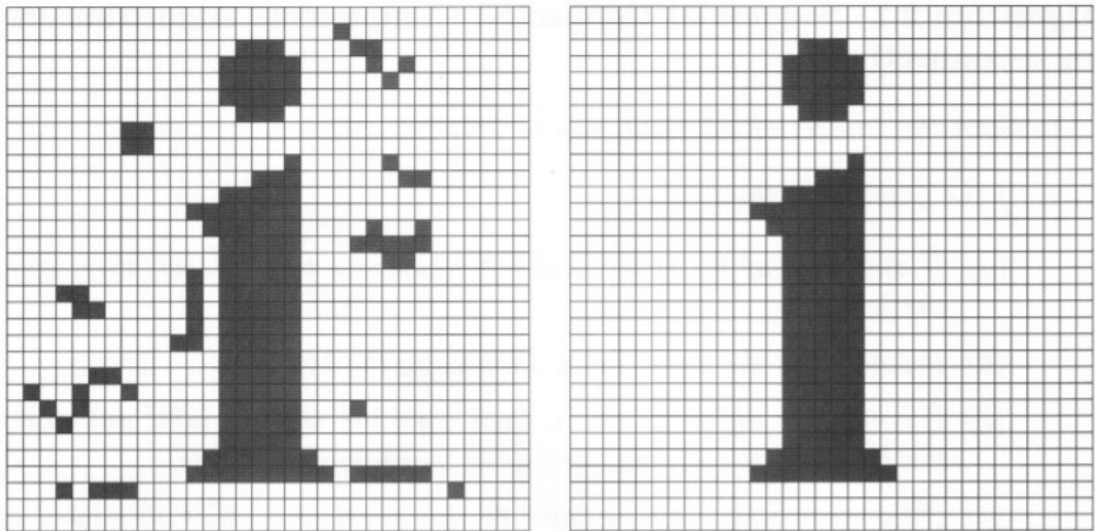
2.5 Esineen sijainti, erottelu ja määrän laskeminen

Monessa konenäön sovellutuksessa, etenkin teollisuudessa tarvitsee tietää kuvassa olevan esineen sijainti. Tämän selvittämisessä käytetään apuna esineen massakeskipistettä tai esineen ympäröivän suorakulmion keskipistettä. Lisäksi tarvitaan tieto kameran sijainnista suhteessa kuvattavaan alueeseen. Kun molemmat ovat tiedossa voidaan kuvassa olevan esineen sijainti sitoa todelliseen xy-koordinaatistoon. Toimintavarma ja ei kohina-altis tapa selvittää esineen sijainti on käyttää binäärikuvasta laskettua alueen massakeskipistettä. Binäärikuvasta massakeskipiste saadaan selville laskemalla keskiarvo esineen pikseleistä suhteessa xy-koordinaatistoon. (Jain et al. 1995 s. 32)

Binäärikuvasta esineiden erottelu perustuu ohjelmoinnissa käytettävään tulvatäyttö operaatioon, jossa kaikki nurkistaan tai sivuistaan yhdistyneet esineen pikselit lasketaan samaksi esineeksi. Tällöin kyseessä on kahdeksan naapurin yhteys. Esineet voidaan kuitenkin myös laskea neljän naapurin yhteydellä, jolloin pikselit katsotaan olevan samaa esine, jos näillä on yhteinen sivu. Kun kuvassa on useampi toisistaan erillään oleva esinettä, voidaan algoritmin avulla laskea esineiden määrä ja nimetä nämä halutun logiikan mukaan, kuten koon perusteella. Itse esineiden pikselit voidaan myös erotella

halutun logiikan mukaan. Esimerkiksi esineen reunapikselit voidaan merkitä, jolloin binäärikuvasta saadaan merkityksi reunat. (Jain et al. 1995 kpl. 2)

Jos kuviosta on selvitetty esineen koko pikseleiden määränä, voidaan hyödyntää kohinan poistoon soveltuvaa kokosuodinta. Kokosuodin on hyödyllinen työkalu kynnysarvomenetelmällä luotujen binäärikuvien kohinan poistamisessa. Suodin poistaa kuvasta pikseliryppäät, jotka ovat haluttua kokoa pienempiä, kuva 7. (Jain et al. 1995 kpl. 2)

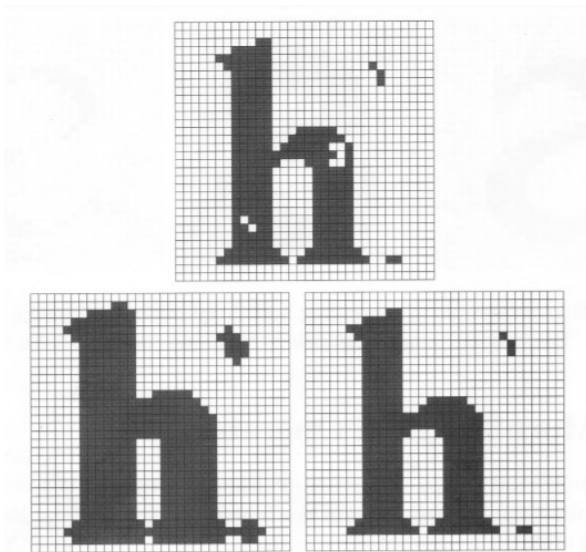


Kuva 7. Esimerkki kokosuotimesta binäärikuvassa. Esimerkissä käytetty kokosuotimen arvoa $T=10$, jolloin kuvasta poistetaan kaikki alle 10-pikseliä pienemmät kappaleet (Jain et al. 1995).

2.6 Optinen merkintunnistus

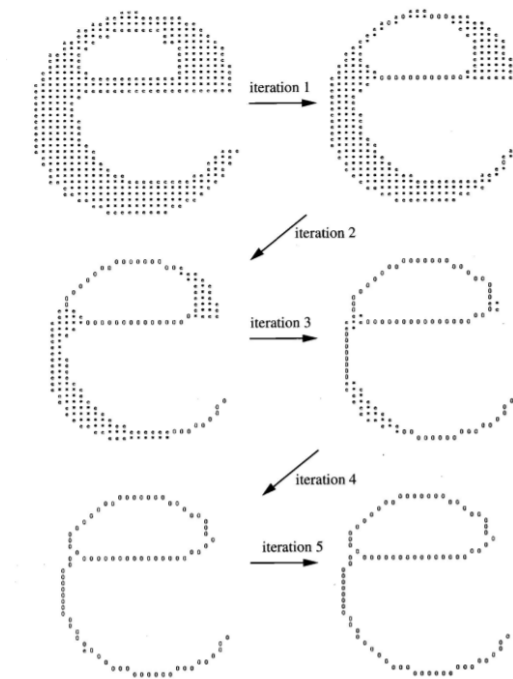
Optinen merkintunnistus (eng. Optical Character Recognition OCR) pystyy tunnistamaan automaattisesti kuvasta tai skannauksesta käsin kirjoitettua ja tulostettua tekstiä. Ensimmäinen vaihe tunnistuksessa on kuvan ottaminen, joko kameralla tai skannerilla. Seuraavaksi kuva muutetaan binäärikuvaksi kynnysarvomenetelmällä. Tämän jälkeen kuvasta poistetaan kohina, sekä reiät ja merkit ohennetaan. Viimeiseksi käytetään opetettua tekoälyä, joka tunnistaa merkit. (Ravina et al. 2013)

Huonolaatuisesta tekstistä, pölystä tai huonosta valaistuksesta johtuvia reikiä poistetaan täyttämällä. Lisäksi tunnistamisen helpottamiseksi käytetään ohennusta. Yksi tapa poistaa binäärikuvamuunnoksessa ilmestyneitä aukkoja on käyttää laajennusta sekä kutistusta peräjälkeen. Laajennuksessa kaikki 0-pikselit, joilla on 1- pikseli naapurinaan muutetaan 1-pikseliksi. Kutistuksessa sen sijaan kaikki 1-pikselit, jotka ovat yhteydessä 0-pikseliin muutetaan 0-pikseliksi, kuva 8. (Jain et al. 1995 kpl. 2)



Kuva 8. Kuvassa käytetty ensin laajennus operaatiota, jonka jälkeen kutistusta. Tämä menetelmä on poistanut merkissä olleet aukot, muttei ympärillä olevaa kohinaa (Jain et al. 1995).

Ohennus on kuvankäsittelyn toimenpide, jossa binäärikuvassa olevasta esineestä tai merkistä poistetaan ulkokehän pikseleitä, kunnes jäljelle jää vain arvioitu keskilinja, kuva 9. Ohennuksen tarkoituksena on jättää kuvasta jäljelle vain oleellinen tieto, jota tarvitaan tunnistamiseen. Tämä myös helpottaa tekoölyn tekemää merkintunnistusta. (Jain et al. 1995 s. 57)



Kuva 9. Kuvassa merkin-e ohennus, kunnes muutosta ei enää tapahdu ja jäljellä on enää merkin ”ranka” (Jain et al. 1995).

