

**Kari Korpi**

**LIKKUVAN LAITTEEN KUSTANNUSTEHOKAS  
KONENÄKÖRATKAISU**

**Opinnäytetyö  
CENTRIA AMMATTIKORKEAKOULU  
Tuotantotalouden koulutusohjelma  
Huhtikuu 2016**

**TIIVISTELMÄ OPINNÄYTETYÖSTÄ**

<b>Yksikkö</b>	<b>Aika</b>	<b>Tekijä/tekijät</b>
YLIVIESKA	Huhtikuu 2016	Kari Korpi
<b>Koulutusohjelma</b>		
Tuotantotalous		
<b>Työn nimi</b>		
Liikkuvan laitteen kustannustehokas konenäköratkaisu		
<b>Työn ohjaajat</b>		<b>Sivumäärä</b>
Jari Kaarela ja Sakari Pieskä		24+1
<p>Konenäkö on tekniikka, jota voidaan hyödyntää tulevaisuudessa yhä uusilla aloilla. Opinnäytetyössä tehtiin konenäköä hyödyntävä lisäys liikkuvaan laitteeseen. Tätä varten etsittiin harrastajan rahavaroille sopiva kamera, jolla tehtävä voitiin suorittaa. Kun valmiina ei ollut liikkuvaa laitetta, myös sellainen tehtiin opinnäytetyön ohella.</p> <p>Teoriaosuudessa käsitellään konenäköön, mikro-ohjaimiin ja niiden väliseen tiedonsiirtoon liittyviä aiheita, joita tässä opinnäytetyössä tarvittiin.</p> <p>Opinnäytetyölle ei ollut tilaajaa.</p> <p>Opinnäytetyössä päästiin tavoitteeseen: edullisella konenäkökameralla saatiin autonomiseen laitteeseen uusi aisti.</p>		
<b>Asiasanat</b>		
Arduino, konenäkö, ohjelmointi, SPI		

## ABSTRACT

<b>Unit</b>	<b>Date</b>	<b>Author</b>
Ylivieska	April 2016	Kari Korpi
<b>Degree programme</b>		
Industrial Management		
<b>Name of thesis</b>		
Cost-effective machine vision solution for a mobile device		
<b>Instructor</b>		<b>Pages</b>
Jari Kaarela and Sakari Pieskä		24+1
<p>Machine vision is a technology that in the future will find a lot of new applications. In this thesis machine vision was utilized in a mobile device. For that purpose I had to find a camera that was possible to buy for an average enthusiast. Because I didn't have any mobile device, I had to built that to alongside of the thesis.</p> <p>The theoretical part deals with machine vision, topics related to micro-controllers and communication exchange between them.</p> <p>The thesis had no orderer.</p> <p>The thesis goal was achieved: an affordable machine vision camera gave the mobile device a new sense.</p>		

### Key words

Arduino, machine vision, programming, SPI

## KÄSITTEIDEN MÄÄRITTELY

---

### A/D-muunnin

Elektroniikassa laite, jolla muunnetaan analoginen signaali digitaaliseen muotoon lukuarvoksi.

### Arduino/Genuino

Harrastajien suosimien mikro-ohjainalustojen valmistaja.

CCD Charge-Coupled Device, valoherkkä kenno, jota käytetään konenäkökameroissa.

CMOS Complementary Metal Oxide Semiconductor, mikropiiritekniikka, jota käytetään CCD- tekniikan tilalla edullisemmissä kameroissa.

### D/A-muunnin

Elektroniikassa laite, jolla muunnetaan numeroarvoinen digitaalidata analogiseksi signaaliksi.

EEPROM Electronically Erasable Programmable Read-Only Memory, sähköisesti tyhjennettävä ja ohjelmoitava ROM-muisti.

EPROM Ultravioletilla tyhjennettävä, uudelleen ohjelmoitava ROM-muisti.

FLASH Sähköisesti kirjoitettava ja tyhjennettävä muistipiiri.

IDE Integrated development environment, samaan pakettiin integroitu ohjelmointiympäristö.

I/O-portti Mikro-ohjaimen nasta, jota ohjataan tai luetaan ohjelmallisesti mikro-ohjaimen rekistereistä.

Library Ohjelmakirjasto, sisältää yleensä toiminnallisuuden tietylle asialle.

### Mikro-ohjain

Sulautetuissa sovelluksissa käytettävä prosessori.

### Pseudokoodi

Ohjelmointikielen tapainen koodi, joka ei ole kuitenkaan minkään ohjelmointikielen mukaista.

RAM Random access memory, tavallinen tietokoneissa käytettävä luku- ja kirjoitusmuisti.

ROM Read only memory, muistityyppi, joka on kertaalleen ohjelmoitavissa eikä ole tyhjennettävissä.

SPI Serial Peripheral Interface, synkroninen tiedonsiirtoväylä.

Tilakone Ohjelmarakenne, jossa on toimintoja eri tiloille.

WLAN Wireless Local Area Network, langaton lähiverkkotekniikka.

## ESIPUHE

Maailmassa on jos jonkinlaisia koneita, alkaen yksinkertaisesta vivusta ja päätyen Mars-planeetalla kulkevaan robottiin. On paljon koneita, joiden suorituskykyä voisi laajentaa lisäämällä niihin uuden aistin. Mitäpä jos vaikkapa kiviä maasta irti väännettäessä käytettävässä vivussa olisi voima-anturi? Tämän anturin antaman tiedon perusteella tiedettäisiin, ollaanko ylittämässä vivun kesto, jotta ei suotta katkaistaisi vipua ylikuormituksella. Säästyisi paljon aikaa ja vaivaa muuhun tarpeelliseen toimintaan. Tai jos on lelu, jota voi ohjata puhekomennoin ”Mene eteenpäin”, ”Käännä vasemmalle” ja lelu liikkuu käskyn mukaan. Olisi paljon mielekkäämpää, jos lelulle voisi sanoa ”Etsi punainen pallo ja vie se laatikkoon”. Mitäpä jos tässä lelussa olisi aisti, jonka perusteella se osaisi tehdä tuon? Ilmiselvästi tuohon tarvitaan näköaisti. Muutoin on vaikea tunnistaa, mikä ja missä on pallo tai minkä värinen se on. Näköaisti on ihmiselle luontainen, ihmisellä on silmät. Koneella ei ole silmiä, mutta siihen voidaan liittää kamera, jonka avulla koneen on mahdollista nähdä.

