

Arto Martikainen

KAMERATEKNIikkaAN PERUSTUVA TÄRINÄMITTAUS

Insinöörityö
Kajaanin ammattikorkeakoulu
Tekniikka ja liikenne
Tietotekniikka
Kevät 2014



Koulutusala Tekniikka ja liikenne	Koulutusohjelma Tietotekniikka
Tekijä(t) Arto Martikainen	
Työn nimi Kameratekniikkaan perustuva tärinämittaus	
Vaihtoehtoiset ammattipinnot Tietoturva	Toimeksiantaja Pentti Romppainen / Kajaanin ammattikorkeakoulu
Aika Kevät 2014	Sivumäärä ja liitteet 38
<p>Tämän insinöörityön tavoitteena oli ottaa käyttöön Kajaanin ammattikorkeakoululle hankittu Basler acA2000-50gm -kamera ja kehittää konenäköjärjestelmä, jonka tehtävä oli asematiedon muodostaminen kuvatiedosta. Sen erotuskyky suunniteltiin olevan mikrometriluokkaa. Järjestelmän tehtävänä oli myös mittausdatan tulkitseminen ja sen esittäminen SI-järjestelmän yksiköissä. Lopuksi kokonaisuus oli tarkoitus integroida osaksi laajempaa mittausjärjestelmää.</p> <p>Asematiedon mittaamiseksi käytetään perinteisesti antureita, jotka kiinnitetään kohteeseen. Joissakin tapauksissa, kuten resonanssi-ilmiöiden tarkastelussa, tämä lähestymistapa ei ole käyttökelpoinen, sillä esimerkiksi anturin paino voi muuttaa kohteen resonanssikäyttäytymistä. Tämän vuoksi tarvitaan mittausmenetelmiä, jotka toimivat kohdetta koskettamatta.</p> <p>Ohjelmointityössä käytettiin National Instrumentsin LabVIEW-ohjelmointiympäristöä ja sen konenäkömoduulia. Sillä rakennettiin ohjelmisto, joka konfiguroi kamerasuorituksen, suorittaa mittauksen, analysoi mittaustulokset sekä esittää ne. Kaksi erillistä algoritmia kehitettiin mittausdatan analysoimista varten. Järjestelmän testaamisessa käytettiin metallilevyä, jossa oli kaksi kapeaa rakoja. Valaistukseksi valittiin taustavalaistus.</p>	
Kieli	Suomi
Asiasanat	Konenäkö, digitaalinen kuvankäsittely, tärinä, kamera, LabVIEW
Säilytyspaikka	<input type="checkbox"/> Verkkokirjasto Theseus <input type="checkbox"/> Kajaanin ammattikorkeakoulun kirjasto



School Engineering	Degree Programme Information Technology
Author(s) Arto Martikainen	
Title Vibration Measurement Based on Camera Technology	
Optional Professional Studies Information security	Commissioned by Pentti Romppainen / Kajaanin ammattikorkeakoulu
Date Spring 2014	Total Number of Pages and Appendices 38
<p>The purpose of this Bachelor's thesis was to implement a Basler acA2000-50gm camera and to develop a machine vision system that acquires positional information from an image. The resolution of the system was intended to be in the range of micrometers. A method of converting the measurement data to the metric system was to be developed. Finally, the system was intended to be integrated into a larger measurement system.</p> <p>To measure the positional information, a sensor is typically attached to the object. On some occasions, e.g. when measuring resonance phenomena, this approach is not feasible because the weight of the sensor may change the objects' resonating characteristics. Thus, there is a need for a method that measures positional information of an object without touching it.</p> <p>The development of the system was done using National Instrument's LabVIEW and its machine vision module. It was used to build a set of functions that configures the camera, acquires the measurement and analyzes the measurement data. Two separate algorithms were developed to analyze the measurement data.</p>	
Language of Thesis	Finnish
Keywords	Machine vision, digital image processing, vibration, camera, LabVIEW
Deposited at	<input type="checkbox"/> Electronic library Theseus <input type="checkbox"/> Library of Kajaani University of Applied Sciences

ALKUSANAT

Haluan kiittää Pentti Romppaista työn ohjaamisesta ja neuvoista, Jussi Ala-Hiiron erinomaisista neuvoista LabVIEW-kehitysympäristön käytössä, ystävääni Anssi Mustosta vertaistuesta sekä muita ystäviäni, jotka tukivat ja kannustivat minua työn aikana.

SISÄLLYS

1 JOHDANTO	1
2 KONENÄKÖ	2
2.1 Konenäköjärjestelmän yleinen rakenne	2
2.2 Kuvailmaisin	3
2.3 Kameraliitännät	6
2.4 Optiikka	9
2.5 Valaistus	10
2.6 Digitaalinen kuvankäsittely	14
2.6.1 Kuvankäsittelymenetelmiä	15
2.6.2 Kuvankäsittelyohjelmistot	16
3 MITTAUSJÄRJESTELMÄN KUVAUS	18
3.1 Kamera	18
3.2 Optiikka	19
3.3 Valaisu	20
3.4 PC-ympäristö	20
4 OHJELMOINTIYMPÄRISTÖN KUVAUS	21
4.1 LabVIEW	21
4.2 Vision Assistant	24
5 OHJELMISTON TOTEUTUS	25
5.1 Työlle asetetut vaatimukset	25
5.2 Kameran konfigurointi	26
5.3 Kalibrointi	27
5.4 Mittaus	27
5.5 Analysointi	28
5.6 Tulosten esittäminen	29
5.7 Ohjelmakoodin testaus	30
6 JÄRJESTELMÄN SUORITUSKYVYN ANALYSOINTI	32
7 YHTEENVETO	35

LÄHTEET

36

LIITTEET

LYHENTEET JA MÄÄRITELMÄT

LabVIEW	Laboratory Virtual Instrumentation Engineering Workbench. National Instrumentsin kehittämä ja ylläpitämä kaupallinen ohjelmointiympäristö.
CCD	Charge Coupled Device, kameran ilmaisimien valmistustekniikka
CMOS	Complementary Metal Oxide Semiconductor, kameran ilmaisimien valmistustekniikka
Jumbo Frame	Yli 1500 ja alle 9000 tavun kokoinen Ethernet-paketti.
GenICam	Generic interface for Cameras. Standardi, joka määrittelee yleisen rajapinnan teollisuuskameroiden käyttämiseksi.

1 JOHDANTO

Kajaanin ammattikorkeakoulu on perustettu vuonna 1992, ja siellä voi suorittaa sairaanhoitajan, terveydenhoitajan, liikunnanohjaajan, insinöörin, restonomin tai tradenomin tutkinnon. Vuosittain oppilaitoksessa aloittaa 400 uutta perustutkinto-opiskelijaa ja 35 ylempään tutkinnon opiskelijaa.

Opetuksen ohella Kajaanin ammattikorkeakoulu ja erityisesti sen tekniikan yksikkö on aktiivinen toimija myös tutkimus- ja kehitystyössä. Yksi tähän toimintaan liittyvä alue on ollut vuonna 2006 aloitettu kehotärinän mittaukseen ja tärinän vaimennusratkaisuihin liittyvä tutkimus- ja kehitystyö, jota aiemmin on tehty TEKESin ja teollisuuden rahoittamissa hankkeissa TÄRVIÖ ja MEWEX

Viimeisin hanke, Tarra, alkoi vuonna 2012, ja sen aiheena oli mm. tärinänvaimennusratkaisujen tutkiminen. Hankkeessa oli yksi case-tutkimus, jonka kohteena oli puoliaktiivisen traktoriohjaamon vaimennusjärjestelmän testaaminen. Sen kannalta olennaista oli liiketiedon mittaaminen yleisesti käytettävien kiihtyvyyssantureiden avulla. Toinen vaihtoehto on käyttää antureita, jotka mittaavat kappaleiden tai niiden osien asemaan perustuvaa liiketietoa. Näillä menetelmillä on se etu, että niiden avulla nähdään suoraan esimerkiksi resonanssikäyttäytyminen. Asematiedon mittaaminen monilla perinteisillä menetelmillä on ongelmallista, sillä esimerkiksi resonoivaan kappaleeseen kiinnitetty mittalaite voi muuttaa mitattavan kappaleen käyttäytymistä. Kameratekniikalla on mahdollista päästä mittaamaan kohdetta koskettamatta.

Työn aiheena oli ottaa käyttöön Kajaanin AMK:lle hankittu kamera ja kehittää järjestelmä, jonka avulla voidaan määrittää kappaleen asematietoa konenäköön perustuen. Lisäksi tavoitteena oli viedä se mittausmenetelmäksi, joka voidaan integroida osaksi laajempaa mittausjärjestelmää.

2 KONENÄKÖ

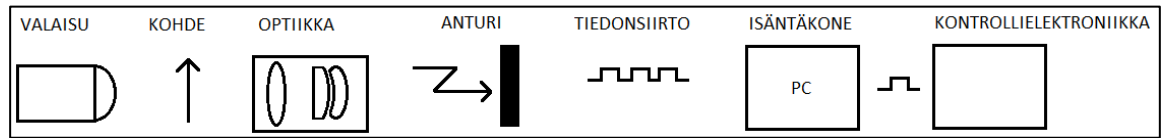
Konenäkö on tietojenkäsittelytieteen osa-alue, josta on viimeisen 20 vuoden aikana tullut tärkeä työkalu teollisessa valmistuksessa. Konenäön perusteet kehitettiin 50- ja 60-luvuilla. Tällöin tutkimusta suoritettiin vain teoreettisella tasolla, sillä digitaaliset kuvailmaisimet kehitettiin vasta 70-luvulla. 80-luvulla konenäköä alettiin tutkimaan kiihtyvässä tahdissa ja tänä aikana kehitettiin uusia teorioita ja konsepteja. Älykkäät kamerat kehitettiin 80-luvun lopulla, mikä johti uusiin sovelluksiin ja tätä myöten laajentuneeseen käyttöön. [1.]

90-luvulla konenäön hyödyntäminen alkoi yleistyä teollisessa valmistuksessa, mikä johti konenäköteollisuuden syntyymiseen. LED-valaisun, anturitekniikan ja hallinta-arkkitehtuurin kehittyminen paransivat konenäköjärjestelmien kykyä. Laajemman käytön myötä konenäköjärjestelmien hinnat alkoivat laskea. Nykyään konenäköjärjestelmät ovat teollisuuden alalla yleisiä, ja tehokkaan tietotekniikan ansiosta tuotteita voidaan kuvata kolmiulotteisena ja suurilla nopeuksilla. [1.]

Tavallisimpia konenäön sovelluksia ovat automaattisen kokoonpanon ja laadunvalvonnan lisäksi liikenteenvalvonta ja -ohjaus, tekstintunnistus sekä biometriset tunnistusmenetelmät, kuten kasvojentunnistus ja sormenjälkitunnistus. Konenäköä käytetään myös korkean teknologian kohteissa, kuten esimerkiksi komeettaluotain Rosettassa, joka otti kuvan tähtitaivaasta herättyään 31 kuukauden horroksesta. Kuvan perusteella luotain käänsi asentoaan siten, että sen antennit osoittivat kohti maata, jonka jälkeen viimeinen vaihe luotaimen matkassa saattoi alkaa. [2.]

2.1 Konenäköjärjestelmän yleinen rakenne

Konenäköjärjestelmä rakentuu seitsemästä osasta: kuvattava kohde, valaisu, optiikka, ilmaisimet, siirtotie, isäntäkone ja kontrollielektroniikka. Näistä kontrollielektroniikka on tarpeellinen vain silloin, kun halutaan käsitellä kuvattavaa kohdetta analyysin jälkeen. Tätä voi olla esimerkiksi viallisen tuotteen poistaminen tuotantolinjalta. Kuva 1 havainnollistaa konenäköjärjestelmän rakennetta. [3.]



Kuva 1. Konenäköjärjestelmän rakenne

Analysointi tapahtuu digitaalisen kuvankäsittelyn avulla, ja sen tuloksena saadaan esimerkiksi paikkatieto esineestä, esineen fyysisiin ominaisuuksiin liittyvät tiedot tai kuvassa oleva teksti- tai numerotieto. [3.]

Isäntäkoneessa toimii konenäköohjelmisto, joka kontrolloi kuvan muodostusta, analysoi kuvat sekä mahdollisesti ohjaa kontrollielektronikkaa. Ohjelmisto suorittaa digitaalisen kuvankäsittelyn sekä kuvan pohjalta tehtävät mittaukset. Sillä voi olla erilaisia tehtäviä, kuten esimerkiksi esineen paikannus, jotta robottikäsi voi nostaa sen, esineen fyysisten ulottuvuuksien, kuten etäisyyden, halkaisijan, kaarevuuden, pinta-alan, korkeuden tai tilavuuden määrittäminen; esineen tiettyjen ominaisuuksien tarkastus; tai teksti- tai numerotiedon lukeminen. [3.]

Kuvaa voidaan ajatella kaksiulotteisena diskreettinä funktiona $f(x, y)$, missä x ja y ovat 2D-koordinaatteja. Funktion $f(x,y)$ arvo harmaasävykuvassa on luminanssi tai värikuvassa esimerkiksi sen RGB-arvo. Yhtä tällaista kuvaelementtiä kutsutaan pikseliksi, ja niiden määrä eri 2D-suunnissa määrää kuvan resoluution. Esimerkiksi kuva, jonka resoluutio on 1680 x 1050, sisältää n. 1,7 miljoonaa pikseliä. [3.] [4, s. 1.]

Kuvantamistekniikat eivät rajoitu pelkästään näkyvän valon aallonpituuksilla tapahtuviksi. Tehtävään käytetään tämän lisäksi muun muassa ultraviolettii- ja infrapunavaloa, lämpösäteilyä, mikro- ja radioaaltoja, röntgensäteilyä, gammasäteilyä ja elektronisuihkua. [4, s. 7-23.]