2.7 Kuvasta mittaaminen

Mittaaminen kuvasta voidaan karkeasti jaotella kolmeen osa-alueeseen. Ensimmäinen näistä on kokoonpanon varmentaminen. Tässä mittaamalla varmistetaan, että kokoonpanossa on kaikki elementit olemassa ja oikeilla kohdillaan. Varmentamisessa voidaan esimerkiksi tarkistaa porattujen reikien olemassaolo. Kokoonpanon varmistamisessa ei tarvita suurta tarkkuutta. Toisena osa-alueena mittaamisessa on muodon tarkastus. Muodon tarkastuksen tarkoituksena on varmistaa, ettei työkappaleessa ole poikkeamia. Esimerkkeinä tästä ovat reunan suoruuden tarkastus, asennettujen komponenttien suoruuden ja asennusmuodon tarkastus. Tämän osa-alueen sisäinen tarkkuus (eng. precision) täytyy olla hyvä, mutta ulkoisen tarkkuuden (eng. accuracy) ei tarvitse olla erittäin tarkka. Viimeinen osa-alue on itse mittaaminen. Kuvasta voidaan mitata trigonometrian operaatioita, kuten pituuksia, säteitä ja kulmia. Tämä on näistä kolmesta tehtävästä raskain. Mittaamisen ulkoisen ja sisäisen tarkkuuden täytyy olla

korkealla tasolla, mikä on laskennallisesti vaativa operaatio. Hyvällä kameralla ja valaistuksella voidaan päästä 1 μm tarkkuuteen. (Demant et al. 2013 kpl. 7)

Mittaamisessa pituus voidaan määrittää esimerkiksi yksittäisten pikseleiden välillä, reunojen avulla tai keskiöiden kautta. Tavan valinta on sovelluskohtainen eikä yksi tapa sovi kaikkiin tarkoituksiin. Mittauksissa sovelletaan perinteistä trigonometriaa. Kun halutaan laskea esimerkiksi kahden reiän etäisyys toisistaan, käytetään apuna reikien keskiöitä ja niiden pikseleiden xy-koordinaatteja. Tällöin pituus saadaan yksinkertaisella Pythagoraan lauseella. (Demant et al. 2013 kpl. 7)

2.8 Värin tunnistus

Yleensä konenäön sovellutuksissa ei käytetä värikuvaa, vaan näissä hyödynnetään harmaasävykuvia. Yksi syy tälle on se, että harmaasävykuvien käsittely vaatii huomattavasti vähemmän laskennallista tehoa ja vie vähemmän muistia. Lisäksi kuvankäsittelyn ohjelmat on kehitetty harmaasävykuvien käsittelyyn ja nämä toimivat hyvin. Myöskään näiden kuvankäsittelyn logiikoiden uudelleen kehittäminen värikuville ei olisi helppoa saatikka kannattavaa. Näistä haitoista huolimatta joskus täytyy hyödyntää värikuvia värin tunnistuksen muodossa. (Demant et al. 2013 kpl. 11)

Värisävyn perusmäärittäminen sijoittuu RGB asteikolle (lyh. red, green, blue). Tällä asteikolla päävärit punainen, vihreä ja sininen saavat arvoja väliltä 0–255. Kun kaikki kolme väriä ovat arvoltaan 255, saadaan valkoinen väri. Kun taas kaikkien ollessa nolli, saadaan aikaiseksi musta. Päävärien eri suhteilla saadaan aikaiseksi muut värikartan variantit. Värisävyn lisäksi ihminen havaitsee värin kirkkauden ja sen haaleuden esim. pastelliväreissä. Kun nämä otetaan huomioon, saadaan aikaiseksi HSI-värikartta. (Demant et al. 2013 kpl. 11)

Yksinkertaisimmillaan värintunnistus perustuu värihistogrammiin. Tällä menetelmällä saadaan RGB tasolla halutun pikselin väri selville. Värihistogrammi toimii kuten harmaavärihistogrammi. Siinä tutkittavan alueen jokaista pikseliä verrataan tietokantaan ja vastaava arvo lisätään histogrammiin. Histogrammista saatavan tuloksen perusteella voidaan määrittää tutkitun osion väri. (Funt and Finlayson 1995 s. 522-529)

Väriin määrittämisessä oikeanlaisen valaistuksen tärkeys korostuu. Varjokohdat ja heijastukset aiheuttavat suuriakin vaihteluita samanvärisen kappaleen väriskaalassa. Myös epätasaisten pintojen väri vaihtelu tulisi huomioida. (Demant et al. 2013 kpl. 11)

2.9 Viivakoodin lukeminen

Perinteisesti viivakoodeja on luettu laserskannereiden avulla. Näissä skanneri heijastaa lasersädettä viivakoodiin, josta se heijastuu takaisin skanneriin. Tämän jälkeen heijastunut valo muutetaan signaaliksi ja dekodataan. Nykytekniikka on kuitenkin mahdollistanut kuvapohjaisen koodin lukemisen. Kameralla skannaamisen etuina ovat parempi lukuprosentti, myös vioittuneiden koodien lukumahdollisuus sekä 2D-koodien, kuten QR-koodien lukemisen mahdollisuus. Kamerapohjainen tunnistus on lisäksi järkevää, jos skannauksen lisäksi prosessiin kuuluu muitakin kameralla toteutettavia tehtäviä. Perinteisen laserskannerin haittoina ovat huono lukuprosentti ei standardin mukaisissa olosuhteissa. Skanneri ei myöskään pysty lukemaan kasvavissa määrin käytettyjä 2D-koodeja. Skannerit käyttävät myös helposti hajoavia ja kuluvia liikkuvia osia. Lisäksi laserskannerilla voidaan käsitellä vain yhtä koodia kerrallaan. Näistä syistä kuvapohjainen koodin tunnistus tulee kehittymään ja yleistymään entistä enemmän tulevaisuudessa. (Silver 2017)

Kuvapohjaisen koodinluvussa ensimmäinen vaihe on ottaa riittävän tarkka kuva. Tämän jälkeen ohjelman täytyy etsiä kuvasta koodi ja poimia 1D-signaali koodista, kuva 10. Viimeisenä tapahtuu signaalin dekoodaaminen. Jotta prosessi olisi tehokas, täytyy kuvanottovaiheessa kameran tarkennuksen olla riittävän nopea ja valaistuksen riittävä. (Silver 2017)



Kuva 10. Kuvassa viivakoodin muunnos signaaliksi (VisionSpectra. 2022).

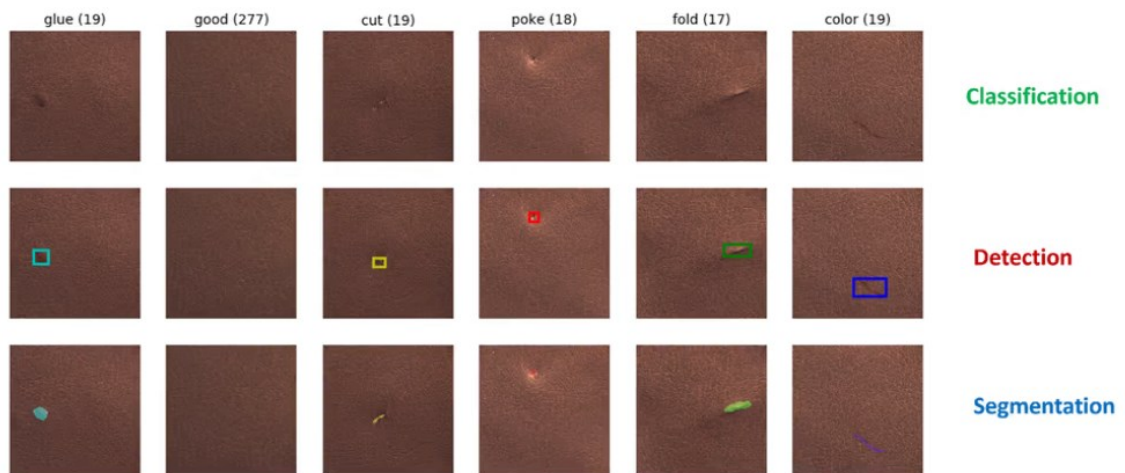
2.10 Syväoppiminen ja konenäkö

Syväoppiminen on konenäössä paljon hyödynnetty menetelmä. Se tukee jo olemassa olevia konenäön työkaluja. Toisin kuin perinteiset työkalut, kuten viivakoodin lukeminen tai mittaaminen, tällä menetelmällä ei saada logiikan mukaan laskettua arvoa, vaan tieto siitä, onko kuvassa oleva esine se mitä haettiin. Syväoppiminen konenäössä on siis kuvan täsmäämistä. (Wilson 2019)

Syväoppiminen on tietokonemalli, jossa ohjelma oppii ja pystyy tämän jälkeen tekemään johtopäätöksiä oppimansa perusteella. Syväoppiminen koittaa emuloida ihmisen aivojen toimintamekanismeja. Syväoppimisen järjestelmiä on useita, mutta yleensä ne kaikki hyödyntävät neuroverkkoja, todennäköisyysmalleja sekä erilaisia oppimismalleja. Syväoppiminen on herättänyt paljon yleistä kiinnostusta, kun sen on huomattu suoriutuvan perinteisiä kuvankäsittelyn metodeja paremmin. (Voulodimos et al. 2018)

Syväoppiminen ei toimi kirjoitetun logiikan pohjalta, kuten perinteiset konenäön menetelmät. Syväoppiminen pohjautuu sen sijaan sille syötettyyn dataan, josta järjestelmä oppii suunniteltuun tehtäväänsä. Tästä datasta järjestelmä poimii erilaisia toistuvia ominaisuuksia, joista se pystyy jatkossa tunnistamaan esineitä ja asioita. (Dechow 2021)

Syväoppimisen käyttöönotto menee yleisesti seuraavien päävaiheiden mukaisesti: Ensimmäisenä valitaan käyttötarkoitukseen sopivin järjestelmä. Seuraavaksi kerätään tarkoituksenmukaisia kuvia ja ne nimetään järjestelmän koulutusta varten. Nimeämisessä määritellään käyttötarkoituksesta riippuen hyvät ja huonot kuvat. Esimerkiksi vianetsinnässä huonot kuvat ovat niitä, joista voidaan todeta virhe esineessä. Huonot kuvat voidaan myös määritellä vikatyypin mukaan, kuva 11. Tämän jälkeen ohjelman oppiminen voidaan käynnistää. Oppiminen todennetaan syöttämällä samantyyppisiä kuvia, ilman opetusdataa. Opetus ja tason todennus toistetaan, kunnes järjestelmä on halutulla tasolla. Järjestelmään voidaan myös lisätä eri luokkia, jälkikäteen opettamalla kyseiset tilanteet edellä mainitulla tavalla. (Dechow 2021)



Kuva 11. Kuvassa esimerkki syväoppimisen tehtävistä, jossa hyvät ja virheelliset kuvat ovat jaoteltuina. Virheelliset kuvat ovat jaoteltuina virheen tyyppin mukaan (Mariner USA 2021).