**TIIVISTELMÄ**

**ABSTRACT**

**KÄSITTEIDEN MÄÄRITTELY**

**ESIPUHE**

**SISÄLLYS**

<b>1</b>	<b>JOHDANTO</b>	<b>1</b>
<b>2</b>	<b>Ihmisen näköaisti ja konenäkö</b>	<b>2</b>
2.1	Ihmisen näköaisti	2
2.2	Koneen näköaisti	4
2.2.1	Konenäkö älykameralla	5
2.2.2	Konenäkö ja valaistus	5
<b>3</b>	<b>AUTOMAATIOSSA KÄYTETTÄVIÄ ÄLYKAMERARATKAISUJA</b>	<b>6</b>
3.1	Cognex	6
3.2	Omron	7
3.3	Pixy	8
<b>4</b>	<b>MIKRO-OHJAIN</b>	<b>10</b>
4.1	Arduino/Genuino	10
4.2	Lego Mindstorms	11
<b>5</b>	<b>TIEDONSIIRTO MIKRO-OHJAIMEN JA KAMERAN VÄLILLÄ</b>	<b>14</b>
5.1	SPI-väylä	14
5.2	Arduino ja Pixy kommunikoivat SPI-väylällä	15
<b>6</b>	<b>LAITTEEN SUUNNITTELU JA TOTEUTUS</b>	<b>16</b>
6.1	Suunnittelu	16
6.2	Kameran valinta	16
6.3	Kameran mekaaninen liittäminen laitteeseen	17
6.4	Elektroniikan kytkeminen	17
6.5	Ohjelmointi	19
<b>7</b>	<b>TULOKSET JA POHDINTAA</b>	<b>20</b>
7.1	Tulokset	20
7.2	Pohdintaa	21
7.3	Jatkokehitys	21
7.3.1	Valaistus	21
7.3.2	Ohjausohjelma	22
	<b>LÄHTEET</b>	<b>23</b>
	<b>LIITTEET</b>	

## **KUVIOT**

KUVIO 1. Pixy	1
KUVIO 2. Silmän rakenne	2
KUVIO 3. Tappi- ja sauvasolut)	3
KUVIO 4. Ihmissilmän, CMOS- ja CCD-kennon herkkyys eri allonpituuksille	4
KUVIO 5. In-Sight 2000-Series	6
KUVIO 6. FQ-M	7
KUVIO 7. Pixy	8
KUVIO 8. Arduino Pro Mini	11
KUVIO 9. Lego RCX	12
KUVIO 10. Lego Minstorms NXT	12
KUVIO 11. Lego Mindstorms EV3	13
KUVIO 12. SPI-väylän rakenne	14
KUVIO 13. 3D-malli tehdyistä osista	17
KUVIO 14. Pixyn liitännät	18
KUVIO 15. Arduino Nano	18
KUVIO 16. Lattakaapeli	18
KUVIO 17. Valmis laite muutoksineen	20



## 1 JOHDANTO

Työssä ja harrastuksissa olen rakentanut erilaisia mekaanisia ja elektronisia laitteita sekä ohjelmoinut erilaisia tietokone-, kännykkä- ja sulautettuja sovelluksia. Konenäkö on kuitenkin yksi osa-alue, johon en ollut syventynyt opintoihin sisältyvää kurssia enempää. Aiheena konenäkö on hyvin kiinnostava, joten päätin perehtyä siihen alueeseen tarkemmin. Tässä opinnäytetyössä keskityn konenäön liittämiseen laitteeseen ja siltä saatavan informaation käyttämiseen laitteen ohjaamiseen.

Lähdin etsimään harrastajan rahavaroille sopivaa konenäkökameraa. Charmed Labs toi markkinoille Pixy-nimisen älykameran, joka voidaan helposti opettaa tunnistamaan kohteita. Tällä tunnistamisella on rajoitteensa, joista lisää myöhemmin. Kuviossa 1 on Pixy sellaisenaan kuin se toimitetaan.



KUVIO 1. Pixy (Charmed Labs 2016)

Konenäkökameran löydyttyä päätin, että se liitetään osaksi jotain liikkuvaa laitetta. Laitteeksi rakensin Härpäkkeen. Suunnittelin sen SolidWorksilla ja tulostin muoviosat 3D-tulostimella. Lisäksi tarvittiin ajomoottorit, pyörät ja mikro-ohjain. Härpäke on kolmipyöräinen itseliikkuva laite, jota voi ohjata puheella, joka osaa vastata puhuen sekä pyydettyä soittaa musiikkia. Härpäkkeelle annoin oman alueen, "hiekkalaatikon", jossa se saa liikkua. Tätä varten tein levyrakenteisen valkoisen laatikon, jonka koko on 1200 mm x 900 mm x 300 mm. Härpäkkeen tai 'hiekkalaatikon' rakentamiseen tässä opinnäytetyössä ei syvennytä sen enempää.

## 2 Ihmisen näköaisti ja konenäkö

### 2.1 Ihmisen näköaisti

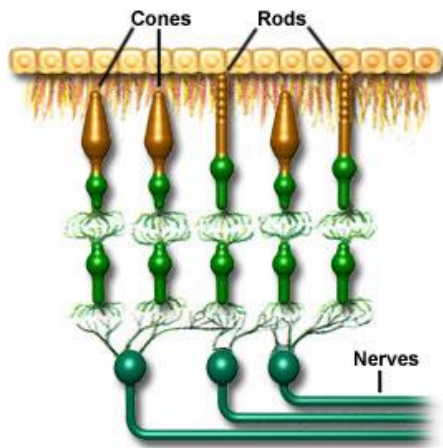
Ihmisen silmä muodostaa kuvan näkökentästä verkkokalvolle. Valo kulkee pupillista mykiön eli silmän linssin läpi, joka yhdessä sarveiskalvon kanssa taittaa valon verkkokalvolle. (Yleisradio Oy 2016.)