2.2 Kuvailmaisin

Digitaalisessa kuvantamisessa käytetään suorakaiteen muotoisia ilmaisimatriiseja. Puolijohteesta valmistettuun ilmaisimatriisiin, kuvakennoon, ohjataan optiikan avulla valoa, jonka määrästä tieto varastoituu matriisissa oleviin yksittäisiin ilmaisinelementteihin eli

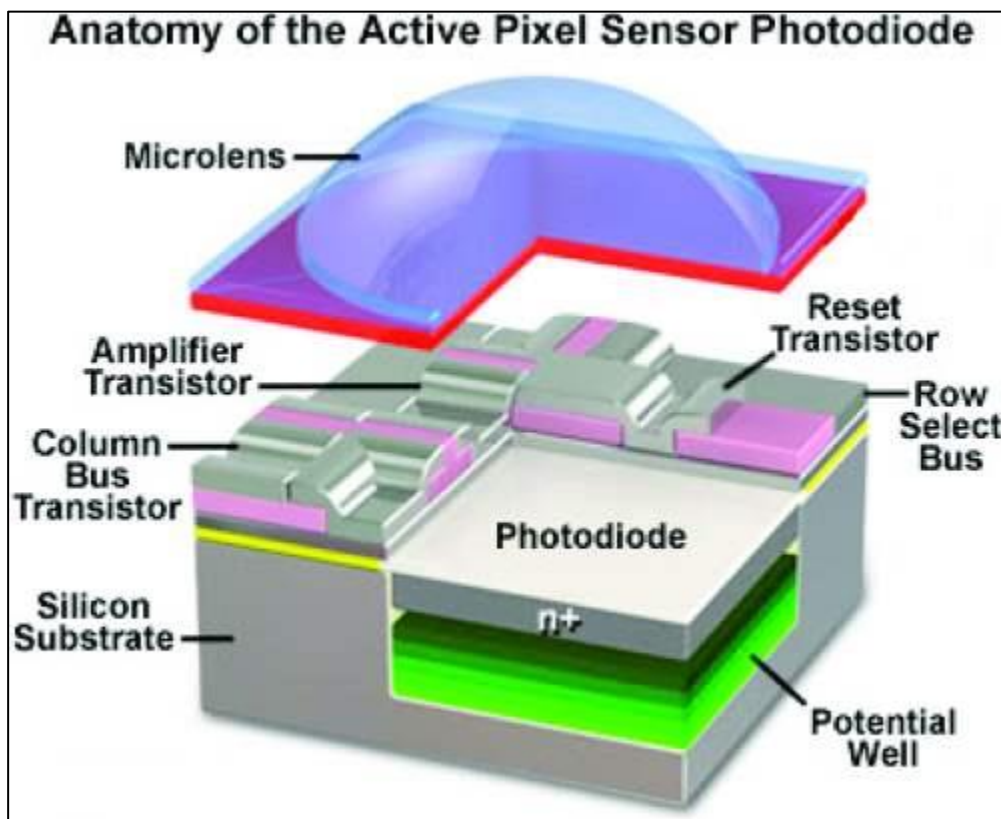
pikseleihin varauksen muodossa. Kuvakennon valolle herkkä osa toimii kuten aurinkokenno: kennoon osuva valo muuttuu sähkövarauksiksi. Se luetaan valotusajan päätyttyä. [3.] [5.]

Valon vaikutuksesta ilmaisinelementteihin muodostuu elektroni-aukkopareja. Näistä elektroneihin liittyvää varausta kerätään talteen koko valotusajan. Yksittäiseen ilmaisinelementtiin liittyvät varaukset pidetään toisistaan erillään ilmaisimen rakenteen avulla. [3.]

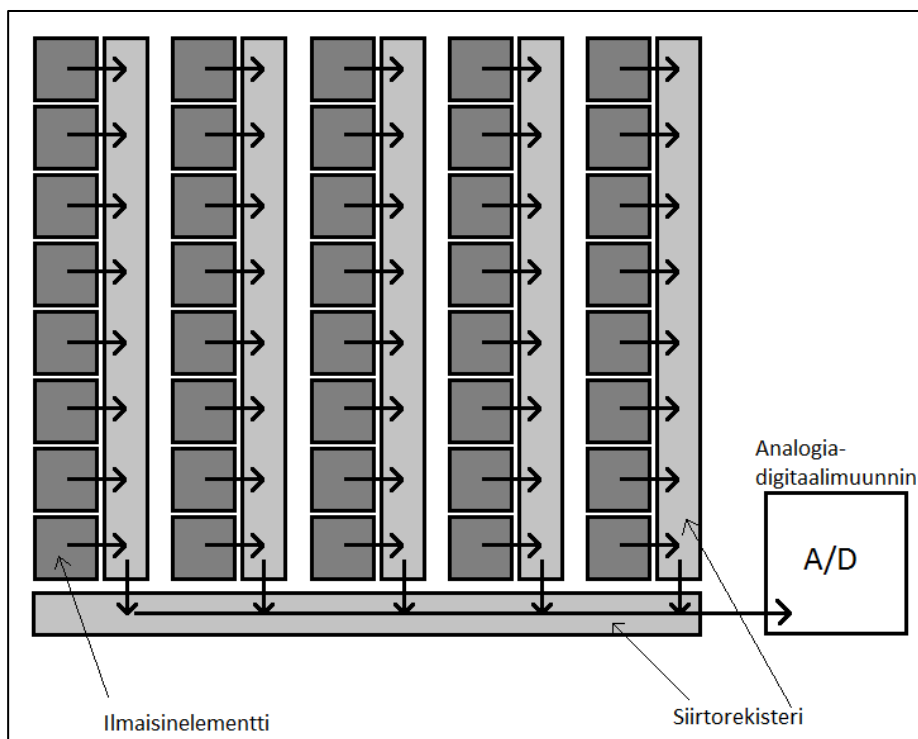
Ilmaisimia on kahta tyyppiä. Viivakennot muodostavat viivamaista kuvaa, ja 2D-kuvan muodostuminen edellyttää joko kuvattavan kohteen tai kameran liikuttamista kontrolloidulla tavalla. Matriisikennot muodostavat suoraan 2D-kuvaa. Yksittäisiä ilmaisinelementtejä voi olla jopa useita tuhansia viivakennoissa ja jopa kymmeniä miljoonia matriisikennoissa. [3.]

Kuvakennojen valmistamisessa käytetään kahta teknologiaa, CMOS (Complementary Metal Oxide Semiconductor) ja CCD (Charge-Coupled Device). Kummatkin teknologiat kehitettiin 60- ja 70-lukujen taitteessa, mutta senaikaisista puolijohdevalmistustekniikoista johtuen CCD-kennoista tuli yleisempiä. 90-luvulle tultaessa puolijohdetekniikat olivat kehittyneet niin, että CMOS-kennoista alkoi tulla käyttökelpoinen teknologia. CMOS-kennot ovat suurimmaksi osaksi syrjäyttäneet CCD:n monissa kuluttajasovelluksissa ja suurimmassa osassa konenäkösovelluksia. [3.] [6.]

Valmistustavan lisäksi näillä kennotyypeillä on myös muita eroja. CMOS-tekniikassa yksittäisen ilmaisinelementin varaus muunnetaan jännitteeksi samassa ilmaisinelementissä. Kuva 2 esittää CMOS-kennon yksittäistä ilmaisinelementtiä. CCD-tekniikassa yksittäisten ilmaisinelementtien varausta kuljetetaan siirtorekistereiden avulla samalle muuntimelle, jossa se muutetaan jännitteeksi. Kuva 3 esittää CCD-kennon siirtorekistereiden toimintaa. Tämän toimintaperiaatteellisen eron vuoksi CMOS-kennoa voidaan lukea osittain, kun taas CCD-kenno on luettava kokonaan. [3.] [5.] [7.]



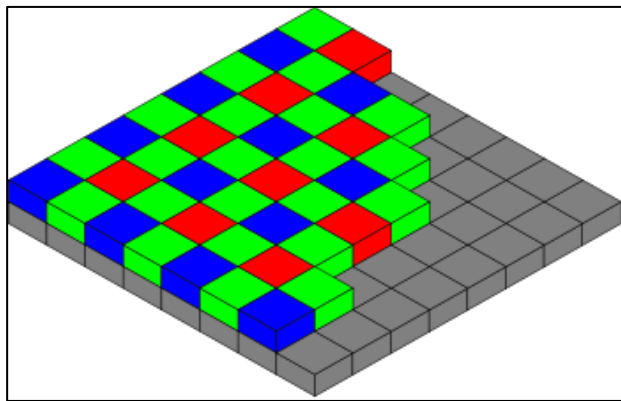
Kuva 2. CMOS-kennon yksittäisen ilmaisinelementin rakenne [5]



Kuva 3. CCD-kennon siirtorekisterit

Koska CMOS-tekniikassa jokainen ilmaisinelementti tekee oman muunnoksensa, niiden välinen yhteneväisyys on huonompi, mutta rinnakkaisuus on suurempi. Huonomman yhteneväisyyden selittää A/D-muuntimien eroavaisuudet ilmaisinelementtien välillä. Eroavaisuudet johtuvat siitä, että puolijohdeiden valmistuksessa on vaikeaa hallita prosessia niin tarkasti, että piirin jokainen puolijohdekomponentti toimisi täsmälleen samalla tavalla. Tämä aiheuttaa sen, että ilmaisimella olevat ilmaisinelementit tulkitsevat saman valomäärän eri tavalla. [5.] [6.]

Värikuvat muodostetaan käyttämällä suodattimia. Kamerassa voi olla kolme kennoa, joille värit erotellaan kennon edessä olevan suodatinsysteemin avulla. Toinen vaihtoehto on käyttää kennolla suodattimia, joista muodostuu väri-informaatio. Kuvassa 4 on esitetty eräs tällainen suodatin. Harmaasävykennoissa ei ole suodattimia. Lisäksi kennon pinnalla olevilla linseillä pyritään keräämään maksimaalisesti valoa. [5.]



Kuva 4. Bayer-suodatin [8]

2.3 Kameraliitännät

Konenäkösovelluksissa kameran ottama kuva on vietävä isäntäkoneelle käsiteltäväksi ja analysoitavaksi. On olemassa useita eri liitännävaihtoehtoja, kuinka kuvadata saadaan siirrettyksi. Liitännän valinta määräytyy käyttökohteen mukaan. Vaadittu kaistanleveys ja liitännän ennustettavuus (*deterministisyys*) ovat seikkoja, jotka monesti ratkaisevat käytettävän liitännän tyypin. Ennustettavuus tarkoittaa sitä, että liitäntä kykenee toimittamaan otetun kuvan isäntäkoneelle tietyn ajan sisällä. Myöhästynyt kuvanvälitys johtaa siihen, että kuvan käsittelyyn on vähemmän aikaa, joka taas johtaa siihen, että kuvia voi jäädä analysoimatta.

Monesti konenäkösovellukset ovat aikakriittisiä, sillä esimerkiksi teollisuudessa konenäköjärjestelmän on toimittava tuotannon nopeuden ehdoilla. [9.]

GigE Vision

Gigabit Ethernet on vanhempien Ethernet-liitäntöjen uusimpia versioita. Kaapeli on usein Cat 5 -kaapelia, sen pituus voi olla jopa 100 metriä ja sen kaistanleveys on 125 MB/s. Konenäkökäytössä liitäntää kutsutaan GigE visioniksi. Sen etuna on laajalti käytössä oleva Ethernet-standardi, jonka mukainen verkko skaalautuu helposti, ja laitteisto on edullista. Ethernet-kaapelin kautta on myös mahdollista viedä laitteeseen käyttöjännite, mutta se vaatii erillisen sovittimen. Lisäksi verkkoon voidaan liittää useampia kameroita. Ethernet-liikenne on pakettimuotoista, mikä tarkoittaa sitä, että jos verkossa on useampia laitteita, voi tiedonvälitys viivästyä, koska toinen verkossa oleva laite lähettää dataa juuri sillä hetkellä. GigE-liitäntä ei ole tämän vuoksi hyvä valinta järjestelmiin, joissa kuvan toimittaminen isäntäkoneelle täytyy tapahtua tietyn aikarajan sisällä. [9.] [10.]

Firewire

IEEE 1394 eli Firewire-standardia aloitettiin kehittämään vuonna 1986, ja se siirtyi valmistukseen 1994. Standardi on käynyt läpi useita versioita, jotka ovat parantaneet sen tiedonsiirtokapasiteettia ja tuoneet siihen uusia ominaisuuksia. Firewire-liitäntään perustuvat kamerat tukevat yleensä joko 1394a- tai 1394b-standardia, joiden tiedonsiirtonopeudet ovat 400 Mbit/s ja 800 Mbit/s. Kaapelin suurin pituus on 4,5 metriä. Optista kaapelia käytettäessä sen pituus voi olla jopa 100 metriä. [9.] [11.]

USB Vision

USB-väylää on käytetty myös konenäkösovelluksissa, sillä se on laajalle levinnyt liitäntä, jonka ansiosta PC:ssä ei tarvitse käyttää erillistä kuvankaappauskorttia. USB 2.0 Vision kilpaili aikanaan Firewire-standardin kanssa, ja sen tiedonsiirtonopeus on 480 Mbit/s. USB 3.0 Vision käyttää USB 3.0 -portteja, jotka ovat yleistymässä kuluttajätietokoneissa. Sen väylän siirtonopeus on 5 Gbit/s, ja kaapelia pitkin voidaan syöttää laitteelle käyttöjännite. [9.] [12.]

Camera Link

Camera Link on rakennettu kameranovelluksia silmälläpitäen, ja se on paljon käytetty ja vakiintunut standardi. Camera Link -väylän siirtonopeus on 850 MB/s. Uudessa 2.0 versiossa on mahdollista viedä käyttöjännite laitteeseen sekä vaihtoehtona pienempi liitin. Camera link on pisteestä pisteeseen -liitäntä, jossa kamera liitetään kuvankaappauskortin kautta järjestelmään. [9.] [13.]