3. KONENÄKÖ JA OHJELMOITAVA LOGIIKKA

3.1 Markkinoilla olevia tuotteita

Kuvankäsittelyä hyödynnetään laajalti ohjelmoitavassa logiikassa ja tällä tekniikalla on kysyntää. Tästä syystä monet valmistajat ovatkin tuoneet markkinoille omat tuotteensa, joilla kuvankäsittely yhdistetään logiikkaan. Seuraavassa osiossa käsitellään suurten PLC-valmistajien konenäön tuotteita sekä niiden ominaisuuksia. Tässä kappaleessa ei käsitellä Beckhoff TwinCat Vision tuotetta vaan sen yksityiskohtaisempi esittely tapahtuu seuraavassa kappaleessa.

Siemens

Siemens omistaa laajan valikoiman tuotteita automaatioon liittyen ohjelmistotasolta komponenttitasolle. Konenäön osalta Siemensin tuoteperhe kulkee nimellä SIMATIC MV. Tämä tuoteperhe tarjoaa kameroita ja ohjelmistoja optiseen lukemiseen. Ohjelmistojen osalta tämä tuoteperhe sisältää 1D/2D-koodien lukemisen, optinen merkin tunnistuksen (OCR), kappaleen tunnistuksen, kappaleen sijainnin tunnistuksen sekä kappalemäärän laskemisen. (Siemens 2022)

ABB

ABB:n konenäön tuoteperhe Integrated Vision sisältää konenäön kameroita, ohjaimia sekä ohjelmistoja. ABB on selkeästi keskittynyt konenäön osalta robottien sovellutuksiin. Valmistajan mukaan tyypillisimpiä käyttötarkoituksia ovat: esineen sijainnin löytäminen, esineen visuaalinen tarkastus, lajittelu ja tunnistus. Konenäön ohjaus perustuu ABB:n tietokonekäyttöympäristössä toimivaan RobotStudioon, joka on myös saatavilla ilmaisversiona. (ABB 2022)

Keyence

Keyencen valikoimaan kuuluvat kaikki konenäköön liittyvät komponentit sekä ohjelmistot. Keyencen oma ohjelmointiympäristö on pc-pohjainen KV studio. Keyencen

konenäön valikoima kattaa seuraavat aihealueet: visuaalinen tarkastus, sijainnin määrittäminen, mitoitus, määrän tarkastus, tekstin- sekä 1/2D-koodin tunnistus. (KEYENCE 2022)

Hitachi

Hitachi on konenäön osalta keskittynyt ainoastaan yhdelle osa-alueelle. Yritys markkinoi tekstinlukujärjestelmää, joka on tarkoitettu teollisuuden mustesuihkuprinttien lukemiseen. Valmistaja tarjoaa tuotesarjassaan kaikki tunnistukseen tarvittavat komponentit sekä tähän liittyvää ohjelmistoa. Lukemisen avulla voidaan pitää asetettujen parametrien, kuten eränumeroiden mukaan kirjaa määristä. Lisäksi järjestelmä voi tunnistaa virheelliset tulosteet. Hitachin ohjelmoitavan logiikan ohjelmat ovat LEW (tikapuulogiikka) ja Pro-H, joka sisältää viisi eri ohjelmointityyppiä. (Hitachi 2022)

IDEC

IDEC:n konenäkö sisältää kameroista lähtien kaiken tunnistamiseen tarvittavan ohjelmistotasolle saakka. Valmistajan tietokonepohjainen ohjelmointi- ja ohjausohjelmisto on AutomationOrganizer. Valmistajan sivujen mukaan konenäköjärjestelmän työkaluja ovat kuvion ja pinnan tunnistus, optinen merkin tunnistus, kierretyn tekstin luenta, sijainnin tunnistus, mittaus, reunojen tunnistus ja laskeminen sekä kirkkaus- ja kontrastiperusteiset operaatiot. (IDEC 2022)

Omron

Omron tarjoaa laajan valikoiman konenäön komponentteja ja ohjelmia. Valikoima sisältää älykameroita, jotka pystyvät käsittelemään kuvaa ja ohjaamaan I/O-laitteita itsenäisesti. Omron valmistaa myös perinteisiä kameroita, jotka vaativat erillisen ohjelmiston kuvan käsittelyyn ja ohjaukseen. Ohjelmistona pc-pohjaiseen ohjelmointiin ja ohjaukseen Omronin tuote on Sysmac Studio. Omronin tuotteilla pystytään laajasti kaikkiin edellä mainittujen valmistajien toimintoihin sekä lisäksi värin tunnistukseen. (Omron 2022)

Yllä mainittujen lisäksi muita merkittäviä konenäkövalmistajia ovat mm. B&R industrial automation systems, Boch Rexroth ja Panasonic.

3.2 Beckhoff Twincat 3 vision

Automaatiovalmistaja Beckhoff on tuonut markkinoille vision konenäköohjelmiston, joka toimii lisäosana TwinCat 3 automaatio-ohjelmaansa. Twincat 3 on Pc-pohjainen automaatio-ohjelmointi- ja ohjaussovellus. TwinCat 3 Vision on laitteistoneutraali, eli järjestelmä on yhteensopiva eri valmistajien kameroiden kanssa. Beckhoff ei itse valmista kameroita. Liitettävät kamerat täytyy olla GigE Vision -standardin mukaisia ja ne voivat olla joko alueskannauskameroita tai juovaskannauskameroita. Itse ohjelmisto tarjoaa valikoiman valmiita työkaluja kuvankäsittelyyn, mutta ohjelma tukee myös itsetehtyjä algoritmeja. Ohjelma pystyy käyttämään livekuvaa, nauhoitettua videokuvaa, kuvia tai videosta kuvakaapattuja kuvia. Työkaluja ja järjestelmiä voidaan siis kehittää ja testata ilman kameraakin. (Beckhoff 2022)

Vision -työkaluja ovat mm. mittauksen saralla: etäisyyden mittaus, halkaisijan mittaus ja pyöreiden mittaus. Havaitsemisen ja tunnistamisen osalta kuvion tunnistus, sijainnin määrittäminen, värin tunnistus, datamatriisi-, QR- ja viivakoodien lukeminen. Ohjelman avulla voidaan myös monitoroida laitteiden toimintaa livekuvan avulla. Visionin työkaluja kehitetään jatkuvasti ja esimerkiksi optinen merkin tunnistus (OCR) on tulossa lisäksi valikoimaan. (Beckhoff 2022)

Vision ohjelmiston asentamiseen tarvitsee TwinCat ohjelman pohjalle, johon vision ohjelma tuodaan lisäosana. Kaikkia Beckhoffin ohjelmia voi käyttää ilmaisversiona, jolloin ilmaislisenssi täytyy uusua viikon välein. Itse Vision ohjelmaa saa neljänä eri versiona. Ensimmäisenä kuitenkin täytyy olla asennettuna vision perusversio TF7100 Base. Tämän jälkeen voidaan asentaa Vision lisäosapaketteja. TF7100 sisältää perustoimintoja, kuten kuvan segmentoinnin, värikuvan prosessoinnin toimintoja, esineiden erottelun ja analyysin. Lisäosan: TF7200 Vision Matching 2D avulla voidaan tunnistaa ja vertailla esineitä opittujen ominaisuuksien perusteella. Tämän kautta