Kuviossa 2 on ihmisen silmän rakenne yksinkertaistettuna ja eri osat selventävästi väritettyinä.



KUVIO 2. Silmän rakenne (Näkövammaisten liitto 2016)

Mykiö heijastaa valon verkkokalvolle niin, että kuva on ylösalaisin. Pupilli supistuu ja laajenee valon määrän mukaan. Verkkokalvolla on valoa aistivia tappi- ja sauvasoluja, joista havainnekuva Kuviossa 3. Tappisolut aktivoituvat valoisassa ja aistivat värejä, kun taas sauvasolut ovat aktiivisia hämäränäkemisessä. Verkkokalvolta valo kulkee näköhermoa pitkin aivojen näköalueelle, jossa kuva tulkitaan oikeinpäin. (Yleisradio Oy 2016.)



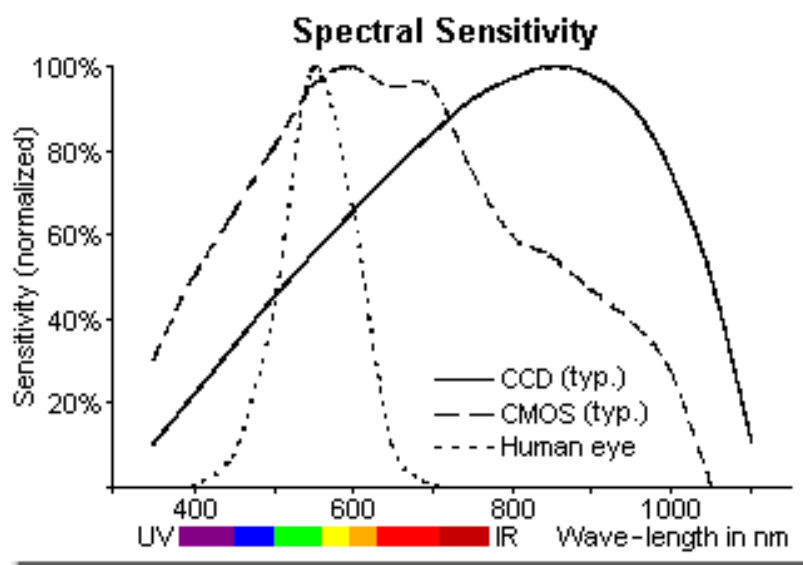
KUVIO 3. Tappi- ja sauvasolut (National High Magnetic Field Laboratory 2016)

Ihmisen silmät ovat herkkiä 380nm-780nm välillä oleville aallonpituuksille. Verkkokalvolla olevia tappisoluja on kolmenlaisia ja nämä reagoivat eri aallonpituuksille (420nm, 530nm, 560nm). Silmän iiris säätyy valaistuksen mukaan. Verkkokalvolle tulleen valon tappi- ja sauvasoluihin aiheuttama ärsyke välittyy näköhermon välityksellä näköaivokuorelle, jonka jälkeen aivot tulkitsevat informaation. (Helsinki University Central Hospital 2016.)

## 2.2 Koneen näköaisti

Kamerassa käytettävän kuvailmaisimen herkkyys eri aallonpituuksille on huomattavasti laajempi kuin ihmissilmän (Walter Preiss Technology Systems 2016).

Kuviossa 4 verrataan ihmissilmän, CCD- ja CMOS-kennon herkkyyttä eri aallonpituuksilla.



KUVIO 4. Ihmissilmän, CMOS- ja CCD-kennon herkkyys eri aallonpituuksille (Walter Preiss Technology Systems 2016)

Konenäköjärjestelmän kameroissa käytetään CCD- tai CMOS-kennoa. CCD-kennolta luettava kuva saadaan analogisessa muodossa ja lukemisen jälkeen informaatio muutetaan digitaaliseksi. CMOS-kennosta kuvainformaatio saadaan suoraan digitaalisena. (Dasys Oy 2015.)

Konenäköllä voidaan tunnistaa kohteen sijainti. Sen avulla voidaan esim. etsiä virheitä kohteesta ja mitata kohde. Konenäköä voidaan käyttää myös tarkistamiseen, sisältääkö kohde halutut asiat. (Metropolia 2016.)

### **2.2.1 Konenäkö älykameralla**

Älykamerajärjestelmä toimii stand-alone -järjestelmänä, joten se ei tarvitse ulkoisia komponentteja kuvan prosessointiin tai tulkitsemiseen. Älykameralle kerrotaan, yleensä erillisellä ohjelmistolla, miten kameran ohjelmiston tulee prosessoida ja tulkita kuvaa. Yksinkertaisimmillaan älykameran opettaminen tapahtuu niin, että kameralle näytetään haluttu objekti ja kameran ohjelmisto osaa tämän jälkeen päätellä, onko aiemmin opetettu objekti kameran näköpiirissä. (Charmed Labs 2016.)

### **2.2.2 Konenäkö ja valaistus**

Konenäössä tarvitaan sovellukseen sopiva valaistus, jotta saadaan luotettavaa kuvainformaatiota. Jos valaistus on riittämätön, kuvan terävyyden kanssa on ongelmia. Jos taas valoa on liikaa, saattaa tästä aiheutua heijastumia ja myös varjoja, jotka haittaavat kuvan tulkitsemistä. Valaistus pitäisi saada mahdollisimman pysyväksi, jos valaistukseen tulee muutoksia, ei voida ennakoida, miten se vaikuttaa kuvainformaatioon. (Metropolia 2016.)

### 3 AUTOMAATIOSSA KÄYTETTÄVIÄ ÄLYKAMERARATKAISUJA

Seuraavaksi esitellään muutamia automaatioissa käytettäviä älykameraratkaisuja.

#### 3.1 Cognex

Cognex Corporation on älykameroiden osalta yksi johtavia valmistajia maailmassa. In-Sight 2000-sarja on Cognexin älykameroista edullisin. In-Sight 2000-sarjan kameroissa, joista esimerkki Kuviossa 5, on 1/3” yksivärinen CMOS-kenno. Kameran mallista riippuen resoluutio on 640x480 pikseliä tai 800x600 pikseliä. Vakiona In-Sight 2000-sarjan kameroissa on 8:lla ledillä toteutettu rengasvalaisin. (Cognex 2016.)