CoaXPress

CoaXPress on koaksiaaliliitäntä, joka on suunniteltu kuvan siirtoa varten. Sen tiedonsiirtonopeus yhdellä kaapelilla on 6,25 Gbit/s, joskin järjestelmään voidaan liittää neljä kaapelia, joiden avulla saavutetaan 20 Gbit/s nopeus. Kaapelilla voidaan syöttää käyttöjännite laitteelle, ja sen pituus voi olla ilman toistimia yli sata metriä. Taulukossa 1 on vertailtu yleisimpien kameraliitännöiden ominaisuuksia. [14.]

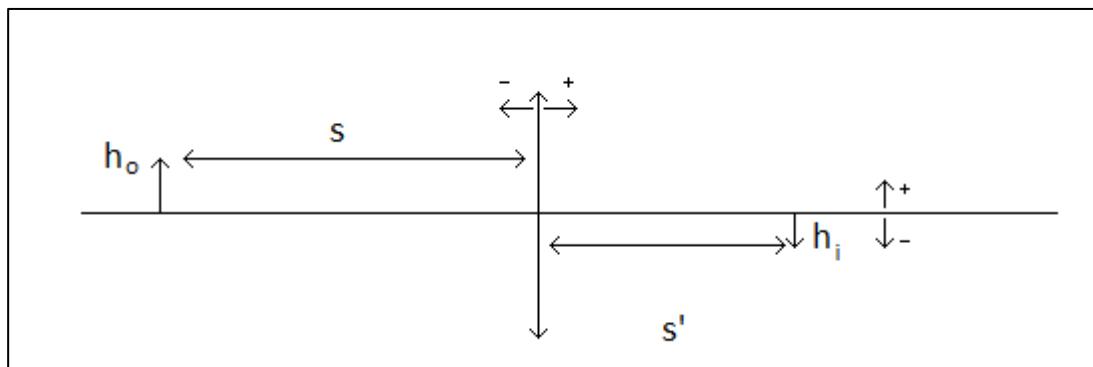
Taulukko 1. Yleisimpien kameraliitännöiden vertailua

Tekniikka	Nopeus	Virransyöttö datakaapelia pitkin	Suurin kaapelin pituus
Gigabit Ethernet	1 Gbit/s	Eriyisjärjestelyllä (PoE)	100 m
USB vision (USB 3.0)	5 Gbit/s	Kyllä	5 m
Camera Link	5,44 Gbit/s	Kyllä	10 m
FireWire 400	400 Mbit/s	Kyllä	4,5 m
FireWire 800	800 Mbit/s	Kyllä	4,5 m
CoaXPress 1x (4x)	6 Gbit/s (20 Gbit/s)	Kyllä	100+ m

Suurin osa konenäkökameroista tukee GenICam-standardia, jonka tehtävä on tarjota tiedonsiirtoteknologiasta riippumaton rajapinta kamerasäädintä, hallintaa ja datan siirtoa varten. Standardi käsittää myös kameroiden yleisten ominaisuuksien nimeämisen yhdenmukaisesti. Standardin ansiosta järjestelmän kamera ja liitäntä voidaan vaihtaa ilman, että järjestelmän ohjelmisto pitäisi kirjoittaa uudelleen. [15.]

2.4 Optiikka

Kameran optiikkaa valittaessa on kiinnitettävä huomiota vaatimuksiin, joita kohteen kuvaaminen järjestelmälle asettaa. Tärkeimmät optiikkaan liittyvät parametrit ovat polttoväli ja f-luku. Esineen etäisyys, kuvan etäisyys ja objektiivin polttoväli riippuvat toisistaan Gaussin kuvausyhtälön mukaisesti [3.]. Jos käytetään kuvan 5 mukaisia merkintöjä,



Kuva 5. Gaussin kuvausyhtälön periaatekuva [3.]

voidaan edellä mainittu yhtälö esittää muodossa

$$\frac{1}{s'} - \frac{1}{s} = \frac{1}{f}$$

Kuvassa olevat merkinnät h_o ja h_i tarkoittavat esineen ja kuvan korkeutta. Kuvan poikittaiselle suurennukselle m saadaan edellä esitetystä kuvasta. [3.]

$$m = \frac{h_i}{h_o} = \frac{s'}{s}$$

Todellinen kuva on ylösalaisin, mistä aiheutuen m on tällöin negatiivinen. [3.]

Kuvattavasta kohteesta ilmaisimelle pääsevää valon määrää kontrolloi objektiivissa oleva säädettävä himmennin, joka yleensä sijaitsee objektiivin muodostavan linssiryhmän keskellä. Himentimen kuvaa nähtynä esineen suunnasta kutsutaan tulopupilliksi. f -luku on objektiivin polttovälin ja tulopupillin halkaisijan suhde. [3.]

$$f - luku = \frac{\text{efektiivinen polttoväli}}{\text{tulopupillin halkaisija}}$$

kun himmennin on täysin auki.

Aukkoluku määritellään samalla tavalla kuin f -luku, mutta himmennin ei ole täysin auki. Aukkoluku ilmaistaan usein $\sqrt{2}$:n kerrannaisina, kuten esimerkiksi 1.4, 2, 2.8, 4, ..., sillä jokaista askelta suuremmalla aukkoluvulla ilmaisimelle pääsevän valon määrä puolittuu. Aukon koon muuttamisella on myös se vaikutus, että syvyystarkkuusalue muuttuu. Syvyystarkkuusalueella tarkoitetaan sitä aluetta näkymän syvyys suunnassa, jossa objektit näkyvät tarkasti. Kun aukon kokoa pienennetään, syvyystarkkuusalue suurenee. [3.] [16.]

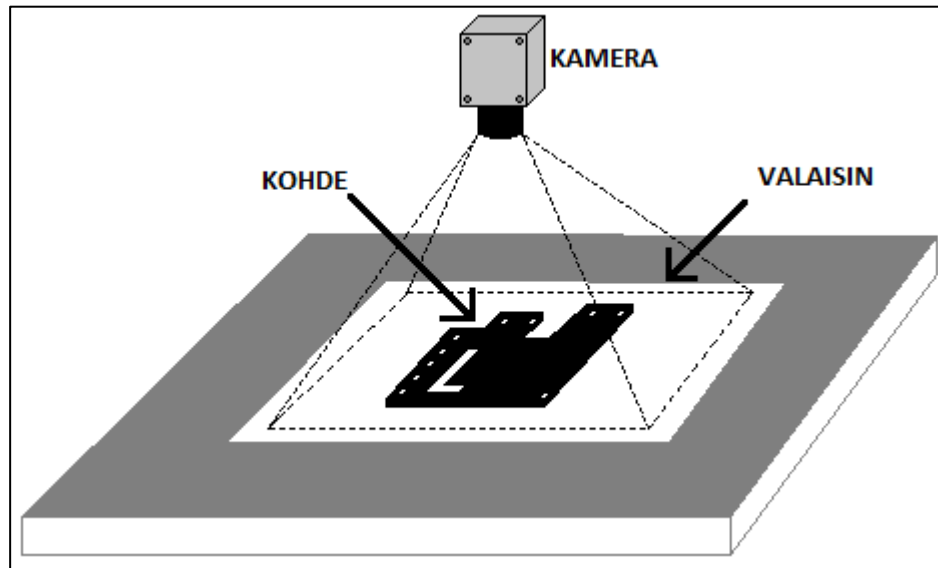
Objektiivin ja kameran väliin voidaan lisätä loittorenkaita, jotka vievät objektiivin kauemmas ilmaisimesta. Niiden avulla lyhyin mahdollinen tarkennusetäisyys pienenee ja suurennussuhde kasvaa. Loittorenkaita käytettäessä valovoima huononee. Esimerkiksi kun loittorenkaiden pituus ja objektiivin polttoväli ovat samat, huononee valovoima puoleen. [17.]

2.5 Valaistus

Valaistus on konenäön kannalta oleellinen osa, sillä kohteesta tutkittavat ominaisuudet saadaan tyypillisesti esille vain tietynlaisessa valaistuksessa. Lisäksi kohteen heijastavuus, läpinäkyvyys ja muoto voivat aiheuttaa rajoituksia käytettävän valaistuksen tyyppiin. [3.]

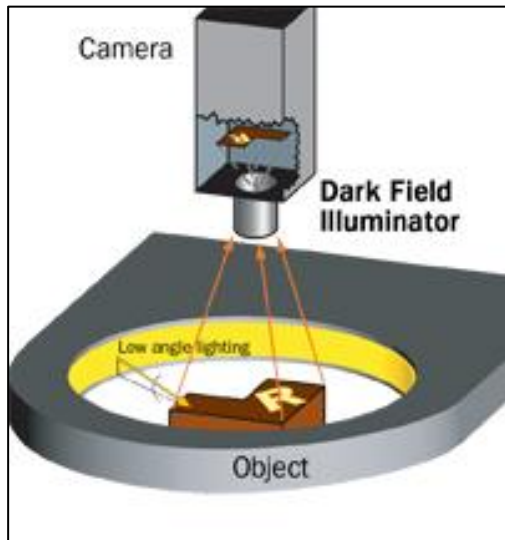
Valaisin voi olla sijoitettu joko kuvattavan kohteen taka- tai etupuolella. Ensimmäisessä tapauksessa kuvassa olevien kohteiden ja sen taustan välinen kontrasti muodostuu kuvattavan kohteen valonläpäisyominaisuuksien perusteella. Käytettävä valo on usein saatettu suunnaltaan satunnaisesti eli diffuusiksi. Taustavalaisulla voidaan tarkastella

esimerkiksi hehkulampun lankaa tai levyyn porattuja reikiä. Kuva 6 havainnollistaa taustavalaistusta. [3.]

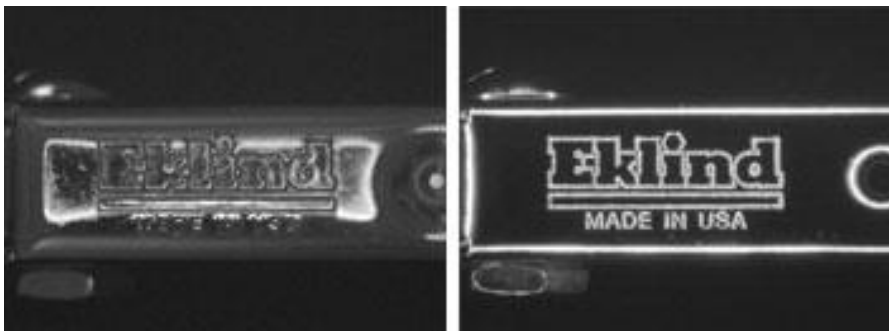


Kuva 6. Taustavalaistus

Tapauksessa, jossa valaisin ja kamera ovat samalla puolella kuvattavaan kohteeseen nähden, voidaan käyttää useita lähestymistapoja. Pimeän kentän valaisussa valon tulokulma on kohteeseen nähden hyvin pieni ja kuvan kontrasti muodostuvat kohteen pinnanmuotojen vaihteluista. Esimerkiksi kohokuvioitu teksti on helppo lukea tällaisessa valaistuksessa. Toinen käyttökohte on esimerkiksi tasaisen pinnan kuvaaminen, jolloin valmistusvirheet tulevat näkyviin. Usein kohteen tausta jää pimeäksi, joten siitä syystä käytetään nimitystä pimeän kentän valaistus. Kuvassa 7 nähdään pimeän kentän valaisun periaatekuva ja kuvassa 8 nähdään tavallisen taustavalaistuksen ja pimeän kentän valaistuksen erot, kun luetaan kohokuvioitua tekstiä heijastavalta pinnalta. [3.]

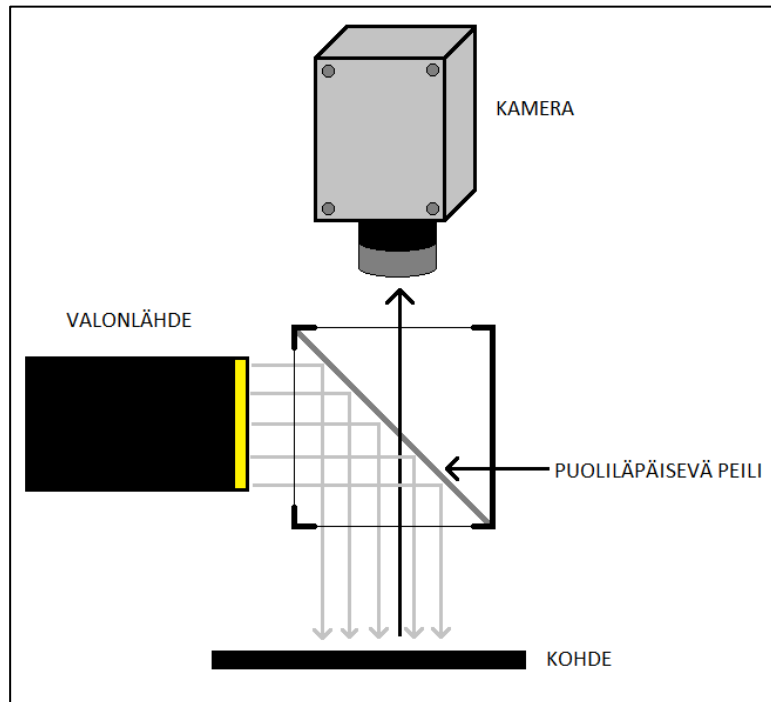


Kuva 7. Pimeän kentän valaisu [18.]