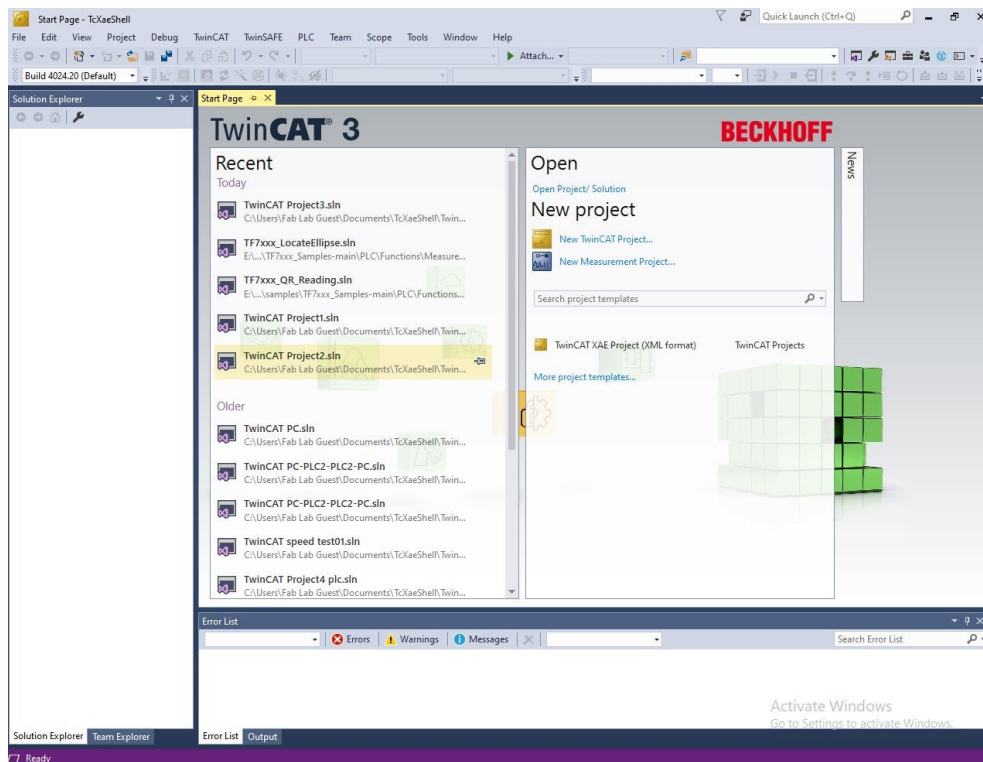
järjestelmä voi esimerkiksi lajitella esineitä. TF7250 Vision Code Reading sisältää työkalut kuvapohjaiseen koodinlukemiseen. Tämän lisäosan kautta pystytään lukemaan 1D ja 2D koodeja. Työkalun avulla voidaan esimerkiksi seurata tuotteita niiden tuotannon eri vaiheissa. TF7300 Vision Metrology 2D on lisäosa mittaamiseen. Tämän avulla voidaan esimerkiksi tunnistaa reunoja, reikiä ja kaaria. Ohjelman kautta on myös mahdollista suorittaa erinäisiä mittaustehtäviä, kuten pituuden mittaus, halkaisijan mittaus, kulman mittaus ja koordinaattien määrittäminen. Tätä lisäosaa voidaan käyttää esimerkiksi robotin tartuntapisteiden löytämiseen. Näiden kuvankäsittelyn sovellusten lisäksi tarvitaan TF7000 TwinCat 3 GigE Vision Connector -paketti, jonka avulla voidaan yhdistää enintään kaksi kameraa ohjelmoitavaan logiikkaan. Tähän on saatavilla useamman kameran mahdollisuus lisäosien kautta, jolloin yhteen järjestelmään voidaan liittää enintään 16 kameraa. (Beckhoff 2022)

Beckhoffin valikoimassa on myös kuvankäsittelyyn integroitu käyttöliittymä eli HMI (Human Machine Interface). Tämä mahdollistaa käyttöliittymästä suoraan prosessin livekuvan tai still-kuvien tarkkailun. Käyttöliittymän kautta voidaan tarkkailla montaa eri kameraa. HMI voidaan määrittää suorittamaan erinäisiä toimintoja, kuten livekuvan pysäyttäminen, kuvien valinta, zoomaus, kuvaan piirtämien, väritasapainon muuttaminen, valitun pikselin koordinaattien näyttäminen, kuvan koon ilmoittaminen, valitun kohdan väritiedot ja muodon ilmoitus. Tämä valmis käyttöliittymä on vaihtoehto itsetehdyille käyttöliittymälle, jonka voi luoda tuettujen C++:n tai Matlabin avulla. (Beckhoff 2022)

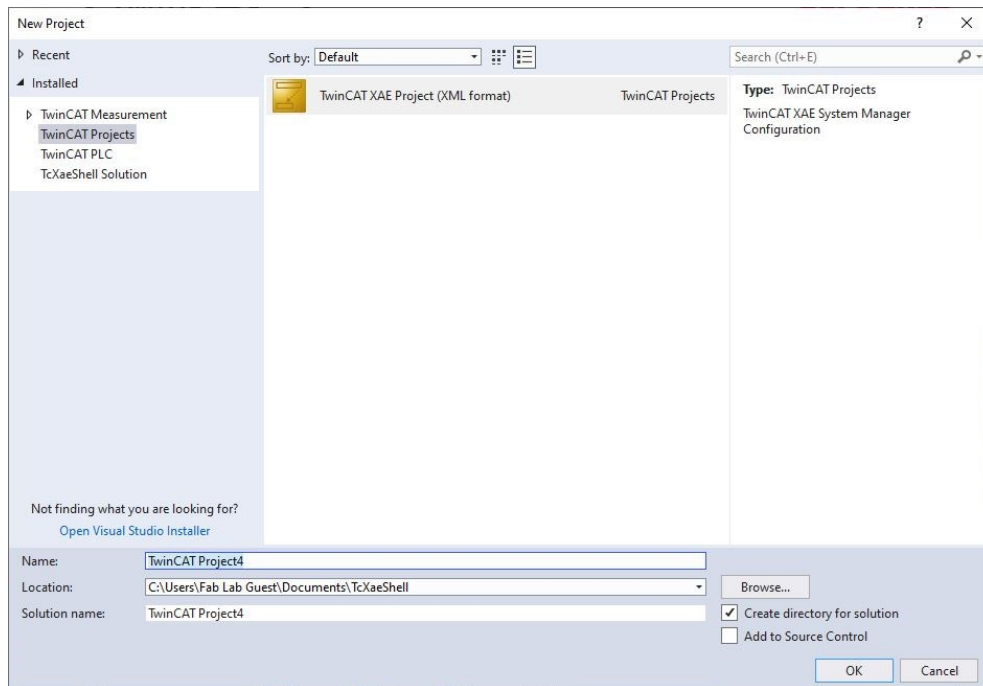
4. TWINCAT VISION KÄYTTÖÄ

4.1 Järjestelmän käyttöönotto

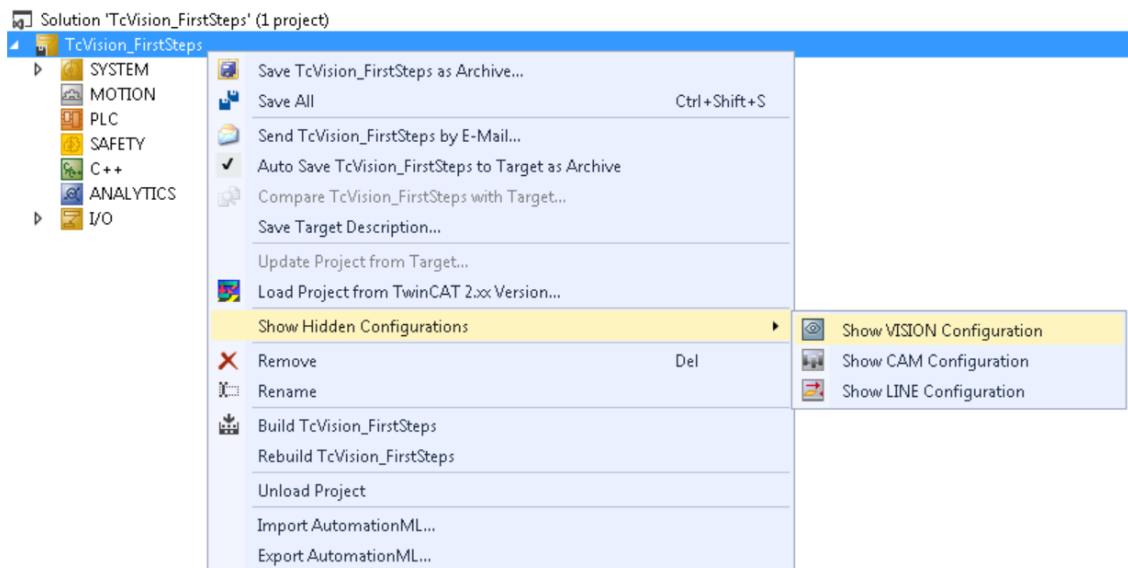
Visionin käyttö aloitetaan lataamalla Beckhoffin sivuilta TF7xxx | TwinCAT 3 Vision sovellus, joka sisältää kaikki Vision ohjelmistot sekä GigE Vision Connector -ohjelman, jonka avulla kamera yhdistetään järjestelmään. Asennuksen jälkeen voidaan aloittaa käyttö luomalla uusi projekti alla olevien kuvien mukaisesti, kuvat 12-14.



Kuva 12. Projektin luomiseksi valitaan ”New Twincat Project”.



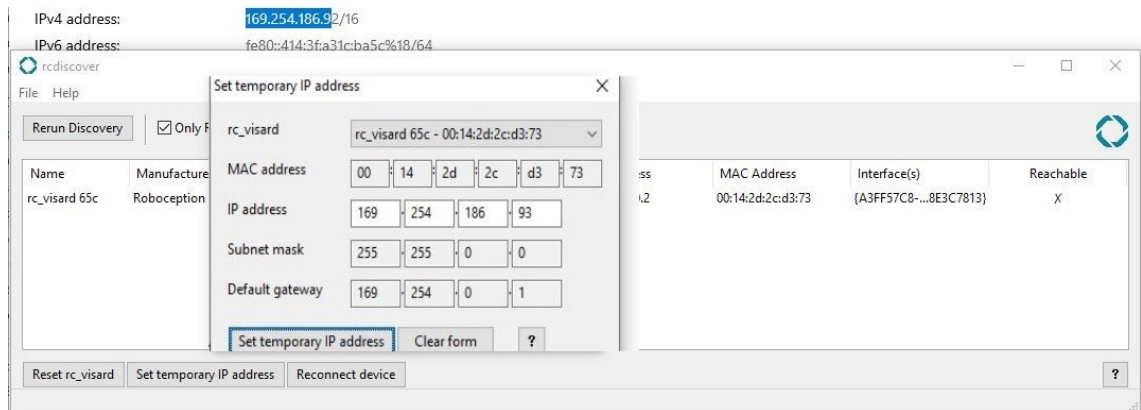
Kuva 13. Seuraavaksi valitaan ”TwinCAT XAE Project”.