KUVIO 5. In-Sight 2000-Series (Cognex 2016)

### 3.2 Omron

Omronin FQ-M-Series kameroissa käytetään 1/3” kennoa. Kuvan käsittelyssä käytetään resoluutiota 752x480 pikseliä. Kuviossa 6 on esimerkki Omronin FQ-M-sarjan kamerasta. Omronilla on Sysmac Studio IDE, jolla voidaan hallita kameran ominaisuuksia. IDE:llä voi asettaa kameran asetukset, ohjelmoida ja debugata tehdyn sovelluksen. Sysmac Studio on saatavissa Windows- ympäristöön. (Omron 2016.)



KUVIO 6. FQ-M (Omron)

### 3.3 Pixy

Pixy on Charmed Labsin ja Carnegie Mellon Robotics Institutun kehittämä konenäkökamera (Charmed Labs 2016). Kuviossa 7 Pixy kamerapuolelta kuvattuna.



KUVIO 7. Pixy (Charmed Labs 2016)

Edullinen hinta sekä pieni koko ja helppo liitettävyys tekevät laitteesta suosittuun harrastajien keskuudessa. Pixyä voi hyödyntää Windowsissa, Macissa, Linuxissa, Raspberry Pi:ssä, Arduinossa ja monissa muissa alustoissa pyörivissä ohjelmissa. Pixystä on saatavilla myös versio, joka on liitettävissä LEGO NXT -ohjaimen. Pixyssä on sekä analoginen että digitaalinen ulostulo sekä useita liitäntöjä erilaisiin väyliin, kuten SPI, I2C, UART, USB. Liitäntöjä käytetään Pixyn ja jonkin muun laitteen väliseen tiedonsiirtoon. (Charmed Labs 2016.)

Pixylle voi opettaa tunnistettavat kohteet viemällä kohteen kameran näkökenttään ja kuvaamalla sen. Havaittavat kohteet voi opettaa myös PixyMon-sovelluksella, jossa videokuvasta otetaan pysäytyskuva ja hiirellä raahaamalla määritellään kuvasta alue, jossa on tunnistettava kohde. PixyMon-sovelluksesta on versiot Windowsiin, Maciin ja Linuxiin. (Charmed Labs 2016.)



Pixylle voidaan määritellä enintään seitsemän erilaista kohdetta, jotka se erottelee värin mukaan. Kohteen muodolla ei ole merkitystä. Näitä seitsemää erilaista kohdetta voi olla kameran näkökentässä samaan aikaan yhteensä satoja kappaleita. Pixyn kuvataajuus on 50 kuvaa sekunnissa ja resoluutio on 640x400 pistettä (Charmed Labs 2016).

## 4 MIKRO-OHJAIN

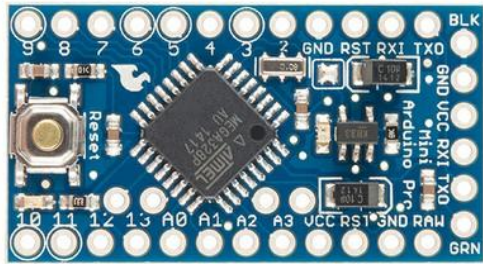
Mikro-ohjain, mikrokontrolleri, millä nimellä sitä kutsutaankin, poikkeaa perinteisestä prosessorista siten, että useimmiten siihen on sisäänrakennettuna ROM-, EPROM-, EEPROM- tai FLASH-ohjelmamuisti ja RAM-muisti. Mikro-ohjaimessa on usein myös I/O-portteja, A/D- ja D/A-muuntimia, ajastimia ja laskureita. (Vahtera 2003, 12-29.)

Mikro-ohjaimia käytetään kohteissa, joissa ei ole tarvetta vaihtaa suoritettavaa ohjelmaa, koska siihen ladattu ohjelma tekee sen, ja vain sen, mitä on haluttu. Jos on tarvetta muuttaa ohjelman toimintaa, ladataan uusi versio mikro-ohjaimen ohjelmamuistiin. (Vahtera 2003, 66-71.)

Mikro-ohjaimia käytetään esimerkiksi autoissa, television kauko-ohjaimissa, pesukoneissa ja mikroaaltouuneissa (Atmel 2015).

### 4.1 Arduino/Genuino

Arduino on avoimeen lähdekoodiin ja elektroniikkaan perustuva sovelluskehitysalusta. Arduinoja on useita erilaisia niin I/O-porttien määrän kuin prosessorin valmistajan osalta. Arduinot ohjelmoidaan Arduino-kielellä. Kieli pohjautuu Wiring-alustoille kehitettyyn Wiring-kieleen. Arduino IDE on kehitysympäristö, joka sisältää editorin, kääntäjän sekä mahdollisuuden ohjelman lataamiseen itse prosessoriin. Arduino IDE pohjautuu Processing-ympäristöön. Kymmenet yritykset ympäri maailmaa valmistavat Arduino-yhteensopivia kortteja, joita ne myyvät eri nimisinä. Mikään ei estä ketä tahansa tekemästä oman sovelluksen piirilevyineen ja prosessoreineen, jonka voi ohjelmoida Arduino-kielellä Arduino IDEn avulla. Arduino-tavaramerkistä on ollut oikeusprosessi, jonka aikana osa Arduinon kehittäjistä otti tuotteidensa tavaramerkiksi Genuinon. (Arduino 2016.) Kuviossa 8 on Arduino Pro Mini, yksi pienimmistä Arduino-alustoista.



KUVIO 8. Arduino Pro Mini (Arduino 2016)

Arduinon on hyvin laaja ohjelmakirjasto erilaisten antureiden ja muiden oheislaitteiden liittämistä varten. Useat yritykset ja vielä useammat harrastajat tekevät ja julkaisevat yhä uusille oheislaitteille kirjastoja, joita jokainen Arduino-käyttäjä voi hyödyntää omissa projekteissaan. (Arduino 2016.)

Arduinoissa käytetään mikro-ohjaimina Atmelin valmistamia AVR ja ARM mikro-ohjaimia. Arduino IDE:llä voi nykyisin tehdä sovelluksia myös muihin kuin Arduino-yhteensopiviin alustoihin. (Arduino 2016.)