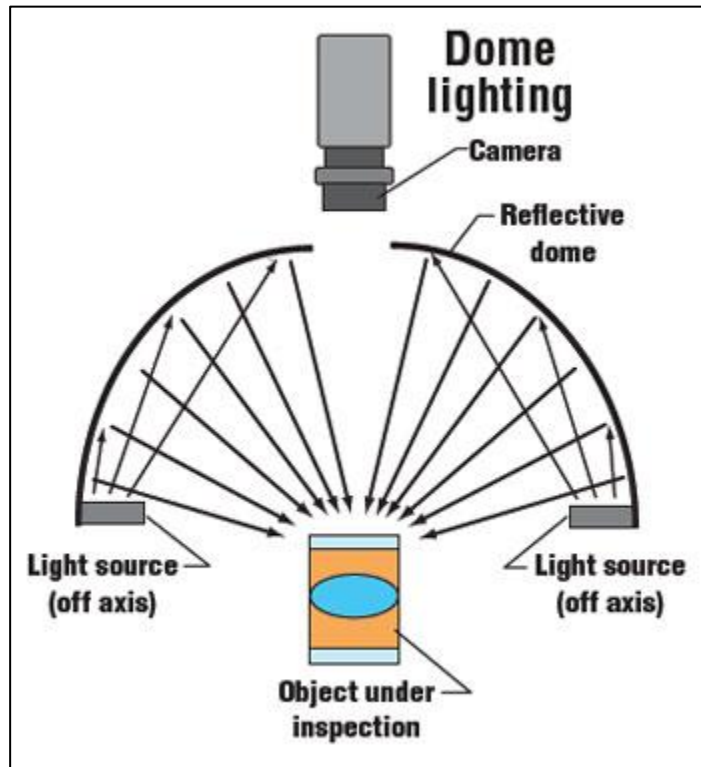
Kuva 8. Kohokuvioitu teksti metallipinnalla tavallisessa taustavalaistuksessa ja pimeän kentän valaistuksessa [18.]

DOAL-valaisimessa (Diffuse On-Axis Light) kamera kuvaa 45 asteen kulmassa olevan puoliläpäisevän peilin läpi. Lamppu valaisee peilin toista pintaa, jolloin osa valon säteistä heijastuu peilistä. Nämä heijastuneet valonsäteet ovat samansuuntaisia kameran kuvaussuunnan kanssa. Kuva 9 havainnollistaa DOAL-valaisimen toimintaa. Tekniikalla saadaan aikaan valaistus, jolla voidaan parhaiten kuvata tasaisia heijastavia pintoja. [3.] [19.]



Kuva 9. DOAL-valaisin

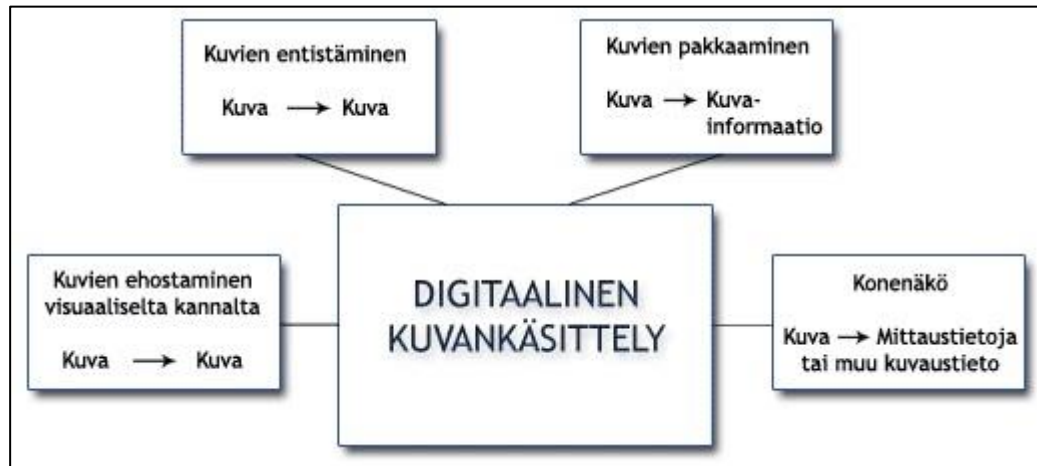
Minkä tahansa kameran kanssa samalla puolella kuvattavaan kohteeseen olevan valaisimen valosta voidaan tehdä diffuusia asettamalla sen eteen diffuusisti valoa läpäisevä levy. Kuitenkaan valo ei ole usein riittävässä määrin diffuusia eliminoimaan heijastuksia kuvattavasta kohteesta. Parhaaseen tulokseen päästään erikoisella diffuusilla valaisimella. Valaisimen muodostaa kupu, jonka sisäpintaa valonlähde valaisee. Kuvusta heijastuva valon suunta on satunnaista eli diffuusia. Kuva 10 havainnollistaa diffuusin valaisimen rakennetta.



Kuva 10. Diffuusi valaisin [20.]

2.6 Digitaalinen kuvankäsittely

Digitaalisessa kuvankäsittelyssä on kyse digitaalisten kuvien muokkaamisesta tietokonealgoritmeilla. Kuvan ehostamista käytetään esimerkiksi digitaalikameralla otettujen kuvien muokkaamiseen tai esimerkiksi lehteen tulevan kuvan käsittelyyn. Kuvan entistämisen on myös digitaalista kuvankäsittelyä ja sen tarkoitus on ehostaa jollakin tavalla huonontuneet kuvan piirteet alkuperäisen kuvan mukaisiksi. Kuvia voidaan pakata, jolloin ne vievät vähemmän tilaa. Tämä on tärkeää silloin, kun käytettävissä oleva tallennus- tai tiedonsiirtokapasiteetti on rajallinen. Kuvassa 11 on esitetty digitaalisen kuvankäsittelyn päämääriä. [3.] [21.]



Kuva 11. Digitaalisen kuvankäsittelyn päämääriä [21.]

2.6.1 Kuvankäsittelymenetelmiä

Suodattaminen

Kuvan suodattamisella pyritään parantamaan kuvan visuaalista ilmettä jatkokäsittelyä varten. Suodattimet jaetaan alipäästösuodatimiin ja ylipäästösuodatimiin. Suodattamisella pyritään parantamaan kuvatietoa myöhempiä vaiheita varten. Ylipäästösuoatimien terävöittää kuvaa; se korostaa alueita, jossa on suuria intensiteetin muutoksia. Alipäästösuoatimien sumentaa kuvaa, sillä se tasoittaa alueita, joilla intensiteetti muuttuu nopeasti. Näin voidaan kuitenkin vähentää kuvassa esiintyvää kohinaa. [21.]

Kynnystäminen

Kuva-analyysissä kuva halutaan jakaa kohteisiin ja taustaan. Tätä kutsutaan segmentoinniksi. Kynnystäminen on segmentointimenetelmä, jolla harmaasävykuvassa olevia harmaan sävyjä pyritään vähentämään. Monesti kynnystyksessä pyritään vähentämään harmaasävyjen määrää kahteen arvoon, joka tarkoittaa täysin mustavalkoista kuvaa. Kynnystettäessä valitaan, minkä kirkkauden alapuolelle jäävät arvot tulkitaan mustaksi. Sen yläpuolelle jäävät arvot tulkitaan valkoisiksi. Kynnystämällä saavutetaan se, että kuvasta häviää analyysille yhdentekevä tieto. [3.] [21.]

Morfologia

Morfologisten operaatioiden avulla paksunnetaan tai ohennetaan kuvassa olevia reunoja. Menetelmällä voidaan esimerkiksi häivyttää kynnyksitetystä kuvasta virheitä tai tehdä rajoista selkeämpiä. Tavallisimmat morfologiset operaatiot ovat eroosio, joka ohentaa kuvassa olevia objekteja, ja dilaatio, joka paksuntaa kuvassa olevia objekteja. [3.] [21.]

2.6.2 Kuvankäsittelyohjelmistot

Kuvankäsittelyyn tarvitaan koneellisesti suoritettavaa ohjelmaa. Niiden toteuttamisessa hyödynnetään kaupallisesti saatavia tai avoimen lähdekoodin ratkaisuja. Ohjelmiston tehtävä on analysoida kuvan informaatio ja ohjata tuotantoa analyysin pohjalta. Esimerkiksi vialliseksi havaittu tuote poistetaan liukuhihnalta koneellisesti. Kaupallisissa tuotteissa on usein paljon laitekohtaisia ajureita, jotka nopeuttavat järjestelmän kehitystä. Suurin osa konenäköohjelmistoista on kirjoitettu C/C++ -kielellä, sillä analysoidavan datan määrä on suuri ja analyysin on tapahduttava hyvin nopeasti.

Kuvankäsittelyyn ja konenäköön erikoistuneet ohjelmointiympäristöt sisältävät tyypillisesti tuen kuvanmuodostukselle, suodattimille, morfologisille operaatioille, kuvien muokkaukselle kuten siirtäminen, kiertäminen, skaalaus, interpolointi ja liittäminen; sekä kohteiden kuvaukseen tarvittavat työkalut, mittaustyökalut ja viivakoodin luvun. Seuraavassa on otettu tarkasteluun yksi avoimen lähdekoodin ja kaksi kaupallista ratkaisua.

OpenCV

OpenCV on ilmainen BSD-lisenssillä toteutettu avoimen lähdekoodin konenäkökirjasto. Se on kirjoitettu C:llä ja C++:lla, ja siitä löytyy ohjelmointirajapinnat C:lle, C++:lle, Pythonille sekä Javalle. OpenCV tukee Windowsia, Linuxia, Mac OS:aa, iOS:aa sekä Androidia. [22.] [23.]

LabVIEW + Vision Development Module

LabVIEW on National Instrumentsin kehittämä kaupallinen ohjelmointiympäristö, joka on toteutettu C-kielellä. Se tukee Windowsia, Linuxia, Macia ja Solarista. Ohjelmointiympäristöön kuuluu kattava määrä erilaisia ajureita, joista yksi osa-alue on konenäkö. [24.]

MVTec HALCON

MVTec HALCON on kattava integroitu kehitysympäristö, jossa on suuri määrä erityyppisiä valmiita konenäköfunktioita. HALCON sisältää rajapinnat C-, C++-, .NET- ja COM -kielille. Kehitystyökalulla rakennettu konenäkösovellus voidaan viedä C-, C++-, C#- tai Visual Basic -ohjelmointikielelle. HALCONilla voidaan yleisimpien konenäkösovelluksien lisäksi kehittää stereonäkösovelluksia ja 3D-mittaussovelluksia. Se tukee myös monisäikeistä laskentaa sekä GPU-laskentaa. [25.]

3 MITTAUSJÄRJESTELMÄN KUVAUS

Tässä osiossa on esitelty mittausjärjestelmän rakenne. Ensin on käyty läpi kamera ja sen ominaisuudet. Seuraavaksi on tarkasteltu kameran kanssa käytettyä optiikkaa ja valaisinta ja lopuksi PC-ympäristö, jossa kehitystyö tapahtui.

3.1 Kamera

Järjestelmän kameraksi valittiin Basler acA2000-50gm, jossa on 2048 x 1088 pikselin CMOS-kenno. Kameran ilmaisimen koko on 2/3” ja pikselin koko ilmaisimella on 5.5 μm * 5.5 μm . Kamera muodostaa harmaasävykuvaa, jonka värisyvyys on 12 bittiä. Se tukee useita eri videokuvaformaatteja, kuten Mono 8, Mono 12 sekä YUV 4:2:2 Packed. [26.]

Järjestelmään valittiin käytettäväksi Mono 8 -formaatti, sillä se on järjestelmän toiminnan kannalta riittävä ja suuremmat harmaasävyvyvydet lisäävät vain analysoitavan tiedon määrää. Koko kennoa käytettäessä kuvausnopeus on 50 kuvaa sekunnissa, mutta rajaamalla kuvan kokoa voidaan ottaa jopa 2000 kuvaa sekunnissa.

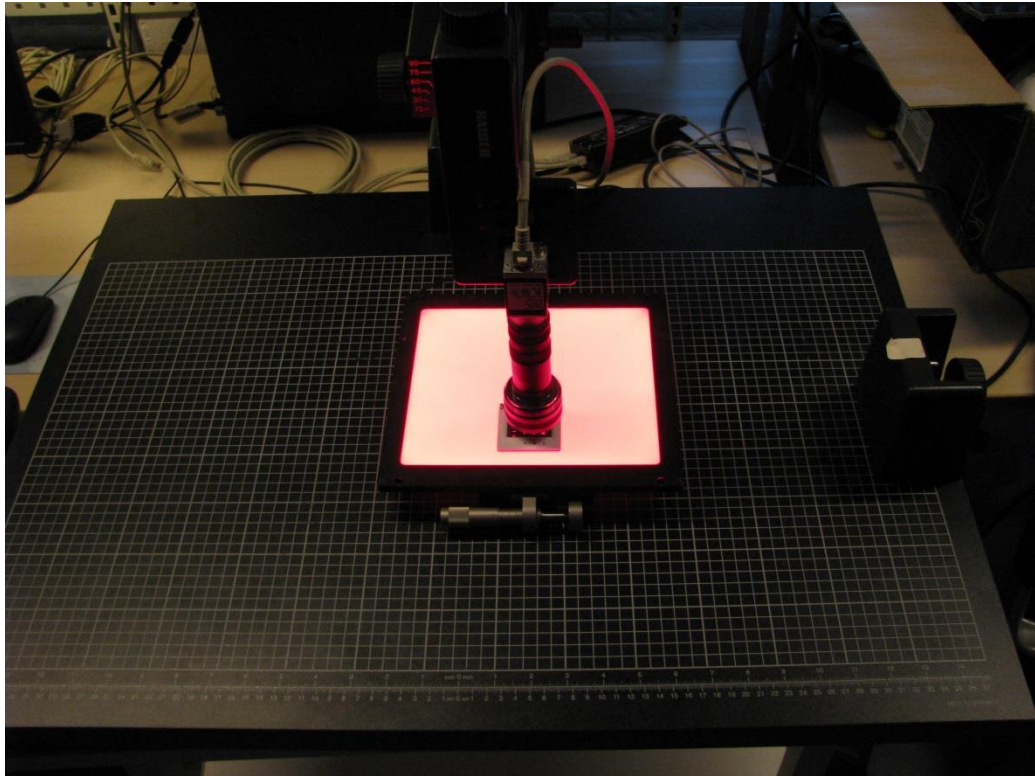
Kameran tiedonsiirto tapahtuu gigabit Ethernetin välityksellä, ja se kykenee lähettämään jumbokokoisia Ethernet-paketteja. Ne ovat kooltaan yli 1500 tavua, ja niiden maksimikoko on 9000 tavua. Jumbokokoisten pakettien tarkoitus on vähentää prosessorin rasitusta, sillä kuuden 1500 tavun kokoisen paketin käsittely vaatii enemmän laskutoimituksia kuin yhden 9000 tavun kokoisen paketin. [27.]