Kuva 14. Projektin luomisen jälkeen asetetaan Vision lisäosa näkyväksi.

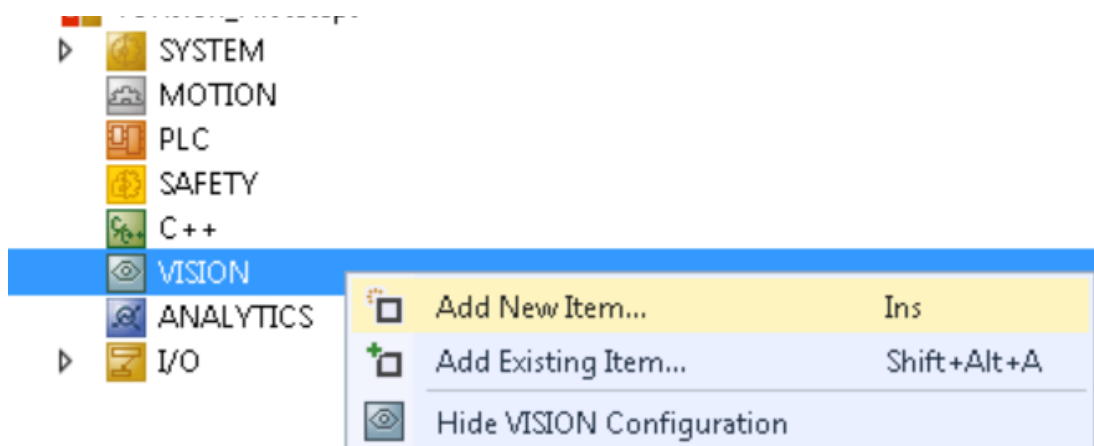
Näiden vaiheiden jälkeen Vision ohjelma on valmis käytettäväksi. Tämän jälkeen seuraava vaihe on kameran IP-osoitteen muuttaminen yhteensopivaksi tietokoneen

kanssa. Käytännössä kameran IP-osoite täytyy olla viimeistä lukua vaille sama kuin tietokoneen. Omassa käytössäni oli 3D-kamera: RC_Wizard 65c, jonka omaa ohjelmaa käyttämällä kameralle asetettiin väliaikainen IP-osoite alla olevan kuvan mukaisesti, kuva 15. Tästä vaiheesta eteenpäin täytyy tietokoneen olla kytkettynä irti internetistä, jottei IP-osoite pääse muuttumaan.

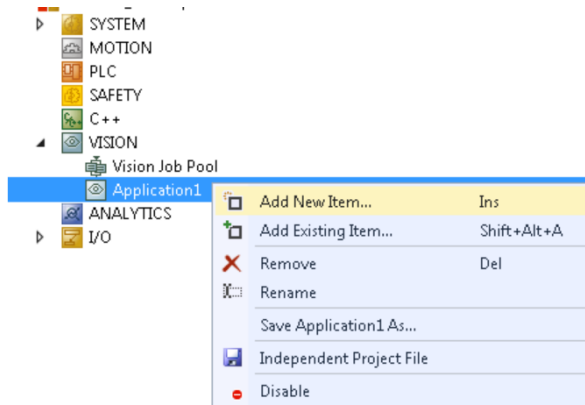


Kuva 15. Kameralle asetetaan IP-osoite, joka vastaa viimeistä lukua vaille tietokoneen IPv4-osoitetta. Tässä tapauksessa viimeisiksi luvuksi valittiin loogisesti 92:sta seuraava luku 93.

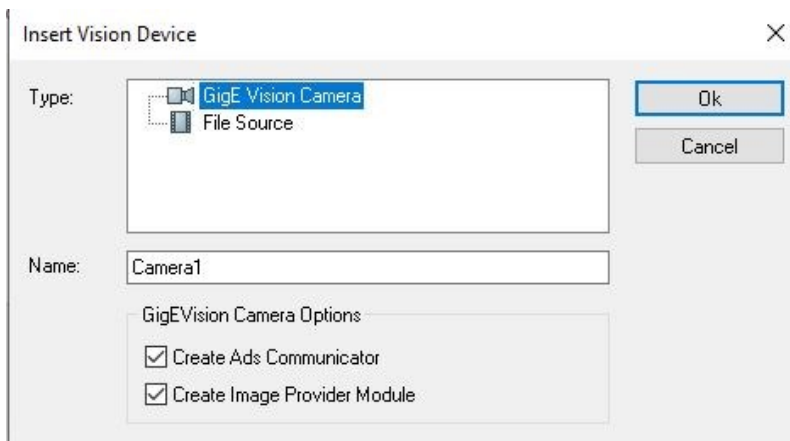
IP-osoitteen vaihtamisen jälkeen kamera on valmis yhdistettäväksi Vision ohjelmassa. Yhdistäminen tapahtuu alla olevien kuvien mukaisesti, kuvat 16–23.



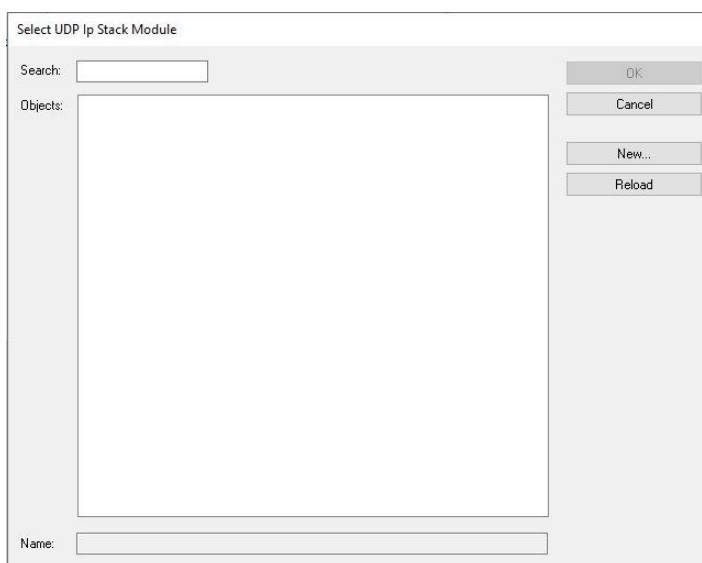
Kuva 16. Vision palkin kohdalla klikkaamalla oikeaa hiirtä saadaan näkyviin valikko, josta valitaan ”Add New Item”.



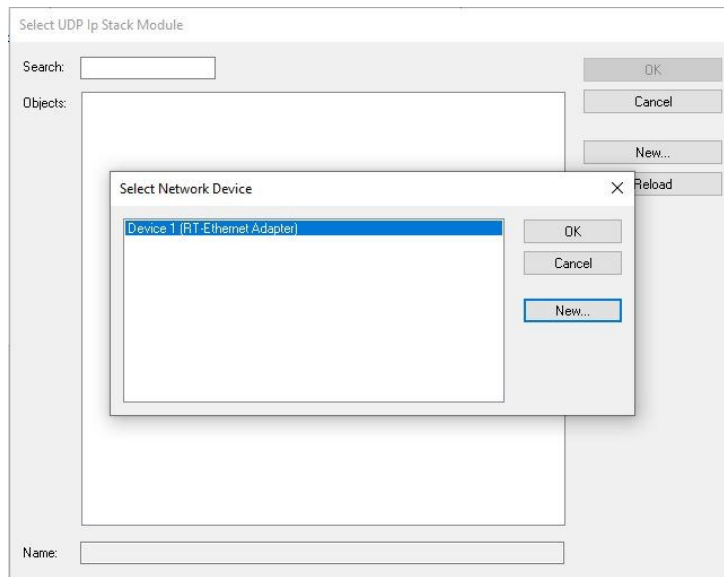
Kuva 17. Seuraavaksi klikataan luotua kansiota ja valitaan ”Add New Item”.



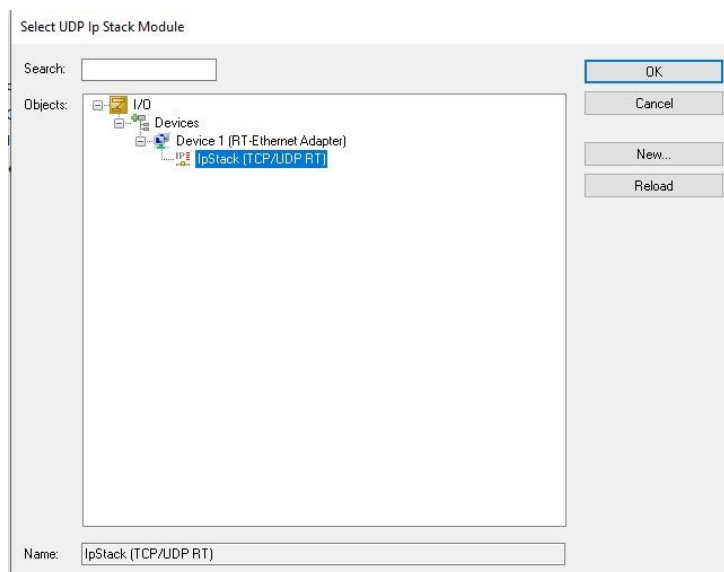
Kuva 18. Auenneesta valikosta valitaan ”GigE Vision Camera” ja painetaan ”Ok”. Jos haluttaisiin käyttää kameran sijaan lähteenä valmiita kuvia, valittaisiin ”File Source”.