## 4.2 Lego Mindstorms

LEGO:n eri ohjaimilla on voinut alusta lähtien saada LEGO-palikoista rakennetuille laitteille erilaisia ohjelmoituja toimintoja. Mindstorms-ohjaimet ohjelmoidaan graafisessa ympäristössä 'vedä ja pudota' -tyyppisellä ohjelmointitavalla. (Lego 2016.)

LEGO Mindstorms -sarjassa on julkaistu vuosien saatossa kolme erilaista ohjainlaitetta ja näistä useita versioita. Ensimmäinen ohjain, RCX, josta kuva Kuviossa 9, tuotiin markkinoille 1998 ja se oli ohjelmoitavissa Windows 95:ssä toimivalla ohjelmointiympäristöllä. Sovellus siirrettiin ohjainlaitteeseen infrapunalinkin välityksellä.



KUVIO 9. Lego RCX (Berea 2015)

RCX-ohjaimesta julkaistiin kaksi uutta versiota, RCX 1.5 vuonna 1999 ja RCX 2.0 vuonna 2001. Seuraavaksi, vuonna 2006, tuotiin markkinoille NXT-sarjan ensimmäinen versio. Vuonna 2009 NXT päivittyi versioon 2.0. Kuviossa 10 on nähtävissä ohjainyksikkö ja se, että liitännät ovat erilaiset kuin aiemmassa RCX-versiossa.



KUVIO 10. Lego Minstorms NXT (Lego 2015)

Vuonna 2013 LEGO julkisti toistaiseksi uusimman ohjaimen, EV3, joka on yhä edelleen saatavilla (Lego 2016). Kuviossa 11 on nähtävissä että ohjaimen liitännät eivät muuttuneet NXT-versiosta.



KUVIO 11. Lego Mindstorms EV3 (Lego 2016)

RCX 1.0:n anturi- ja toimilaittevalikoima oli suppea. Sarjassa tuli mukana valo-, kosketus- sekä lämpötila-anturi. Toimilaitteina oli kaksi moottoria. NXT-sarjan myötä tuli myös äänianturi sekä ultraäänellä toimiva etäisyysanturi. Uutena toimilaitteena oli servomoottori. Uusimpaan EV3:seen on saatavilla lisäksi väri-, infrapuna- sekä gyroanturi. Näiden lisäksi eri valmistajien toimesta on saatavilla muitakin antureita eri Mindstorms-versioille. (Lego 2016.)

Lego Mindstorms alustoissa on käytetty Renesas Electronics Corporationin valmistamaa mikro-ohjainta (RCX), Atmelin valmistamaa ARM7 mikro-ohjainta (NXT) ja uusimmassa (EV3) on mikro-ohjaimena ARM9 (Lego 2016).

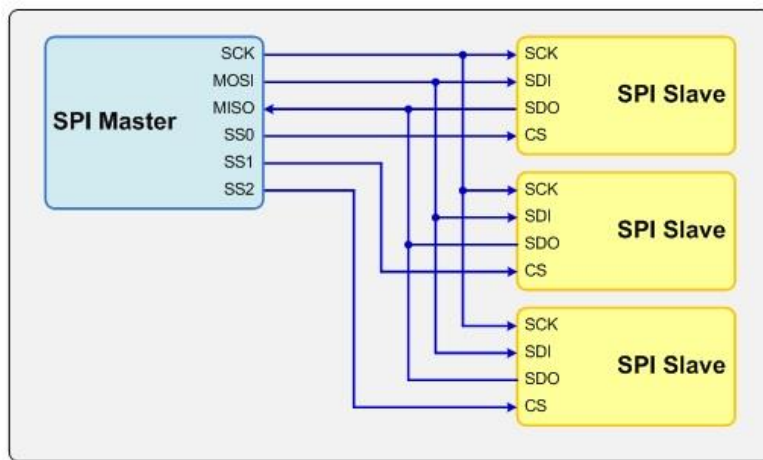
LEGO MINDSTORMS -sarjat ovat tässä mukana vaihtoehtona ohjaimiksi, sillä sekä NXT:hen että uusimpaan EV3:seen voi liittää myös Pixy-kameran. En kuitenkaan valinnut ohjaimeksi tätä, koska yksi asettamani tavoite, edullinen hinta, ei enää olisi toteutunut. Vaikka vuosien saatossa Mindstorms-sarjojen hinta on reaalisesti laskenut, maksaa uusin sarja, EV3, kuitenkin n. 350 € (Verkkokauppa.com 2016).

## 5 TIEDONSIIRTO MIKRO-OHJAIMEN JA KAMERAN VÄLILLÄ

### 5.1 SPI-väylä

Serial Peripheral Interface (SPI) -väylä siirtää tietoa synkronoidusti sarjamuotoisena. Tietoa voidaan siirtää samaan aikaan kahteen suuntaan (full duplex). SPI:tä kutsutaan joskus myös nelijohtimiseksi sarjaliikenteeksi. Liitännästä on myös 3-johtiminen ja moni-IO versiot. (Corelis 2016.)

Liikenne tapahtuu kahden laitteen välillä, joista toinen on "master" ja toinen "slave". Yksi "master" voi kommunikoida usean "slave":n kanssa. "Slave" valitaan ss-linjalla ja sen jälkeen kommunikointi kohdistuu tähän "slave":een. (Corelis 2016.) Kuviossa 12 on esimerkkikuva yhden "Masterin" ja kolmen "Slaven" kytkennästä.



KUVIO 12. SPI-väylän rakenne ( Corelis 2016)

SCK, eli Serial clock, on liitäntäpiste, jonka kautta "master" ohjaa "master":in ja "slave":n välillä tapahtuvaa tiedonsiirtoa. SCK:sta käytetään joskus myös lyhenteitä SCLK tai CLK. (Corelis 2016.)

MOSI, eli Master Out Slave In, on liitäntäpiste, josta välitetään siirrettävät bitit "slave":lle "slave":n datainput-liitäntäpisteeseen. MOSI:sta käytetään joskus myös lyhenteitä SIMO, DI, DIN, SI, MTST. (Corelis 2016.)