Kamera saa käyttöjännitteen Power over Ethernet –tekniikalla. Se tarkoittaa sitä, että laite saa käyttöjännitteensä Ethernet-kaapelia pitkin, jolloin erillistä virtajohtoa ei tarvita. [26.]

Kuvaus voidaan käynnistää ohjelmallisesti Ethernet-yhteyden kautta tai erillisen liipaisukaapelin kautta. Tätä käytettäessä kennon valotusaika voidaan määrittää kahdella eri tavalla: toisessa kennon valotus on yhtä pitkä kuin liipaisusignaali ja toisessa valotusajan voi määrittää kameran asetuksien avulla liipaisun ilmoittaessa vain valotuksen alkamisajankohdan. [26.]

Kameran kennon ja Ethernet-kontrollerin välissä on kuvapuskuri, joka sallii datan lukemisen kennolta ilman, että tarvitsee odottaa datan siirtoa pois kameran. Tämän ansiosta tiedonsiirtonopeudella ei ole vaikutusta kuvausnopeuteen. Kuvapuskuri on kuitenkin rajallinen, sillä muistin täytyessä se alkaa korvata vanhimman puskurissa olevan kuvan uudella.

Kamera asennettiin repropöytään, johon saa kiinnitettyä erilaisia instrumentteja siihen tarkoitettujen palkkien avulla. Kuvassa 12 on esitetty repropöytä ja sen säädettävään kameravarteen kiinnitetty kamera.



Kuva 12. Repropöytä, kamera ja taustavalo

3.2 Optiikka

Kameraan valitun objektiivin f-luku oli 1.4 ja polttoväli 25 mm. Lisäksi käytettiin sarjaa loittorenkaita, joilla saavutettiin riittävään erotustarkkuuteen tarvittu suurennus.

3.3 Valaisu

Järjestelmässä valaisuun käytettiin diffuusia taustavalaisua, sillä kuvattavaksi kohteeksi valittiin ohut metallilevy, jossa oli kaksi kapeaa, pitkänomaista reikää. Taustavalo pääsee reikien lävitse, jolloin saavutetaan kuvassa erinomainen kontrasti seurattavan kohteen ja taustan välille. Todellisuudessa käytettäisiin korkealaatuista tulostetta, joka kiinnitettäisiin kohteeseen ja valaistus olisi toteutettu riittävän intensiteetin ja diffuusioasteen tuottavalla valaistuksella.

3.4 PC-ympäristö

Ohjelma toteutettiin Windows 7 -ympäristössä LabVIEW:llä. Lisäksi käytettiin LabVIEW:n Machine Vision -moduulia, joka sisältää konenäköön liittyviä funktioita. Järjestelmän käyttäminen Linux-ympäristöstä ei pitäisi olla mikään ongelma, sillä LabVIEW on alustariippumaton. LabVIEW:llä toteutettujen ohjelmien ajamiseen tarvitaan kuitenkin vähintään LabVIEW runtime engine.

4 OHJELMOINTIYMPÄRISTÖN KUVAUS

4.1 LabVIEW

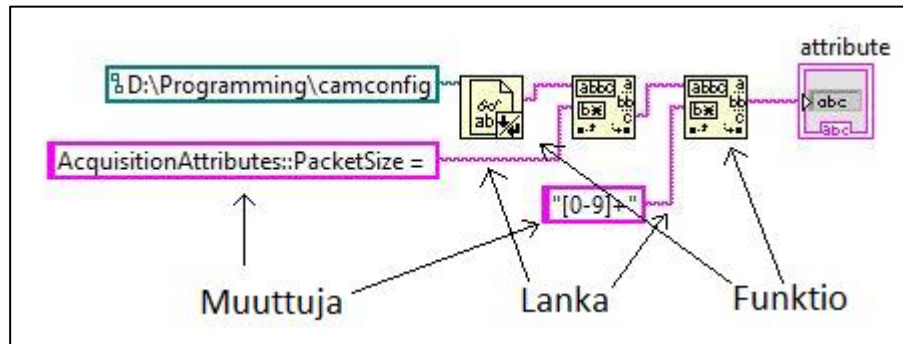
Järjestelmä toteutettiin LabVIEW-ohjelmointiympäristössä. LabVIEW tulee sanoista Laboratory Virtual Instrument Engineering Workbench ja on National Instrumentsin kehittämä kaupallinen ohjelmointiympäristö. National Instruments on yritys, joka keskittyy testaus- ja mittauslaitteiden valmistamiseen. [28.]

LabVIEW:n ohjelmointikieli on G-kieltä, joka eroaa perinteisistä ohjelmointikielistä, kuten esimerkiksi C++:sta tai Pythonista siten, että ohjelmakoodi on graafista [26]. Yksinkertaisuuden vuoksi jatkossa ohjelmointikieleen viitattaessa käytetään ohjelmointiympäristön, LabVIEW:n, nimeä.

Perinteiset ohjelmointikieliset ovat tekstipohjaisia, joissa syntaksi on sanoja, kirjaimia, numeroita ja erikoismerkkejä. Niissä tiedon liikkuminen toteutuu antamalla funktiolle muuttujia. LabVIEW sen sijaan pohjautuu tietovirtoihin, joissa tieto liikkuu muuttujien ja funktioiden välillä lankoja pitkin [29.]. Kuvassa 13 on esimerkki Python-kielisestä tekstipohjaisesta ohjelmointikielestä, joka avaa tekstitiedoston, etsii sieltä rivin, jossa esiintyy teksti "PacketSize" ja tulostaa rivillä olevan attribuutin arvon. Kuvassa 14 on LabVIEW-kielinen ohjelma, joka suorittaa saman toiminnon.

```
f = open('camconfig', 'r').read()
lines = f.splitlines()
for line in lines:
    if "PacketSize" in line:
        break
attribute = line.split()
print attribute[2]
```

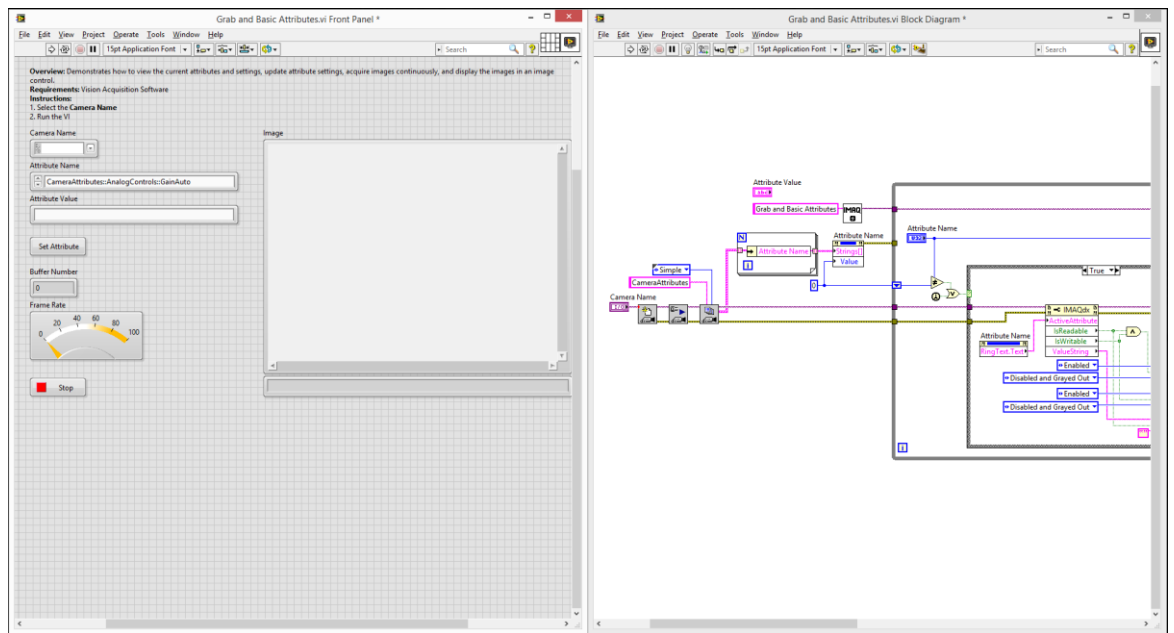
Kuva 13. Esimerkki tekstipohjaisesta ohjelmointikielestä



Kuva 14. Esimerkki LabVIEW-ohjelmasta

Tekstipohjaisen koodin esimerkissä attribuutti etsitään tarkistamalla jokaiselta riviltä, sijaitseeko sillä avainsana. Osuman jälkeen rivi pilkotaan osiin, ja riviltä tulostetaan attribuutin arvoa vastaava osa. LabVIEW-koodissa attribuutin nimi annetaan funktiolle, joka etsii attribuutin paikan tekstitiedostosta, jonka jälkeen etsitään mitä tahansa numeroarvoa, joka attribuutin nimen jälkeen tulee.

LabVIEW-kehitysympäristön käyttöliittymä koostuu kahdesta osasta: etupaneeli ja vuokaavioikkuna. Etupaneeli koostuu ohjelmoijan valitsemista käyttöliittymäkomponenteista, joiden avulla käyttäjä voi hallita ohjelman toimintaa. Tällaisia komponentteja ovat muun muassa napit, merkkivalot, teksti-ikkunat, kaaviot, numerokontrollit ja säätimet. Vuokaavioikkunassa laaditaan ohjelmakoodi, joka rakentuu funktioista, muuttujista ja langoista. LabVIEW:ssä funktiota kutsutaan Virtual Instrumentiksi eli VI:ksi. Se on yleensä neliömäinen ikoni, jonka kuva toimii sen tunnisteena. VI:ssä on myös liittimet, joita pitkin se saa tarvitsemansa muuttujat ja joihin se tulostaa toimintansa tulokset. Molemmissa ikkunoissa on mahdollista aukaista valikko, joista löytyvät kaikki komponentit, joita LabVIEW-ohjelman laatimisessa tarvitaan [28.]. Kuvassa 15 on esitetty LabVIEW-ohjelmointiympäristön tavanomainen asettelu, jossa etupaneeli on vasemmalla ja vuokaavioikkuna on oikealla puolella näyttöä.



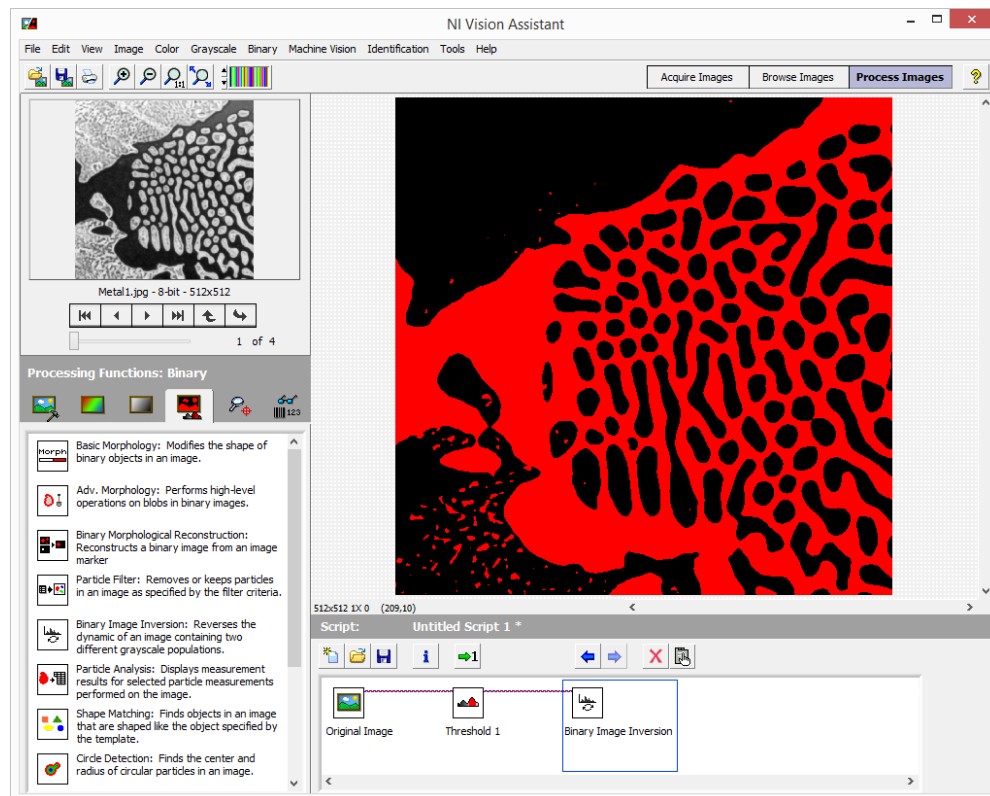
Kuva 15. LabVIEW-kehitysympäristö

LabVIEW perustuu tietovirtoihin. Kun ohjelman funktiolla on kaikki sen tarvitsemat tiedot, se on valmis suoritettavaksi. Tästä syystä ohjelman eri osien suoritusjärjestys ei ole välttämättä joka kerta sama. Kun useampi funktio ohjelmassa on yhtä aikaa siinä tilassa, että se voidaan ajaa, LabVIEW päättää mielivaltaisesti, missä järjestyksessä ne ajetaan. [28.] [29.]

Vaikka moniajolla saavutetaan nopeasti toimivia ohjelmia, on joissain tapauksissa ohjelman eri osien ajo rinnakkain haitaksi ohjelman toiminnalle. LabVIEW:ssä ohjelmakoodin suoritusjärjestyksen voi pakottaa useammalla eri tavalla. Näistä tavoista helpoin ja suositelluin on virheklustereiden käyttö, joka varmistaa ohjelmakoodin peräkkäisen suorituksen. Toinen tapa on käyttää sekvenssirakennetta, jota voidaan verrata GOTO-rakenteeseen tekstipohjaisissa kielissä. Samoin kuin GOTO-rakennetta, sekvenssirakennetta ei suositella käytettäväksi, ellei muuta keinoa ohjelmakoodin suoritusjärjestyksen pakottamiseksi ole. [29.]