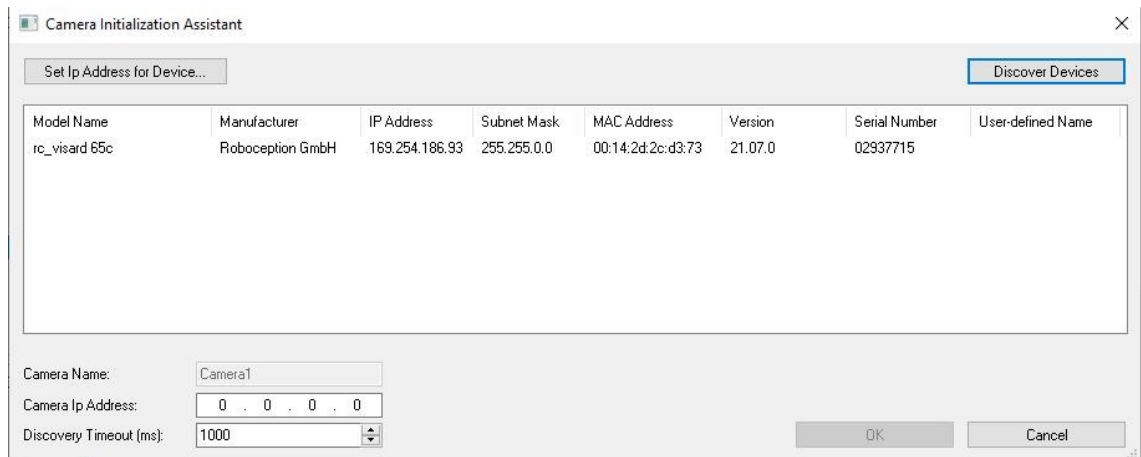
Kuva 19. Seuraavasta valikosta valitaan ”New”.



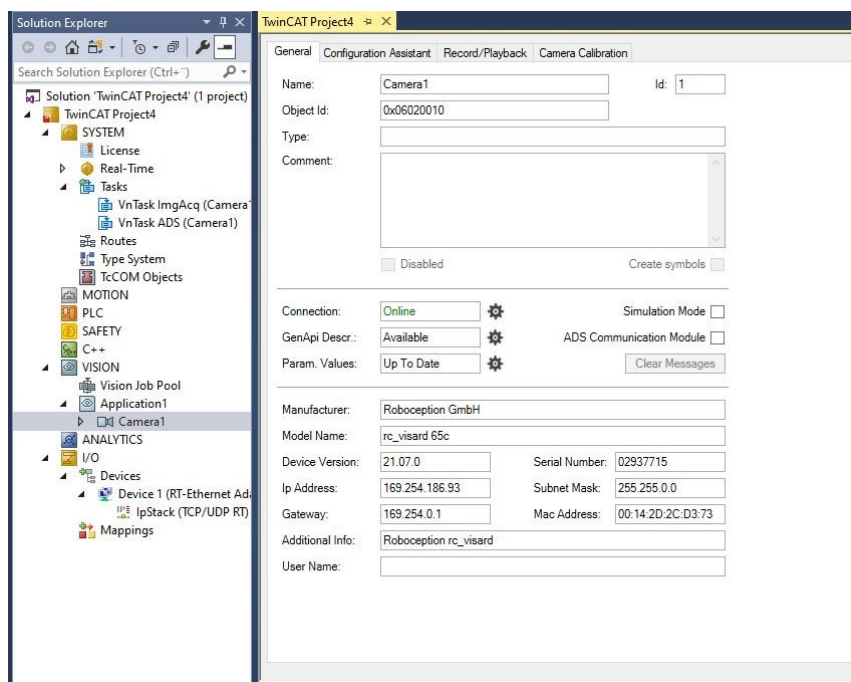
Kuva 20. Seuraavaksi avautuneesta valikosta valitaan ”New”, jolloin kameran liittymä tulee näkyviin. Valitaan haluttu laite ja painetaan ”Ok”.



Kuva 21. Auenneesta valikosta valitaan ”IpStack” ja painetaan ”Ok”.



Kuva 22. Seuraavassa valikossa painetaan ”Discover Devices”, jonka jälkeen Ethernet-kaapelilla yhdistetty kamera tulee näkyviin. Valitaan laite ja painetaan ”Ok”.

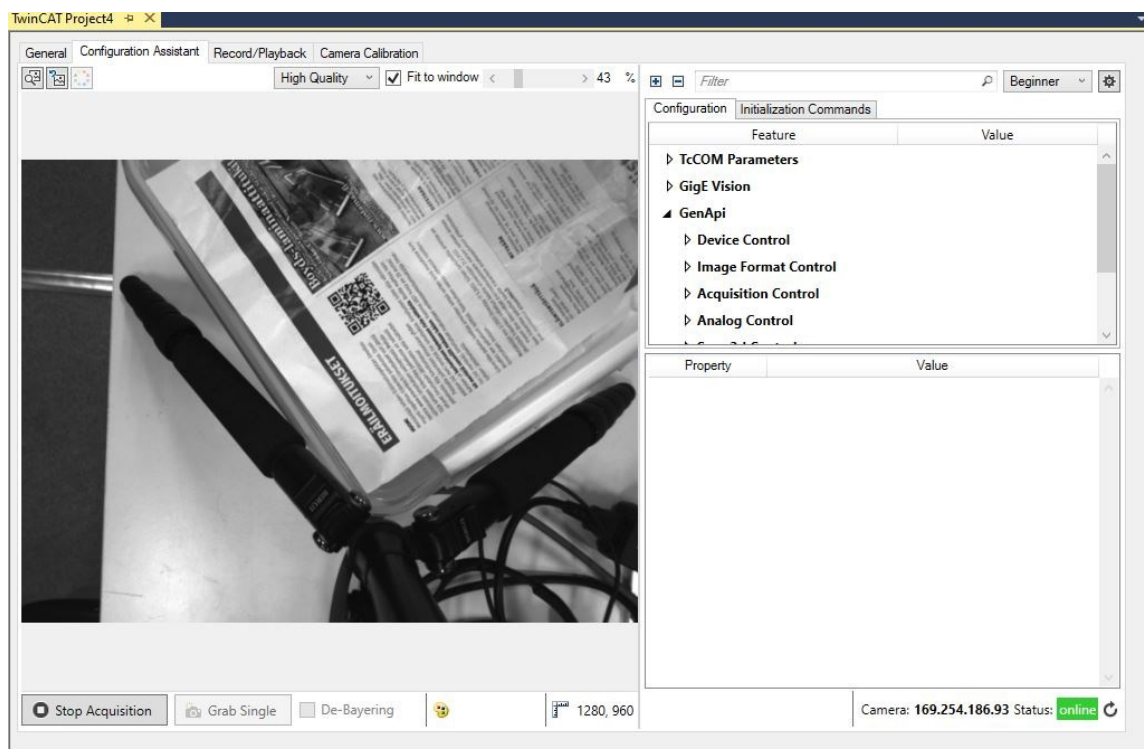


Kuva 23. Nyt kamera näkyy Application 1:n alla. Tuplaklikkaamalla kameraa pääsemme General-välilehdelle, jossa ”Connection” tilana näkyy vihreällä ”Online”.

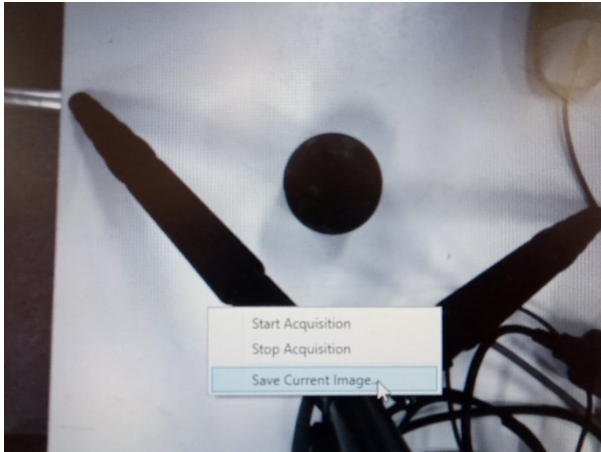
Näiden vaiheiden jälkeen kamera on yhdistetty ja valmis käytettäväksi. Seuraavassa kappaleessa esitellään muutamaa Beckhoffin mallisovellusta kameraa hyödyntäen.

4.2 Vision mallisovelluksien esittelyä

Kandintyön laajuuden vuoksi seuraavassa osiossa esitellään vain lyhyesti muutamaa Beckhoffin sivujen kautta ladattua esimerkkitsovellusta. Sovelluksiin ei tehdä muutoksia, vaan niitä käytetään sellaisinaan. Tämä tarkoittaa sitä, ettei esimerkeissä hyödynnetä live-kuvan mahdollisuutta, vaan näissä käytetään otettuja- sekä mallikuvia tiedostolähteenä. Malleissa suoran kuvayhteyden käyttäminen vaatisi koodiin perehtymistä ja tämän muokkaamista. Työssä ei myöskään luoda omaa sovellusta, jottei työn laajuus kasvaisi liian suureksi. Oman sovelluksen luominen onnistuisi kuitenkin käyttämällä Beckhoffin sivuilla olevaa koodikirjastoa, joista haluttu ohjelma rakennettaisiin. Testikuvien ottaminen tapahtuu alla olevien kuvien mukaisesti, kuvat 24–25.

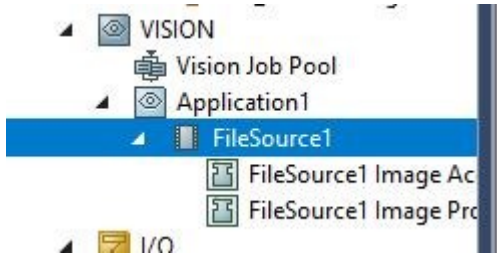


Kuva 24. Kuvayhteyden saamiseksi mennään kuvan mukaisesti kameran ”configuration assistant” välilehdelle ja valitaan ”start acquisition”.