MISO, eli Master In Slave Out, on liitäntäpiste, josta välitetään siirrettävät bitit "slave":lta "master":ille. MISO:sta käytetään joskus myös lyhenteitä DO, DOUT, SO, MRSR. (Corelis 2016.)

SS, eli Slave Select, ovat liitäntäpisteitä, joilla valitaan haluttu slave. SS:stä käytetään joskus myös lyhenteitä nCS, CD, CSB, CSN, EN, nSS, STE, SYNC. (Corelis 2016.)

## **5.2 Arduino ja Pixy kommunikoivat SPI-väylällä**

Arduino SPI-kirjasto tulee Arduino-asennuspaketin mukana. Kirjastossa on valmiit funktiot yhteyden alustamiseen, tiedonsiirtoon ja yhteyden sulkemiseen. (Arduino 2016.)

Arduinon ja Pixyn väliseen kommunikointiin löytyy valmis kirjasto. Kirjastossa on funktio 'getBlocks', jolla kameran havaitsemat kohteet saa kerralla siirrettyä Arduinon muistiin. (Charmed Labs 2016.)

## 6 LAITTEEN SUUNNITTELU JA TOTEUTUS

### 6.1 Suunnittelu

Kun työn tarkoitus oli etsiä konenäkökamera ja liittää se liikkuvaan laitteeseen, Härpärkeeseen, työn suunnittelu oli suoraviivaista:

- etsitään sopiva kamera
- tehdään tarvittavat mekaaniset rakenteet kameras kiinnittämiseksi
- sähköinen kytkentä kameras ja Arduinon välille
- ohjaavan ohjelman muutokset.

### 6.2 Kameran valinta

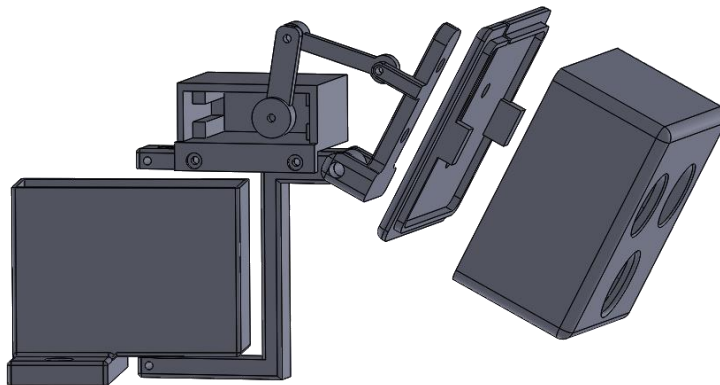
Markkinoilla on tarjolla hyvin monenlaisia älykameraratkaisuja. Lähes kaikissa on yksi ongelma, hinta. Älykameroiden hinnat vaihtelevat sadoista euroista tuhansiin euroihin. Etsin Internetistä harrastajan rahavaroille sopivaa kameraa, kunnes lopulta löysin saksalaisen Watterott-verkkokaupan sivuilta Pixyn. Pixyn hinta oli tuossa verkkokaupassa tilaushetkellä 68 € ja toimitus Suomeen maksoi 13 €. Tuolla hinnalla ei ollut saatavilla edes käytettyjä ns. parempia kameroita, joten valinta oli helppoa. Valitsin ainoan mahdollisen, eli Pixyn.

Heti selvisi, että edullinen kamera ei tietenkään vastaa ominaisuuksiltaan näitä kalliimpia teollisuudessa, lääketieteessä ja vaikkapa liikenteen seurannassa käytettäviä kameroita. Pixy kuitenkin kykenee erottelemaan eri väriset kohteet toisistaan. Se riittää tässä käytössä, koska kohteen muodot eivät ole tunnistamisen kriteerinä. Opinnäytetyön rajausta miettiessäni olin jo päättänyt, että älykameralla varustettu liikkuva laite tulee toimimaan 'hiekkalaatikossa', jossa pohja ja seinät ovat saman värisiä. Tämä poistaa sen mahdollisen häiriön, jossa joku taustalla oleva värikäs kohde aiheuttaa väärän tunnistamisen.



### 6.3 Kameran mekaaninen liittäminen laitteeseen

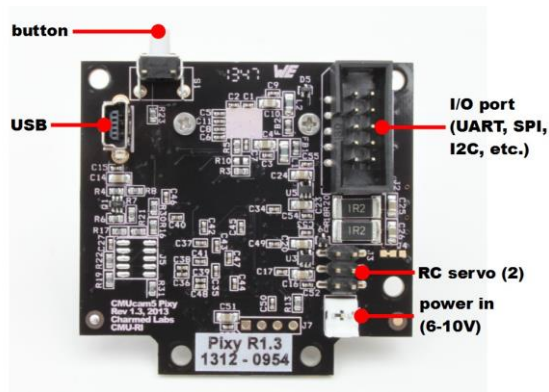
Jotta kameralla voisi havaita objekteja kameran kuva-alaa laajemmalla ympäristöstään, sen pitää kääntyä. Tein kameralle kahdesta servosta kahteen suuntaan kääntyvän telineen ja kiinnitin sen kahdella ruuvilla laitteeseen. Koska kamera on pelkkä piirilevy komponentteineen, sille tarvittiin myös suojakotelo. Suunnittelin tarvittavat osat SolidWorksilla ja lopuksi tulostin ne 3D-tulostimella. Kuviossa 13 on havainnekuva kameran liittämiseksi tehdyistä osista.



KUVIO 13. 3D-malli tehdyistä osista

### 6.4 Elektroniikan kytkeminen

Pixyn sähköinen kytkentä on yksinkertaista, koska kameran käyttöjännite ja SPI-väylä kulkevat samassa lattakaapelissa. Kiinnitin Pixyn mukana tulevan 6-johtimisen lattakaapelin kameran piirilevyllä olevaan I/O-liittimeen ja lattakaapelin toisen pään Arduino Nanon ICSP-liittimeen. Kuviossa 14 on Pixyn eri liitännät merkittyinä, Kuviossa 15 on Arduino Nano, jossa SPI-liitäntä on oikeassa päädyssä. Kuviossa 16 on Pixyn ja Arduino Nanon välille kytkettävä lattakaapeli.



