

2014•2015  
FACULTEIT INDUSTRIËLE INGENIEURSWETENSCHAPPEN  
*master in de industriële wetenschappen: energie*

## Masterproef

Integratie van robotvolgsysteem met 2D-visie voor vlakke voorwerpen

Promotor :  
dr. ir. Johan BAETEN

Promotor :  
dr. ir. WIM PERSOONS

Bert Dingenen , Rob Mertens

*Scriptie ingediend tot het behalen van de graad van master in de industriële wetenschappen: energie*

Gezamenlijke opleiding Universiteit Hasselt en KU Leuven

2014•2015  
Faculteit Industriële  
ingenieurswetenschappen  
*master in de industriële wetenschappen: energie*

## Masterproef

Integratie van robotvolgsysteem met 2D-visie voor vlakke  
voorwerpen

Promotor :  
dr. ir. Johan BAETEN

Promotor :  
dr. ir. WIM PERSOONS

Bert Dingenen , Rob Mertens

*Scriptie ingediend tot het behalen van de graad van master in de industriële  
wetenschappen: energie*

# Voorwoord

Eerst en vooral willen we een dankwoord uitbrengen naar alle personen die ons ondersteund hebben voor de thesis. In de eerste plaats willen we onze externe promotor Wim Persoons bedanken die ons de mogelijkheid heeft gegeven onze thesis uit te voeren bij het bedrijf KUKA Automatisering + Robots N.V. te Houthalen. Ten tweede willen we onze interne promotor Johan Baeten bedanken voor zowel de technische als taalkundige ondersteuning. Ten derde nog een woord van dank voor de firma KUKA Automatisering + Robots N.V. alsook automatiseringscentrum ACRO die het benodigde materiaal aan ons ter beschikking heeft gesteld. Ten slotte bedanken we iedereen waarbij we terecht konden voor advies. De ervaring die we hebben opgedaan bij het plannen, uitvoeren en verslaggeven van de masterproef zal in onze carrière zeker nog van nut zijn.

Bert Dingenen & Rob Mertens



# Inhoudsopgave

Tabellenlijst.....	5
Figuurlijst .....	7
Verklarende woordenlijst.....	9
Abstract .....	11
Abstract English.....	13
Inleiding .....	15
1 Literatuurstudie.....	19
1.1 Optische eigenschappen.....	19
1.2 Sluiterijd.....	20
1.3 Herkennen van voorwerpen .....	21
2 Algemeen.....	23
2.1 Principiële werking.....	23
3 Algemene componenten.....	27
3.1 Ethernetcommunicatie.....	27
3.1.1 Visiesysteemzijde .....	27
3.1.2 Robotzijde.....	29
3.2 Bandsynchronisatie .....	31
3.3 Industriële robot .....	32
3.3.1 Inbedrijfname .....	32
3.3.2 Snelheidsmeting.....	34
3.3.3 Hoofdprogramma .....	36
4 Visiesysteem Cognex 7010.....	41
4.1 Inbedrijfnamemodus.....	41
4.2 Bedrijfsmodus.....	42
4.3 Specificaties Cognex 7010 .....	43
4.3.1 Cameramontage.....	43
4.3.2 Software.....	44
4.3.3 I/O-configuratie.....	48
4.4 Verbetermogelijkheden .....	49
4.4.1 Lenskeuze .....	49
4.4.2 Externe triggermogelijkheid .....	49
4.4.3 Robot – camera kalibratie .....	49
5 Visiesysteem IDS uEye-2230RE-C en Halcon .....	51