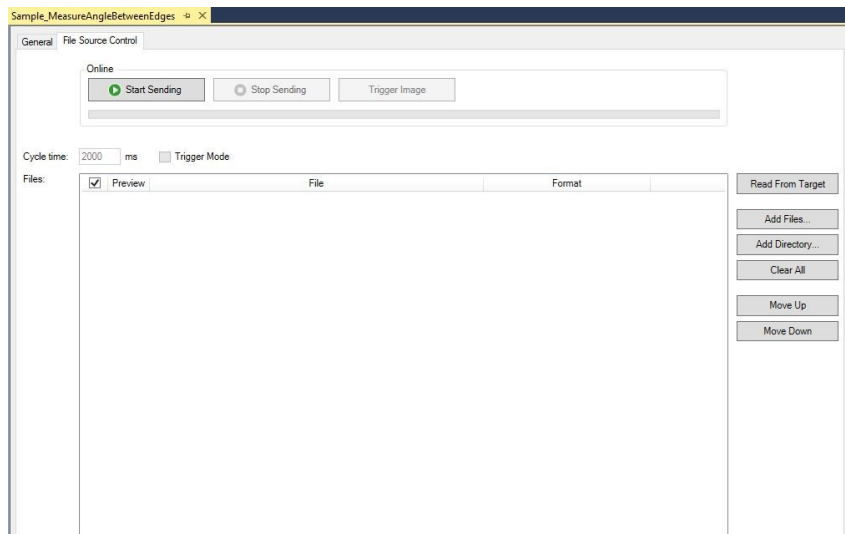


Kuva 25. Seuraavaksi painetaan kuvan kohdalla hiiren oikeaa näppäintä ja valitaan ”save current image”, jolloin otettu kuva voidaan tallentaa haluttuun sijaintiin.

Kuvien ottamisen jälkeen voidaan sovelluksien testaaminen aloittaa. Seuraavat kuvat näyttävät, kuinka kuva valitaan lähteeksi sekä minkälaisia tuloksia testikuvilla saavutettiin, kuvat 26-33.



Kuva 26. Halutun sovelluksen avaamisen jälkeen kaksoisklikataan ”FileSource” kohdalta.

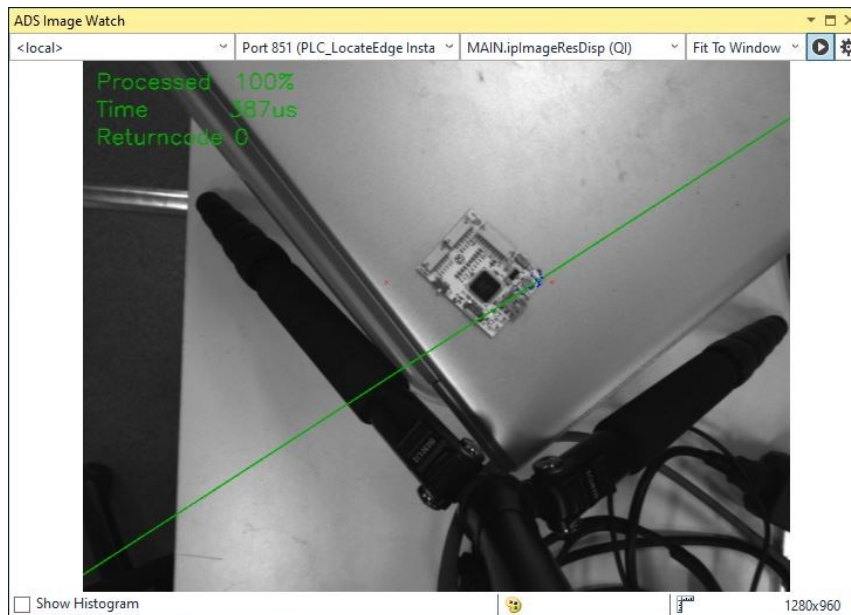


Kuva 27. Tämän jälkeen päästään tiedoston valinta välilehdelle, josta voimme lisätä haluttuja kuvia lähteiksi.

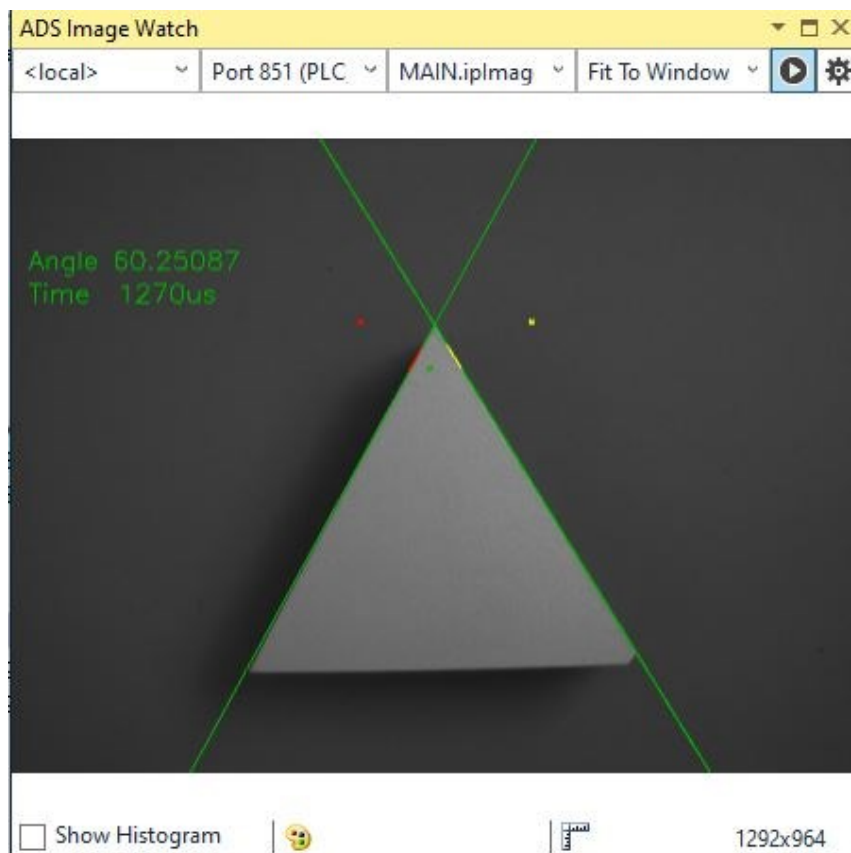
Kuvalähteiden lisäyksen jälkeen voidaan sovellus asettaa aktiiviseksi, jolloin kuvien prosessointi voidaan aloittaa. Tulosten katsomiseksi avataan ”ADS Image Watch”. Tämä löytyy seuraavan polun kautta: TwinCAT > Windows > ADS Image Watch.



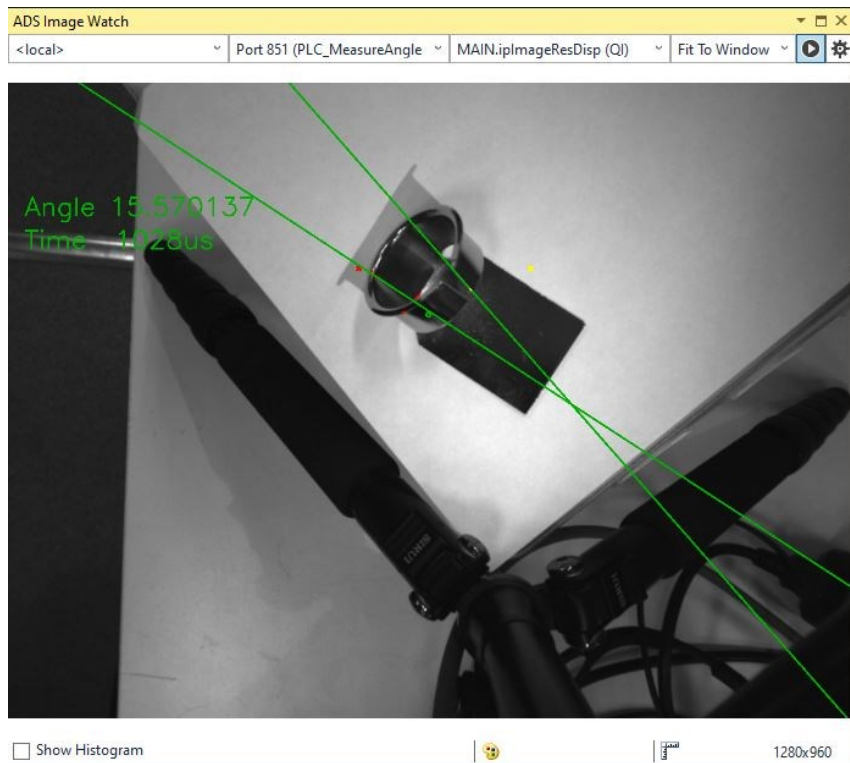
Kuva 28. Kuvassa onnistunut reunantunnistus piirilevyn selkäpuolelta.



Kuva 29. Kuvassa epäonnistunut reunantunnistus saman piirilevyn toiselta puolelta. Luultavimmin piirilevyn komponentit aiheuttavat virheellisen tunnistuksen.