KUVIO 14. Pixyn liitännät (Charmed Labs 2016)



KUVIO 15. Arduino Nano (Arduino 2016)



KUVIO 16. Lattakaapeli

Kytkein kameran kääntämisen mahdollistavat servot Arduino Nanon kahteen I/O-porttiin, käyttöjännitteeseen ja maahan. Pixyssä on liitännät kahdelle servolle. En kuitenkaan käyttänyt tätä mahdollisuutta, koska ajatuksena oli ohjata kameran liikkeitä itse omasta sovelluksesta.

## 6.5 Ohjelmointi

Härpäkkeessä oli valmiina ohjausohjelma, jolla sitä saattoi ohjata eteen, taakse, vasemmalle ja oikealle. Sen saksia sai avattua, suljettua, laskettua ja nostettua, eli niillä pystyi tarttumaan esineisiin.

Sen jälkeen, kun olin liittänyt kameran Härpäkkeeseen, tein muutoksia ohjausohjelmaan. Sovellus oli tehty Arduino IDE:ssä, Arduinon omalla kielellä, joten muutoksetkin tein samalla työkalulla ja samalla kielellä. Suurin muutos oli se, että lisäsin ohjelmaan tilakoneen, joka pitää kirjaa siitä, mitä laite on tekemässä. Tilakoneen pseudokoodi on liitteessä 1.

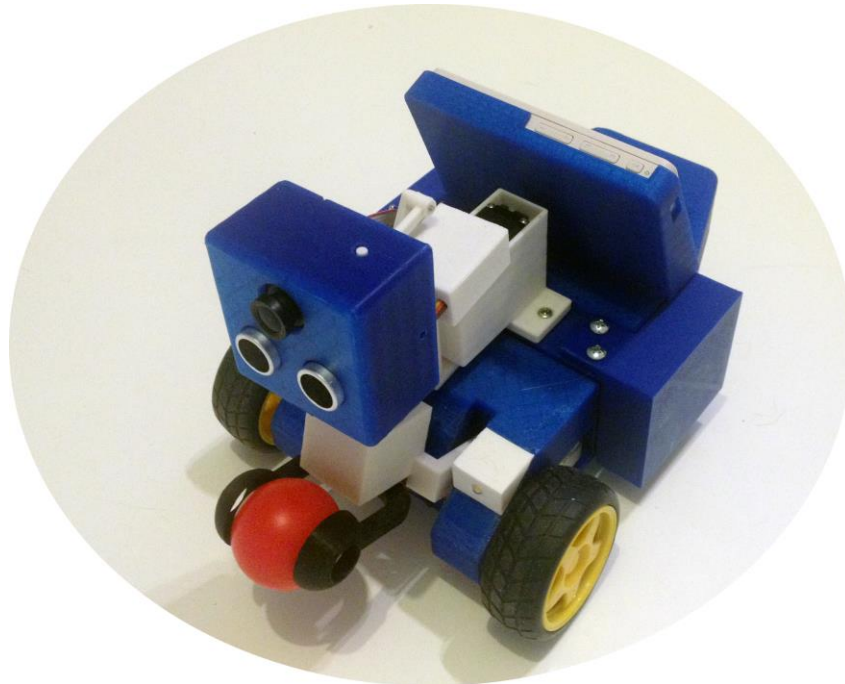
Tuohon tilakoneeseen voidaan vaikuttaa lähettämällä WLAN-yhteyden välityksellä ulkoisesta ohjaavasta laitteesta komentoja, kuten "PUNAINEN", "VIHREÄ", "SEIS" jne. Komennolla "PUNAINEN" Härpäke etsii punaisen pallon kääntymällä ensin paikallaan ja pallon tullessa kameran näkökenttään Härpäke liikkuu kohti palloa. Saavuttuaan pallon lähelle Härpäke laskee sakset alas ja avaa ne. Sen jälkeen se jatkaa lähestymistä hitaampaa vauhtia kohti palloa. Kun pallo on saksien välissä, se sulkee sakset ja nostaa niillä pallon ylös. Tämän jälkeen Härpäke kääntyy paikallaan, kunnes sen näkökenttään tulee kohde, johon pallo pitää viedä. Sen jälkeen Härpäke suuntaa kulkunsa kohteeseen ja ollessaan määrätyllä etäisyydellä kohteesta, se avaa sakset, jolloin pallo putoaa.

Kameralta saadaan tietoa siitä, mitkä objektit sen näkökentässä ovat. Se kertoo kohteiden koordinaatit kuva-avaruudessa, leveyden, korkeuden sekä tunniste-numeron. Samaan aikaan voidaan tunnistaa monia saman värisiä sekä eri värisiä kohteita. Jokaisesta tunnistetusta kohteesta saadaan ennen mainitut tiedot.

## 7 TULOKSET JA POHDINTAA

### 7.1 Tulokset

Sain rakennettua laitteen suunnitelmieni mukaan. Lisäsin olemassa olevaan laitteeseen, Härpäkkeeseen, konenäkökameran ja muutin laitteen ohjausohjelmaa siten, että se hyödyntää kameralta saatavan informaation. Kuviossa 17 on lopullinen laite kameroineen.



KUVIO 17. Valmis laite muutoksineen

## 7.2 Pohdintaa

Valitsemani kamera toimii tällaisessa käytössä oikein hyvin. Normaali huonevalaistus riittää, tosin valaistuksen muuttuessa tunnistamisen onnistuminen vaihtelee.

Konenäön liittäminen liikkuvaan laitteeseen oli yksinkertaista siihen nähden minkälainen työ oli rakentaa harrastajan välinein ja rahavaroilla liikkuva laite, johon kameran saattoi liittää. Vaikka kamera itsessään on kevyt, painaa se koteloineen ja kääntöservoineen niin paljon, että Härpäkkeen kokonaispaino kasvoi ajomoottorien voimien ylärajoille.

## 7.3 Jatkokehitys

Laitetta ja kameraa ohjaava sovellus toimii tässä siten kuin olin suunnitellut. Härpäke etsii, löytää, hakee ja kuljettaa pallon kohteeseen.