4.2 Vision Assistant

Vision Assistant on National Instrumentsin ohjelmisto, joka on tarkoitettu kuvankäsittelysovellusten testaamiseen. Se on osa LabVIEW:n Vision Development -moduulia, ja sillä tuotetut skriptit voidaan kääntää LabVIEW-, C- ja .NET-ohjelmointikielille [30.]. Kuvassa 16 on esitetty Vision Assistantilla rakennettu skripti.



Kuva 16. Vision Assistant

Skript-ikkuna näyttää skriptissä olevat vaiheet ja näiden vaiheiden asetukset. Skriptejä voidaan käyttää yhden kuvan tai kuvajoukon analysoimiseen. Niitä voidaan muokata ja tallentaa. Image browser -ikkuna sisältää kaikki kuvat, jotka on ladattu Vision Assistantiin. Ikkunan kautta voidaan valita kuva prosessointia varten. Processing-ikkuna päivittää kuvaa sitä mukaa, kun skripti sitä muokkaa. Ikkuna näyttää muutokset heti, joten parametrejä voidaan muokata, kunnes tulos on halutunlainen. Functions-ikkuna listaa funktioita, joita voidaan käyttää skriptien rakentamiseen. Lisäksi tässä ikkunassa näkyvät valitun funktion parametrit.

5 OHJELMISTON TOTEUTUS

Työn tavoite oli ottaa käyttöön hankittu kamera ja toteuttaa ohjelmisto, jonka avulla voidaan mitata kappaleen liiketietoa. Ohjelmistosuunnittelua tehtäessä järjestelmä jaettiin osakokonaisuuksiksi. Niiksi muodostuivat suunnittelun edetessä kameran konfigurointi, mittauksen teko, sen analysointi sekä tulosten esittäminen. Ohjelmisto sovittiin toteutettavaksi siten, että jokainen osakokonaisuus olisi erikseen kutsuttavissa, sillä se oli tarkoitus integroida osaksi suurempaa mittausjärjestelmää.

5.1 Työlle asetetut vaatimukset

Kameran konfiguroinnin tarkoitus on antaa käyttäjälle mahdollisuus päästä muuttamaan kameran asetuksia. Koska kamerassa oli lähes kaksisataa erilaista asetusta, oli selvää, ettei kaikkia voitu ottaa mukaan. Ohjelmaan valittiin joukko asetuksia, jotka pääasiassa vaikuttavat kuvan laatuun.

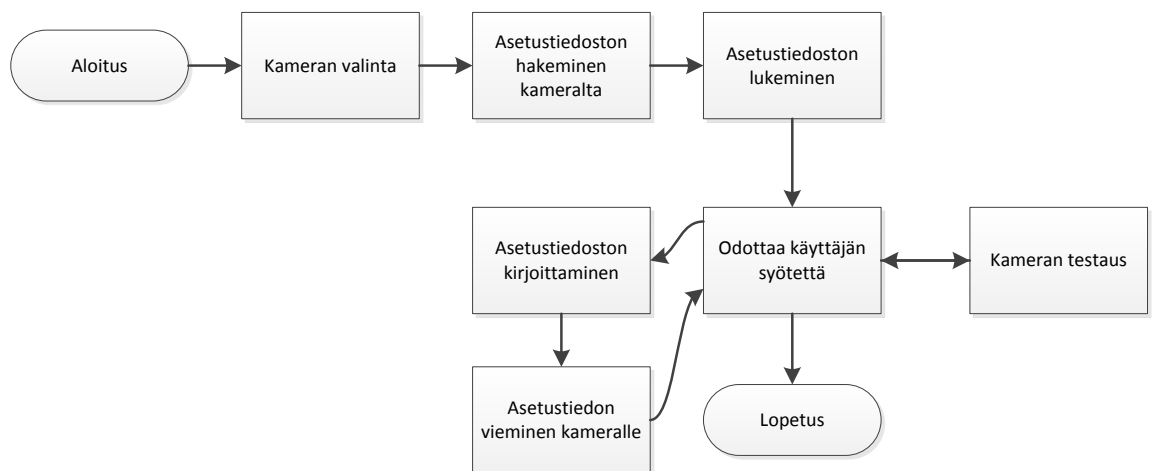
Mittauksen teko suunniteltiin tehtäväksi siten, että mittaus voidaan aloittaa käyttäjän syötteen avulla tai ulkoisen liipaisun avulla. Kehitystyössä ilmenneet ongelmat aiheuttivat ajanpuutteen, jonka takia ulkoisen liipaisun toteutus jäi tekemättä.

Järjestelmän erääksi osaksi suunniteltiin kalibraatioaliohjelma, jonka tehtävänä oli antaa analyysille tiedot kohteen mittasuhteista. Sen perusteella mittauksen liiketieto olisi saatu SI-järjestelmän yksiköissä ilmaistuksi. Kohteen valinnasta kuitenkin aiheutui se, että erillistä kalibraatiota ei tarvitse tehdä ennen mittausta.

Mittauksen analysointia varten suunniteltiin kehitettäväksi kaksi erillistä algoritmia, joista toisen olisi toteutettu työssä ja toinen olisi pohjautunut LabVIEW:n konenäköfunktioihin.

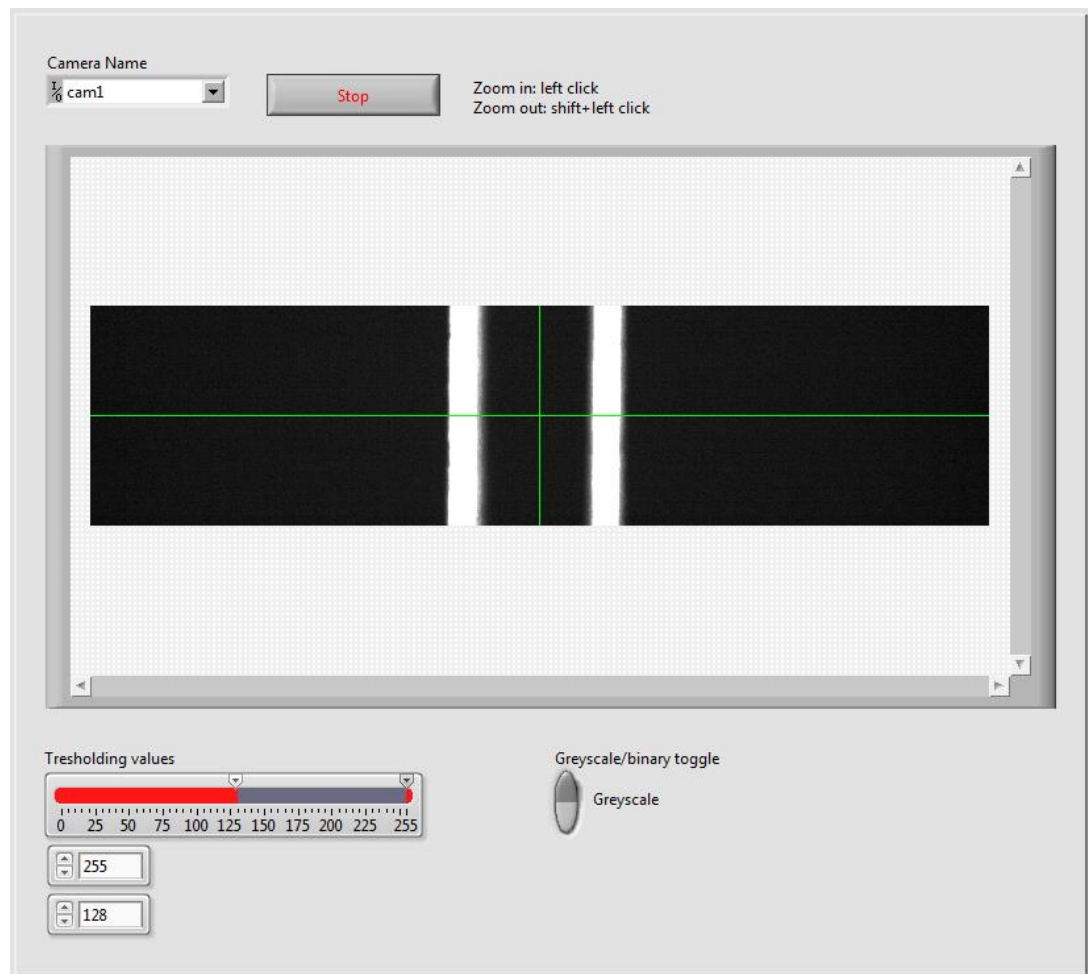
5.2 Kameran konfigurointi

Kameran konfigurointi tapahtuu erillisessä aliohjelmassa, joka hakee asetukset kameralta. Käyttäjä voi muuttaa näitä asetuksia, jonka jälkeen ne voidaan tallentaa kameralle. Aliohjelma listaa käyttäjälle ne asetukset, jotka on katsottu tärkeiksi järjestelmän toiminnan kannalta. Siinä on myös mahdollisuus testata kameraa. Kuvassa 17 on esitetty konfiguroinnin lohkokaavio.



Kuva 17. Konfiguroinnin lohkokaavio

Kameran testauksessa näytetään kuva-alaa, joka on suurempi kuin mittauksessa käytettävä. Näin helpotetaan kameran asennon säätämistä ja kohteen löytymistä. Testikuvassa näkyy pysty- ja vaaka-akselilla keskiviivat, joiden tarkoitus on myös helpottaa kameran kohdistamista. Kameralta tulevaa kuvaa voidaan tarkastella harmaasävykuvan lisäksi binäärisessä muodossa, jolloin on helpompi todeta, onko kuvan kontrasti riittävä kohteen löytymiseksi. Kuvassa 18 on esitetty testausohjelman etupaneeli.



Kuva 18. Kameran testaus -aliohjelman etupaneeli

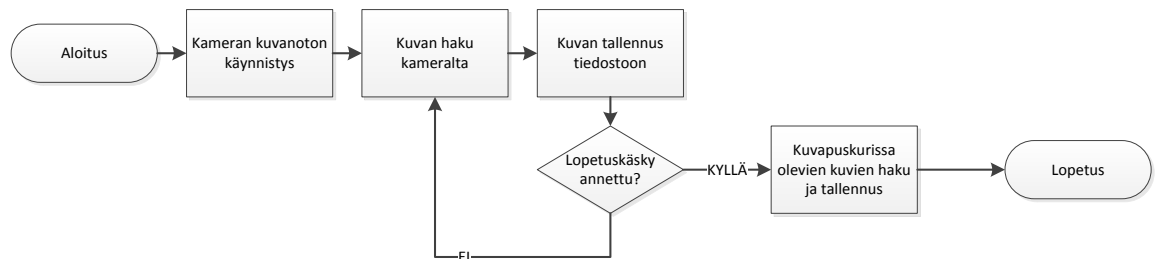
5.3 Kalibrointi

Kalibrointi tapahtuu samalla, kun mittaustuloksia analysoidaan. Kohde voi olla kaksi pystysuoraa viivaa, joiden välinen matka tiedetään. Kun viivojen välinen matka millimetreinä jaetaan niiden välisellä matkalla pikseleissä, saadaan yhden pikselin leveys millimetreinä.

5.4 Mittaus

Heti mittauksen alettua PC alkaa lukemaan kameran kuvia Ethernet-yhteyttä pitkin. Saadut kuvat tallennetaan välittömästi kiintolevyllä pakkaamattomaan .avi-muotoon. Koska

kameran kuvausnopeus saattaa olla hieman nopeampi kuin PC:n kyky ottaa vastaan ja tallentaa kuvia, menevät otetut kuvat kameran kuvapuskuriin. Siellä voi olla useita tuhansia yksittäisiä kuvia odottamassa lukua PC:lle. Kun mittaus on loppunut, PC lukee kameran puskurista jäljellä olevat yksittäiset kuvat. Kuvassa 19 on esitetty ohjelman lohkokaavio.



Kuva 19. Mittauksen teon lohkokaavio

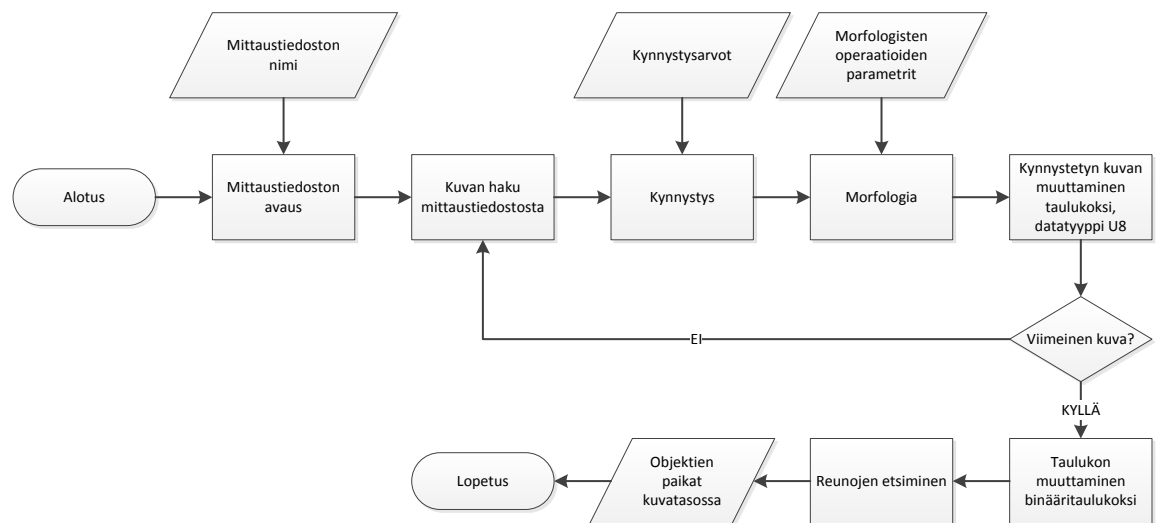
Kiintolevylle tallentaminen havaittiin nopeimmaksi tavaksi ottaa kuvatietoa vastaan. Eri tallennusmenetelmiä vertaillen todettiin, että esimerkiksi kuvatiedon vieminen tietokoneen RAM-muistiin on hitaampaa kuin kiintolevylle tallentaminen. Lisäksi RAM-muistiin tallentamisen heikkoudeksi todettiin se, että pitkän mittauksen aikana muisti saattaa täytyä, jonka seurauksena ohjelma kaatuu. Tästä aiheutuisi kaiken mittausdatan menetys.