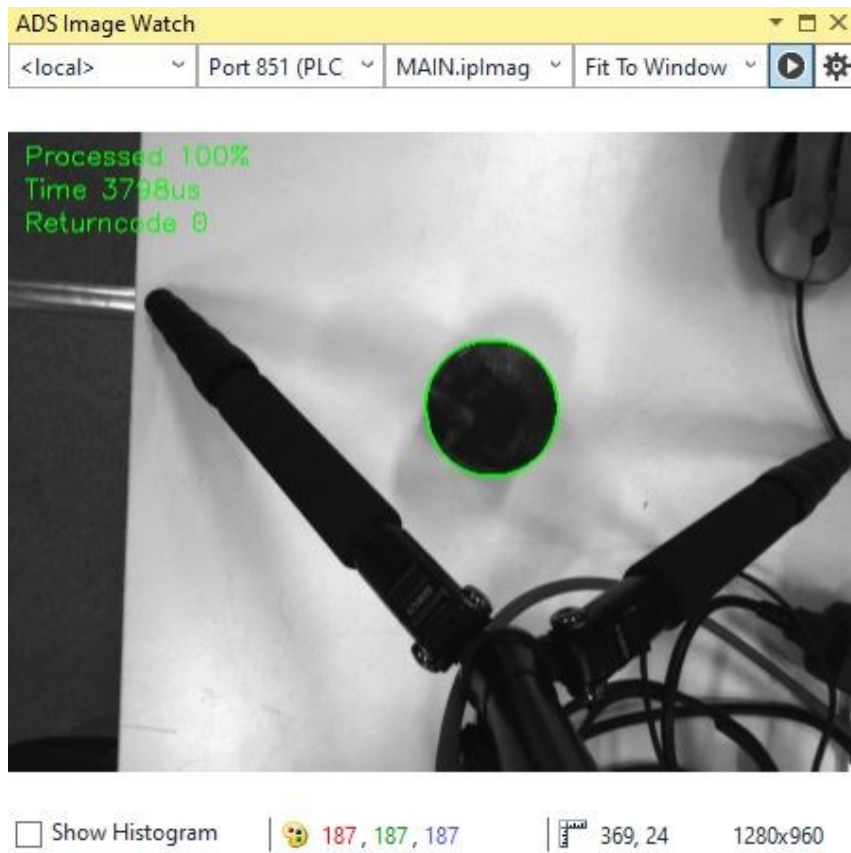
Kuva 30. Kulmantunnistus esimerkkikuvalla.



Kuva 31. Virheellinen kulmantunnistus. Tämän aiheuttaa todennäköisesti kiiltävän pinnan heijastukset sekä heikko valaistus.



Kuva 32. Kuvassa onnistunut QR-koodin lukeminen.



Kuva 33. Onnistunut esineen havaitseminen. Männän ja taustan välinen selkeä kontrasti edesauttaa onnistunutta havaitsemista.

Suoritettujen kokeilujen perusteella huomasi hyvin, kuinka kuvaolosuhteet vaikuttavat kuvantunnistuksen suorituskykyyn. Suurimpana haasteena omissa kokeissani huomasin kiiltävät esineet ja puutteellisen valaistuksen aiheuttamat varjot. Havaitsemani ongelmat ovat aiemman teoriaosion haasteiden mukaisia. Onnistuneen tunnistuksen saavuttamiseksi tarvitaan siis selkeä tausta, jossa vain käsiteltävä esine näkyy ja kontrasti on riittävän suuri. Lisäksi hyvä ja tasainen valaistus on avaintekijä. Valaistus ei saa olla toispuoleinen, jottei varjokohtia pääsisi muodostumaan.

5. POHDINTAA

On ilmeistä, että konenäköä hyödynnetään tällä hetkellä laajalti. Konenäön sovellutukset tulevat kuitenkin kehittymään ja uskon niiden korvaavaan perinteisiä prosesseja kasvavissa määrin. Teknologiaa kehitetään jatkuvasti ja etenkin teollisuudessa tälle on kysyntää. Ei ole siis ihme, että moni yritys kehittää omia järjestelmiään konenäön saralla. Tällä hetkellä teollisuudessa konenäköä hyödynnetään robotiikan ohjauksessa, tuotannon tarkastuksessa sekä ohjauksessa. Konenäön lisääminen on kallis, mutta monesti ilmeisen kannattava investointi. Järjestelmän tarjoajan valinnassa kannattaa pohtia, minkä kaltainen tämänhetkinen tarve on nyt ja tarvitaanko tulevaisuudessa mahdollisesti toisen kaltaista sovellutusta. Tämä määrittää integroinnin tarpeen. Jos esimerkiksi nyt ja tulevaisuudessa on tarvetta vain tietyn tyyppiselle sovellukselle, on usein järkevää investoida yritykseen, jonka tarjonta erikoistuu kyseiseen sektoriin.

Konetekniikan opiskelijana konenäköä ei omilla opinnoillani ollut käsitelty, mutta näkisin tämän olevan tarpeellinen aihe käsiteltäväksi esimerkiksi valinnaisissa opinnoissa. Aiheen pinnallinenkin tietämys auttaa esimerkiksi tuotantolinjoja suunniteltaessa ja ratkaisujen pohdinnassa. Kirjallisuudessa suomalaisten kirjojen puolesta tarjonta on olematon ja jouduinkin tässä työssä tukeutumaan eksklusiivisesti englanninkieliseen materiaaliin. Myös Beckhoffin Vision ohjelmiston käytössä tieto rajoittuu valmistajan englanninkieliseen oppaaseen. Opas on mielestäni anniltaan rajallinen ja ainoana netissä saatavana olevana tiedonlähteenä tämä vaikuttaa ohjelmiston opetteluun sekä sen hyödyntämiseen.

6. LÄHTEET

- ABB, 2022. *Integrated Vision - ABB Robotics - ABB Robotics - Vision systems (Robot Equipment and Accessories | ABB Robotics)* [online]. Available from: <https://new.abb.com/products/robotics/application-equipment-and-accessories/vision-systems/integrated-vision> [Accessed 11 Feb 2022].
- Beckhoff, 2022. *TwinCAT Vision | Beckhoff USA* [online]. Available from: <https://www.beckhoff.com/en-us/products/automation/twincat-vision/> [Accessed 15 Feb 2022].
- Davies, E. R., 2012. Computer and Machine Vision. *Computer and Machine Vision*.
- Dechow, D., 2021. *Machine Vision Systems Integration: Deep Learning | Quality Magazine* [online]. Quality Magazine. Available from: <https://www.qualitymag.com/articles/96663-machine-vision-systems-integration-deep-learning> [Accessed 9 Feb 2022].
- Demant, C., Streicher-Abel, B. and Garnica, C., 2013. Industrial Image Processing. *Industrial Image Processing*.
- Funt, B. v. and Finlayson, G. D., 1995. Color Constant Color Indexing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17 (5), 522–529.
- Hitachi, 2022. *Marking System : Vision System : Hitachi Industrial Equipment Systems* [online]. Available from: https://www.hitachi-ies.co.jp/english/products/vision_system/index.htm [Accessed 14 Feb 2022].
- IDEC, 2022. *IDEC - Vision* [online]. Available from: <https://www.mfp99.com/products/idec/sensing/vision/> [Accessed 14 Feb 2022].
- imagingknowledgebin.blogspot, 2022. *understanding signals and photographs: exploring image types and formats* [online]. Available from:

<http://imagingknowledgebin.blogspot.com/2013/06/exploring-mage-types-and-formats.html> [Accessed 10 Feb 2022].

Jain, R., Kasturi, R. and Schunck, B. G., 1995. Machine vision, 549.

KEYENCE, 2022. *Vision Systems | KEYENCE America* [online]. Available from: <https://www.keyence.com/products/vision/vision-sys/> [Accessed 11 Feb 2022].

Labudzki, R., Legutko, S. and Raos, P., 2014. The essence and applications of machine vision. [online]. Available from: <https://www.researchgate.net/publication/286283684> [Accessed 2 Feb 2022].

Omron, 2022. *Machine Vision | Omron* [online]. Available from: <https://automation.omron.com/en/us/products/category/machine-vision> [Accessed 14 Feb 2022].

Ravina, M., Supriya, I. and Nilam, D., 2013. Optical Character Recognition . *International Journal of Recent Technology and Engineering (IJRTE)* [online], 2 (1). Available from: <http://code.google.com/p/tesseract-ocr>. [Accessed 2 Feb 2022].

Siemens, 2022. *Image-Based Barcode Reading Fulfills its Promise | Features | Vision Spectra* [online]. Available from: https://www.photonics.com/Articles/Image-Based_Barcode_Reading_Fulfills_its_Promise/a62890 [Accessed 10 Feb 2022].

Siemens, 2022. *Optical identification - Industry Mall - Siemens WW* [online]. Available from: <https://mall.industry.siemens.com/mall/en/WW/Catalog/Products/10007478?tree=CatalogTree> [Accessed 10 Feb 2022].

Silver, B., 2017. *Image-Based Barcode Reading Fulfills its Promise | Features | Vision Spectra* [online]. Vision-Spectra.com. Available from: https://www.photonics.com/Articles/Image-Based_Barcode_Reading_Fulfills_its_Promise/a62890 [Accessed 9 Feb 2022].

Voulodimos, A., Doulamis, N., Doulamis, A. and Protopapadakis, E., 2018. Deep Learning for Computer Vision: A Brief Review. *Computational Intelligence and Neuroscience*, 2018.

Wilson, A., 2019. *Deep learning brings a new dimension to machine vision* | *Laser Focus World* [online]. LaserFocusWorld.com. Available from: <https://www.laserfocusworld.com/home/article/16556323/deep-learning-brings-a-new-dimension-to-machine-vision> [Accessed 9 Feb 2022].