Kun asetetut reunaehdot oli täytetty, en lähtenyt laajentamaan toimivuutta. Tässä alla on ajatuksia siitä, kuinka toimivuutta voisi kehittää edelleen.

### 7.3.1 Valaistus

Jotta tunnistuksen saisi toimimaan paremmin erilaisissa olosuhteissa, valaistusta pitäisi parantaa. Kun kyseessä on liikkuva laite, valaistus voisi olla osana laitetta. Tällä saavutettaisiin se, että valaistus on laitteen asennosta riippumatta sen kameran näkökentässä lähes samanlainen.

### 7.3.2 Ohjausohjelma

Ohjelmaa voisi muuttaa siten, että servojen kameralle mahdollistamaa kääntymiskykyä hyödynnettäisiin enemmän. Nyt käytössä on vain ylös/alas-suuntainen liike ja sivuttainen kääntyminen hoidetaan kääntämällä koko laitetta.

Härpäke toimii omassa "hiekkalaatikossa", jossa etäisyydet eivät ole suuria. Se havainnoi etsintäaluettaan ainoastaan kääntymällä paikallaan. Jos etsintäaluetta haluttaisiin laajemmaksi, pitäisi ohjaussovellukseen lisätä toiminto, jonka ohjaamana Härpäke liikkuisi alueella etsien haluttua kohdetta.

## LÄHTEET

- Arduino. 2016. Learning. Saatavissa: <https://www.arduino.cc>. Luettu 4.8.2015.
- Atmel. 2015. Microcontrollers. Saatavissa: <http://www.atmel.com/products/microcontrollers>. Luettu 19.9.2015.
- Berea. 2015. Lego RCX. Saatavissa: <http://faculty.berea.edu>. Luettu 14.4.2016.
- Charmed Labs. 2015. Pixy. Saatavissa: <http://charmedlabs.com/default/pixycmucam5>. Luettu: 1.6.2015.
- Cognex. 2016. Älykamerat. Saatavissa: <http://www.metric.fi/tuotteet/konenako-anturit-ja-lasermerkinta/konenakoratkaisut/alykamerat>. Luettu 3.4.2016.
- Corelis. 2016. SPI Interface. Saatavissa: [http://www.corelis.com/education/SPI\\_Tutorial.htm](http://www.corelis.com/education/SPI_Tutorial.htm). Luettu 1.4.2016.
- Dasys Oy. 2015. Kamerat. Saatavissa: <http://www.dasys.fi/tuki-ohjeet/kamerat>. Luettu 14.9.2015.
- EngineersGarage. 2015. MicroControllers. Saatavissa: <http://www.engineersgarage.com/microcontroller>. Luettu 16.8.2015.
- Helsinki University Central Hospital. 2015. Näköaisti. Saatavissa: <http://www.biomag.hus.fi/braincourse/L5.html>. Luettu: 3.9.2015.
- Lego. 2016. Www-sivusto. Saatavissa: <http://www.lego.com>. Luettu 28.3.2016.
- Metropolia. 2016. Konenäkö – Machine Vision. Saatavissa: <http://wiki.metropolia.fi/download/attachments/11637671/Konenako.pdf>. Luettu 3.4.2016.
- National High Magnetic Field Laboratory. 2015. Saatavissa: <http://micro.magnet.fsu.edu/optics/lightandcolor/vision.html>. Luettu: 1.9.2015.
- Näkövammaisten liitto ry. 2016. Silmän rakenne. Saatavissa: <https://www.nkl.fi/fi/etusivu/nakeminen/rakenne>. Luettu 18.1.2016.

Omron. 2016. In-Sight 2000 Vision Sensors. Saatavissa:  
[www.cognex.com/products/machine-vision/in-sight-2000-series-vision-sensors](http://www.cognex.com/products/machine-vision/in-sight-2000-series-vision-sensors) .  
Luettu 4.4.2016.

Vahtera P. 2003. Mikro-ohjaimen ohjelmointi. 1. painos. Helsinki: WSOY.

Verkkokauppa.com. 2016. Hinnasto. Saatavissa:  
<https://www.verkkokauppa.com/fi/product/18889/dksxv/LEGO-Mindstorms-EV3-31313>. Luettu 28.3.2016.

Walter Preiss Technology Systems. 2016. SloMo CCD and CMOS Sensor Info.  
Saatavissa: <http://www.fen-net.de/walter.preiss/e/slomoinf.html>. Luettu: 15.3.2015.

Yleisradio Oy. 2016. Näköaisti. Saatavissa:  
<http://yle.fi/aihe/artikkeli/2007/01/18/nakoaisti>. Luettu: 20.3.2016.



## LIITE 1

### Ohjausohjelman pseudokoodi

Alustus

```
tila=lepää  
etsittävä=eiMitään
```

KunTuleeOhjausKomento

```
Jos komento=PUNAINEN niin  
    tila=etsi, etsittävä=punainen  
Jos komento=VIHREÄ niin  
    tila=etsi, etsittävä=vihreä  
Jos komento=SEIS niin  
    tila=lepää, etsittävä=eiMitään
```

Tilakone

```
Jos tila=etsi niin  
    Jos etsittävä näkyy niin tila=keskitä  
    Muutoin käänny vasemmalle  
Jos tila=keskitä niin  
    Käänny kohti etsittävää kunnes pallo on keskellä kameran  
    kuvaa  
    tila=lähesty  
Jos tila=lähesty niin  
    Aja eteenpäin kunnes etsittävä on kameran kuvassa  
    määrättyssä kohdassa  
    tila=otapallo  
Jos tila=otapallo niin  
    Laske sakset  
    Avaa sakset  
    Laske kameraa  
    Aja eteenpäin kunnes pallo on kameran kuvassa määrättyssä  
    kohdassa  
    Sulje sakset  
    Nosta sakset  
    tila=etsikohde  
Jos tila=etsikohde niin  
    Jos kohde näkyy niin tila=ajakohteeseen  
    Muutoin käänny vasemmalle  
Jos tila=ajaKohteeseen niin  
    Etsi kohde kameran kuvan keskelle  
    Lähesty kohdetta kunnes etäisyys on oikea  
    Avaa sakset  
    tila=lepää
```