5.5 Analysointi

Mittausdatan analysointi tapahtuu kuva kovalta. Mittauksen data on kuvan 18 mukaista, mutta siinä pystyresoluutio on 10 pikseliä. Analyysin ideana oli löytää kuvassa näkyvät vaaleat, pystysuorat kohteet ja laskea niiden välinen keskipiste. Kun kaikki kuvat on analysoitu, esitetään löydetty keskipisteet graafisella näytöllä. Tämä kertoo kohteen liikkeen kameraan nähden.

Kohteiden havaitsemista varten laadittiin kaksi eri algoritmia, joiden avulla saatu data voidaan analysoida. Toisessa käytettiin hyväksi LabVIEW:n omaa funktiota, joka etsii harmaasävykuvasta objekteja. Funktiolle määriteltiin etsittävän objektin minimikoko ja löydettyistä objekteista haluttiin tietää niiden massakeskipiste. Sen x-koordinaattia käytetään mittaustuloksen analysoimiseen.

Toisessa algoritmiossa harmaasävykuva muutettiin binäärikuvaksi, josta kohteet etsittiin työssä toteutetulla funktiolla. Algoritmiossa otettiin kuvan pikselirivit yksitellen käsittelyyn ja etsittiin niistä kohteiden reunat vertaamalla kahden peräkkäisen pikselin arvoja. Jos pikselin arvot eivät olleet samat, voitiin todeta, että reuna on löytynyt. Kun kummankin kohteen reunojen paikat ovat tiedossa, niistä otetaan keskiarvo, joka kuvaa kohteiden keskipistettä x-akselilla. Kuvassa 20 on esitetty binäärianalyysin lohkokaavio.

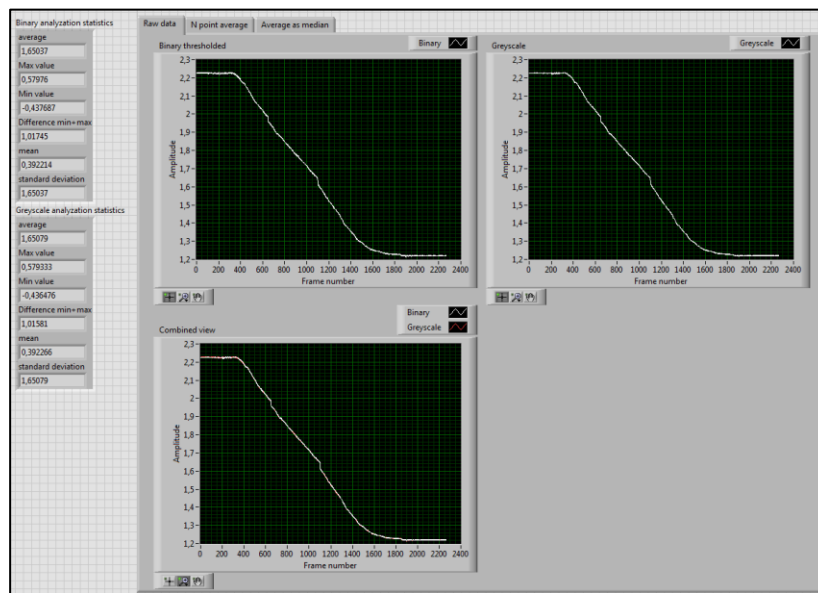


Kuva 20. Binäärianalyysin lohkokaavio

Kummatkin algoritmit tuottavat taulukon, jossa on mittauksen jokaisen kuvan kohteiden keskipisteet. Tämä taulukko esitetään käyttäjälle graafisessa muodossa.

5.6 Tulosten esittäminen

Tulokset esitettiin graafina, josta ilmenee kohteen liike paikkatasossa. Kuvassa 21 on esitetty esimerkki analyysin tuloksista. Analyysit on suoritettu kummallakin algoritmilla. Kuvassa ylhäällä keskellä oleva graafi esittää työssä toteutetun algoritmin tuottamat tulokset ja ylhäällä oikeassa reunassa oleva graafi esittää LabVIEW:n konenäköfunktioihin pohjautuvan algoritmin tuottamat tulokset. Keskellä alhaalla olevassa graafissa kummankin algoritmin tulokset on asetettu päällekkäin. Kuvan vasemmassa reunassa on analyysin datapisteiden pohjalta laskettuja tietoja, kuten suurin ja pienin arvo, näiden arvojen erotus ja keskihajonta. Kuvan arvot ovat millimetreinä.



Kuva 21. Mittaustulosten esittäminen

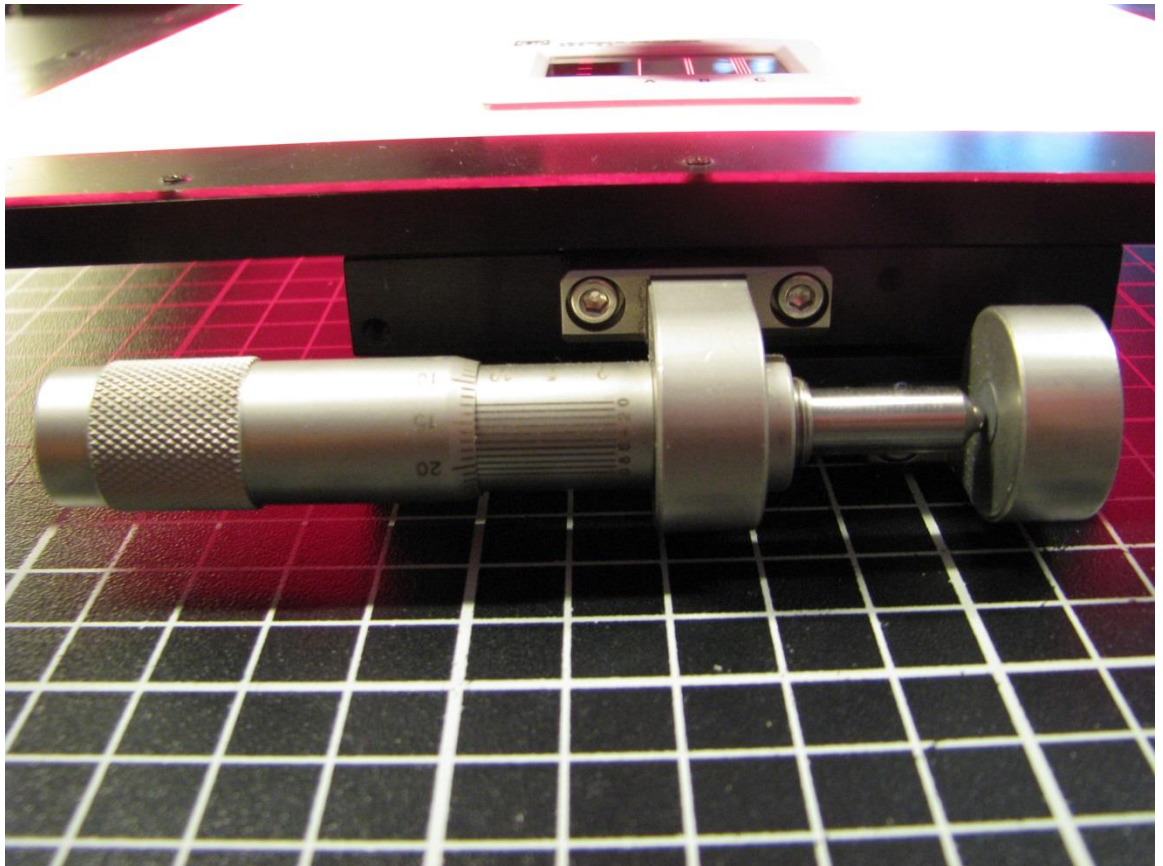
5.7 Ohjelmakoodin testaus

Ohjelmakoodia testattiin kehityksen yhteydessä, sillä aikaisemmin valmistuneita osakokonaisuuksia käytettiin hyväksi seuraavan osakokonaisuuden kehityksessä. Ensiksi toteutettu ohjelmiston osa oli kameran konfigurointi, jonka avulla kameran säätöjä voitiin muuttaa. Tätä työkalua käytettiin myöhemmin runsaasti muun ohjelmiston kehityksen yhteydessä, ja tänä aikana konfiguroinnissa havaitut ohjelmointivirheet korjattiin pikaisesti, jotta kehitystyö seuraavan osakokonaisuuden parissa pystyi jatkumaan.

Mittauksen teko -funktio on toiminnallisuudeltaan hyvin yksinkertainen, sillä se aloittaa mittauksen heti, kun se ajetaan ja lopettaa sen stop-napista. Suurimmat haasteet olivat toimivan algoritmin kehittämisessä, jonka vuoksi testausta suoritettiin mittava määrä kehityksen yhteydessä. Esimerkiksi mittauksen jälkeen kuvapuskuriin jääneiden kuvien haku vaati paljon kokeiluja eri menetelmillä. Kun osio oli valmis, oli sen toiminnallisuus jo kattavasti testattu.

Mittauksen analysoinnin testaus suoritettiin ensin tarkastelemalla mittausta, jossa kohteen liike oli sattumanvaraista. Ensiksi valmistunut algoritmi oli harmaasävyanalyysi. Sen antamia tuloksia verrattiin seuraavaksi valmistuneeseen ja koodiltaan huomattavasti

monimutkaisempaan binäärianalyysiin. Kun kummatkin algoritmit antoivat identtiset tulokset mitattaessa sattumanvaraista liikettä, todettiin niiden olevan valmiita. Seuraavaksi algoritmien testausta varten saatiin lineaarinen siirtoyksikkö, jonka avulla voitiin siirtää kuvattavaa kohdetta 10 μm askelvälein. Kuvassa 22 on esitetty repropöytään kiinnitetty lineaarinen siirtoyksikkö ja sen päällä oleva taustavalo. Kuvan yläreunassa keskellä on nähtävissä metallilevy, jossa seurattava kohde on.



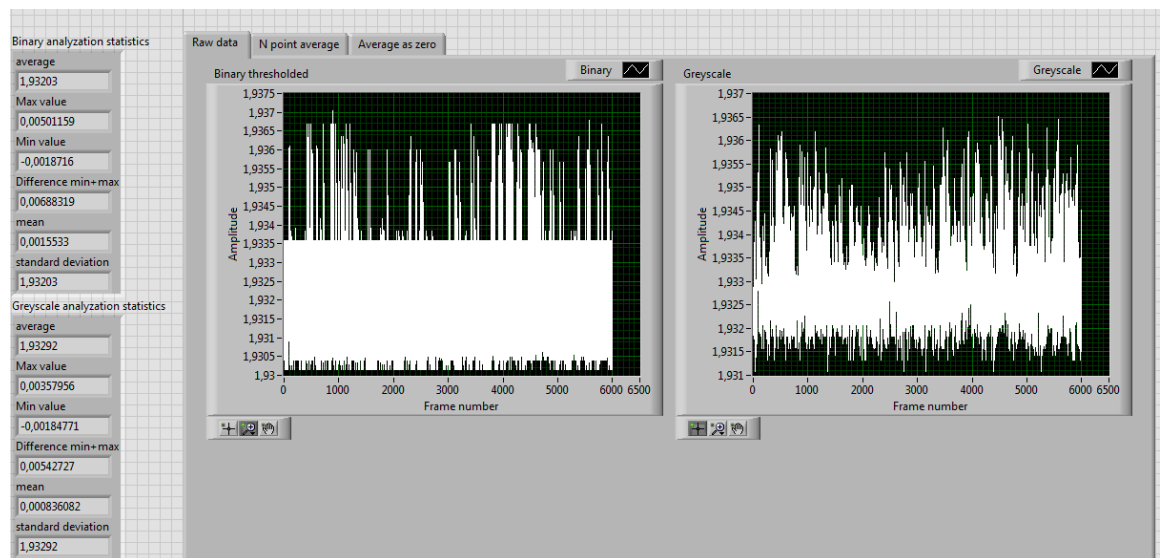
Kuva 22. Repropöytään kiinnitetty lineaarinen siirtoyksikkö

Lineaarista siirtoyksikköä käyttämällä suoritettiin mittaus, jossa kohteen liikkuma matka oli 1 millimetri. Harmaasävyanalyysi ilmoitti liikutuksi matkaksi 1,015 millimetriä ja binäärianalyysi 1,017 millimetriä. Tämän mittauksen avulla varmistui, että kummatkin algoritmit analysoivat mittauksen oikein ja että mittaustiedon muuttaminen SI-järjestelmän yksiköiksi toimii. Seuraavaksi voitiin siirtyä analysoimaan järjestelmän suorituskykyä.

6 JÄRJESTELMÄN SUORITUSKYVYN ANALYSOINTI

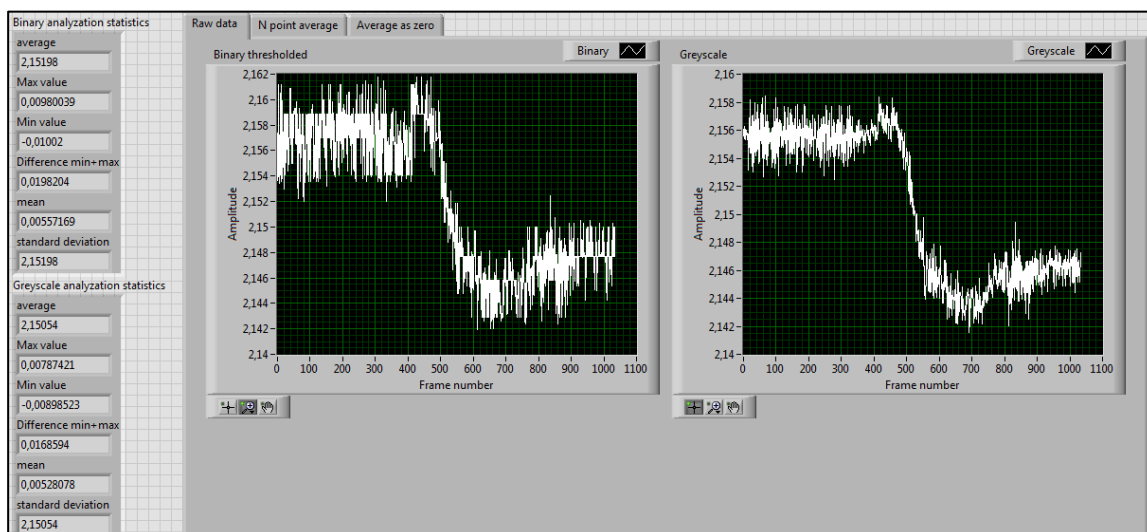
Mittaustuloksia tarkastellessa on otettava huomioon, että repropöytä oli sijoitettu alustalle, joka ei ollut mittauksen tekoa varten ideaalinen. Lisäksi lineaarista siirtoyksikköä liikutettiin käsin pyörittämällä säädintä halutun matkan verran. Nämä seikat vaikeuttavat järjestelmän suorituskyvyn arviointia.

Ensimmäiseksi tutkittiin ympäristön tuottamasta tärinästä ja kamerajärjestelmästä aiheutuvaa kohinaa. Tätä varten suoritettiin 15 sekunnin pituinen mittaus, jonka aikana kohdetta ei liikutettu. Harmaasävyanalyysi ilmoitti kohteen paikan suurimmaksi vaihteluväliksi 0,005 mm keskihajonnan ollessa 0,0008 mm. Binäärianalyysi ilmoitti suurimmaksi vaihteluväliksi 0,007 mm keskihajonnan ollessa 0,0015 mm. Kuvassa 23 on esitetty kohinan mittauksen tulokset.



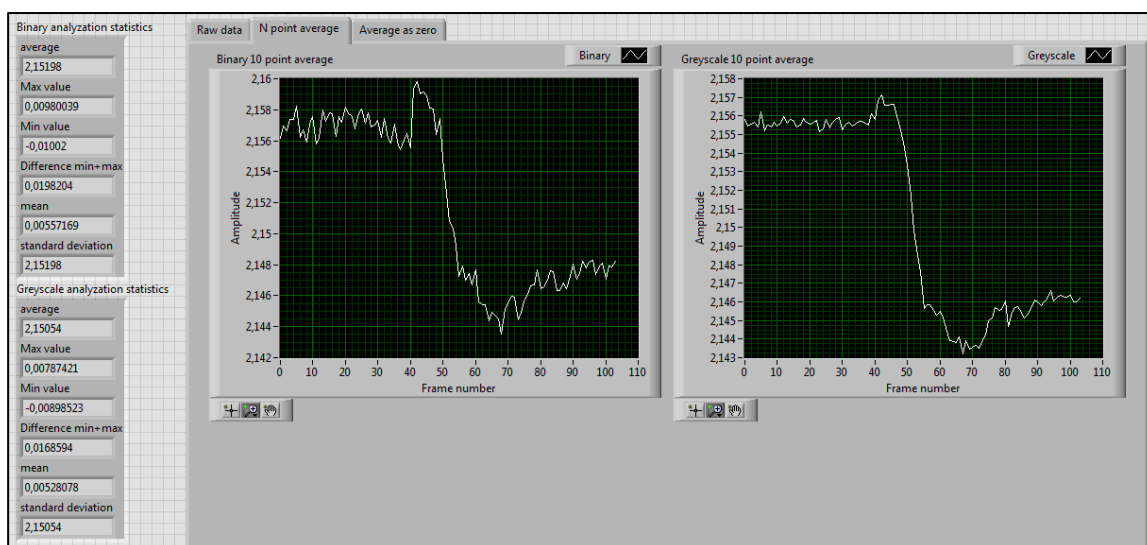
Kuva 23. Kohinan mittaus

Seuraavaksi suoritettiin sarja mittauksia, joiden avulla oli tarkoitus selvittää, kuinka pientä liikettä järjestelmällä kyetään mittaamaan. Mittauksia tehtiin järjestyksessä 0,5, 0,1, 0,05 ja 0,01 millimetrin siirroilla. Selvisi, että 0,01 millimetrin mittauksessa kohinan suurin vaihteluväli oli lähes puolet liikutusta matkasta. Kohinan suuruudesta johtuen tämä alkaa olla raja, mihin mittaustarkkuuteen järjestelmä kykenee. Kuvassa 24 on esitetty 0,01 millimetrin siirron mittauksen tulokset.



Kuva 24. 0,01 millimetrin siirron mittaustulokset

Kohinan määrää voidaan vähentää laskemalla mittaukselle liukuva keskiarvo. Tämä tarkoittaa keskiarvon laskemista mittauksen datapisteiden osajoukoista, ja sen avulla voidaan poistaa nopeat, lyhyen aikavälin vaihtelut. Mittaukselle, jossa kohde liikkui 0,01 millimetriä, laskettiin liukuva keskiarvo. Saaduista tuloksista havaittiin, että sitä käyttämällä voidaan tarkastella vieläkin pienempää liikettä kuin 0,01 millimetriä. Tämä kuitenkin edellyttää, että liike on suhteellisen hidasta. Kuvassa 25 on esitetty mittaus, jonka tulokselle on laskettu liukuva keskiarvo 10 datapisteiden avulla.



Kuva 25. 0,01 millimetrin liike, liukuva keskiarvo laskettu 10 pisteen avulla

Näiden mittausten perusteella voidaan todeta, että järjestelmällä voidaan mitata alle 0,01 millimetrin vaihteluita. Tämä kuitenkin vaatii liukuvan keskiarvon käyttämistä, jolloin nopea värähtely voi jäädä huomaamatta. Esimerkiksi jos kuvausnopeus on 400 kuvaa sekunnissa, ja liukuva keskiarvo lasketaan 10 datapisteen avulla, saadaan sekunnin ajalta 40 datapistettä. Nyquistin teoreeman mukaisesti tämän esimerkin näytteistystaajuudella voidaan mitata vain alle 20 hertsin värähtelyä. Ilman liukuvaa keskiarvoa datapisteiden mittausepävarmuus kohinan takia harmaasävyanalyysissä on noin $\pm 2,5$ mikrometriä ja binäärianalyysissä noin $\pm 3,4$ mikrometriä.

7 YHTEENVETO

Insinööriyön tavoitteena oli luoda konenäköön perustuva järjestelmä, joka mittaa liiketietoa. Ideana kameratekniikan käyttämisessä oli kehittää menetelmä, jonka avulla voidaan suorittaa mittaus koskettamatta kohdetta. Kuvaus suoritettiin CMOS-kameralla, jolla päästiin kuva-alueita rajaamalla parhaimmillaan 2000 kuvan kuvausnopeuteen, joskin mittauksissa käytettiin 400 kuvan nopeutta. Järjestelmän mittausepävarmuudeksi kaavailtiin mikrometrejä.

Työssä toteutettiin ohjelmisto, jonka tehtävänä oli konfiguroida kamera, suorittaa mittaus, analysoida mittaus sekä esittää tulokset. Ohjelmisto laadittiin National Instrumentsin LabVIEW-ohjelmointiympäristössä. Lisäksi apuna käytettiin Vision Assistant -ohjelmaa konenäköskriptien luomiseen. Kohteena käytettiin metallilevyä, jossa oli kaksi 0,6 mm etäisyydellä olevaa rakoja. Valaistukseen käytettiin diffusoitua taustavaloa, jonka avulla saavutettiin erinomainen kontrasti. Todellisuudessa kohde olisi ollut muodoltaan samankaltainen, laadukas tuloste, ja valaistuksessa olisi käytetty taustavalaistusta.

Työ onnistui pääpiirteittäin hyvin. Ohjelmiston kehitystyön hitaudesta johtui, ettei kaikkia työssä asetettuja tavoitteita ehditty saavuttaa. Menetelmää on mahdollista käyttää liiketiedon mittaukseen. Se kuitenkin vaatii jatkokehitystä, sillä tällaisenaan se ei ole valmis mittausjärjestelmä.

LÄHTEET

- 1 EPIC Systems Inc, Quick History of Machine Vision [WWW-julkaisu]
<http://www.epicsysinc.com/blog/machine_vision_history> (Luettu 15.1.2014)
- 2 ESA, The most important alarm clock in the solar system [WWW-julkaisu]
<http://www.esa.int/Our_Activities/Space_Science/Rosetta/The_most_important_alarm_clock_in_the_Solar_System> (Luettu 6.3.2014)
- 3 Romppainen Pentti, Konenäön luentomateriaali. Kajaanin ammattikorkeakoulu.
- 4 Gonzales, R., Woods, R., 2002, Digital Image Processing, Second edition, Upper Saddle River, Prentice-Hall Inc, ISBN: 0-201-18075-8
- 5 Olympus Microscopy, Introduction to CMOS Image Sensors [WWW-julkaisu]
<<http://www.olympusmicro.com/primer/digitalimaging/cmosimagesensors.html>>
(Luettu 10.2.2014)
- 6 Teledyne Dalsa, CCD vs. CMOS [WWW-julkaisu]
<<http://www.teledynedalsa.com/imaging/knowledge-center/appnotes/ccd-vs-cmos/>> (Luettu 5.3.2014)
- 7 Molecular expressions, Anatomy of a Charge-Coupled Device [WWW-julkaisu]
<<http://micro.magnet.fsu.edu/primer/digitalimaging/concepts/ccdanatomy.html>>
(Luettu 10.2.2014)
- 8 IHS GlobalSpec, CMOS Cameras Information [WWW-julkaisu]
<http://www.globalspec.com/learnmore/video_imaging_equipment/video_cameras_accessories/cmos_cameras> (Luettu 5.3.2014)
- 9 Vision Systems Design, Camera Connections [WWW-julkaisu] <<http://www.vision-systems.com/articles/print/volume-13/issue-4/features/integration-insights/camera-connections.html>> (Luettu 8.3.2014)
- 10 AIA Vision Online, GigE Vision - True Plug and Play Connectivity [WWW-julkaisu]
<<http://www.visiononline.org/vision-standards-details.cfm?type=5>> (Luettu 9.3.2014)
- 11 1394 Trade Association, 1394 Trade Association press release [WWW-julkaisu]
<http://www.1394ta.com/press/TAPress/2008_0709.html> (Luettu 9.3.2014)

- 12 AIA Vision Online, USB3 Vision - High Bandwidth Yet Simple Connectivity [WWW-julkaisu] <<http://www.visiononline.org/vision-standards-details.cfm?type=11>> (Luettu 9.3.2014)
- 13 AIA Vision Online, Camera Link – The Only Real-Time Machine Vision Protocol [WWW-julkaisu] <<http://www.visiononline.org/vision-standards-details.cfm?type=6>> (Luettu 9.3.2014)
- 14 CoaXPress, What is CoaXPress? [WWW-julkaisu] <<http://www.coaxpress.com/coaxpress.php>> (Luettu 15.3.2014)
- 15 EMVA, GenICam™ standard [WWW-julkaisu] <<http://www.emva.org/cms/index.php?idcat=27>> (Luettu 2.4.2014)
- 16 Saari, M., Valokuvauksen perusteita: aukko [WWW-julkaisu] <<http://www.mikkosaari.fi/aukko/>> (Luettu 18.4.2014)
- 17 Saari, M., Valokuvauksen perusteita: makrokuvaus [WWW-julkaisu] <<http://www.mikkosaari.fi/makrokuvaus/>> (Luettu 21.4.2014)
- 18 Microscan, Dark Field Illuminators [WWW-julkaisu] <<http://www.microscan.com/en-us/products/NERLITE-Machine-Vision-Lighting/Dark-Field-Illuminators.aspx>> (Luettu 15.2.2014)
- 19 Multipix, Doal lighting [WWW-julkaisu] <<http://multipix.com/products/machine-vision-lighting/on-axis-lighting/doal-lighting/>> (Luettu 15.2.2014)
- 20 Machine Design, Cameras at work [WWW-julkaisu] <<http://machinedesign.com/technologies/cameras-work>> (Luettu 3.5.2014)
- 21 Romppainen Pentti, Konenäkö digitaalinen kuvankäsittely. Kajaanin ammattikorkeakoulu.
- 22 OpenCV, OpenCV (Open source computer vision) [WWW-julkaisu] <<http://opencv.org/>>
- 23 Itseez, OpenCV [WWW-julkaisu] <<http://itseez.com/OpenCV/>> (Luettu 26.2.2014)
- 24 LabVIEW [WWW-julkaisu] <<http://en.wikipedia.org/wiki/LabVIEW>> (Luettu 26.2.2014)
- 25 MVTec Halcon [WWW-julkaisu] <<http://www.halcon.com/>> (Luettu 26.2.2014)

- 26 Basler, User's manual for GigE cameras [WWW-julkaisu]
<http://www.cosyco.de/kameras/flaechenkameras/basler/xxlpdf/ace_GigE_Handbuch.pdf> (Luettu 1.5.2014)
- 27 NetworkWorldFusion, Alteon still stumping of Jumbo Frames [WWW-julkaisu]
<<http://www.networkworld.com/news/1022alteon.html>> (Luettu 5.1.2014)
- 28 Halvorsen, H., Introduction to LabVIEW [WWW-julkaisu]
<<http://home.hit.no/~hansha/documents/labview/training/Introduction%20to%20LabVIEW/Introduction%20to%20LabVIEW.pdf>> (Luettu 29.3.2014)
- 29 Bitter, R., Mohiuddin, T., Nawrocki, M., 2007, LabVIEW Advanced Programming Techniques, Boca Rato, Taylor & Francis Group LLC, ISBN: 0-8493-3325-3
- 30 National Instruments, NI Vision Assistant Tutorial [WWW-julkaisu]
<<http://physics.ucsd.edu/neurophysics/Manuals/National%20Instruments/NI%20Vision%20Assistant%20Tutorial.pdf>> (Luettu 18.4.2014)