5.1	Algemeen.....	51
5.2	Specificaties visiesysteem IDS RE2330-C en Halcon .....	52
5.2.1	Cameramontage.....	52
5.2.2	Software.....	53
5.2.3	I/O-configuratie.....	58
5.3	Verbetermogelijkheden.....	58
5.3.1	Image acquisition.....	58
5.3.2	Externe triggermogelijkheid .....	59
5.3.3	Robot - camera kalibratie .....	59
6	Metingen .....	61
6.1	Nauwkeurigheid .....	61
6.1.1	Meetresultaten voorwerpen onder een vaste hoek .....	64
6.1.2	Meetresultaten voorwerpen onder een willekeurige hoek .....	65
6.1.3	Meetresultaten meerdere voorwerpen .....	68
6.1.4	Meetresultaten JPG-formaat als model.....	69
6.1.5	Meetresultaten DXF-formaat als model .....	70
6.2	Verwerkingsnelheid .....	70
6.2.1	Meetresultaten voorwerpen onder een vaste hoek .....	70
6.2.2	Meetresultaten voorwerpen onder een willekeurige hoek .....	71
6.2.3	Meetresultaten twee voorwerpen.....	72
6.2.4	Meetresultaten JPG-formaat als model.....	73
6.2.5	Meetresultaten DXF-formaat als model .....	73
6.3	Overige parameters .....	74
6.4	Vergelijkingsmatrix.....	76
7	Besluit .....	77
	Bibliografie .....	79
	Bijlagen .....	81

# Tabellenlijst

Tabel 1: Verklaring XML-tags uit de te verzenden data .....	28
Tabel 2: Verklaring belangrijkste XML-tags en hun parameters uit de XmlApiConfig.xml .....	30
Tabel 3: Gebruikte functies binnen het technologiepakket EthernetKRLXML [10] .....	37
Tabel 4: gebruikte functies binnen het technologiepakket ConveyorTech [10] .....	37
Tabel 5: Variabelen met de ontvangen meetgegevens in KRL.....	38
Tabel 6: Afstanden voor de grootte van het beeld in verschillende eenheden .....	39
Tabel 7: Specificaties Cognex 7010 [13].....	43
Tabel 8: Specificaties IDS uEye-2230RE-C camera en extern besturingssysteem [17].....	52
Tabel 9: Resultaten voor niet verdraaide voorwerpen .....	64
Tabel 10: Resultaten voor voorwerpen onder een variabele hoek voor de opstelling met het Cognex-systeem .....	65
Tabel 11: Resultaten voor voorwerpen onder een variabele hoek voor de opstelling met het IDS-systeem.....	66
Tabel 12: Resultaten voor twee voorwerpen onder een vaste hoek.....	68
Tabel 13: resultaten voor twee voorwerpen onder een vaste hoek voor het IDS-systeem .....	68
Tabel 14: Resultaten voor het zoeken op basis van JPG-formaat met het visiesysteem van Cognex .....	69
Tabel 15: Resultaten voor het zoeken op basis van JPG-formaat met het visiesysteem van IDS..	69
Tabel 16: Resultaten voor het zoeken van voorwerpen op basis van DXF-formaat met het visiesysteem van IDS.....	70
Tabel 17: Berekening maximale bandsnelheid voor niet verdraaide voorwerpen.....	70
Tabel 18: Berekening maximale bandsnelheid voor willekeurig verdraaide voorwerpen .....	71
Tabel 19: Berekening maximale bandsnelheid voor twee voorwerpen.....	73
Tabel 20: Berekening maximale bandsnelheid indien model aangeleerd met JPG-formaat.....	73
Tabel 21: Berekening maximale bandsnelheid in dien model aangeleerd met DXF-formaat.....	73
Tabel 22: Vergelijkingsmatrix tussen beide opstellingen .....	76





# Figuurlijst

Figuur 1: Schematische voorstelling van de opstelling [1] [2] [3] [4] .....	16
Figuur 2: Schematische voorstelling meetprocedure.....	18
Figuur 3: Optische parameters voor het ophangen van een camerasysteem [5] .....	19
Figuur 4: Vereenvoudigd lensmodel voor een convexe lens .....	20
Figuur 5: Beeld met: (a) een te hoge sluitertijd (b) een lage sluitertijd.....	21
Figuur 6: Bovenaanzicht transportband met het voorwerp: (a) binnen het beeldbereik (b) buiten het beeldbereik.....	22
Figuur 7: Principiële opbouw van de opstelling.....	23
Figuur 8: Flowchart principiële werking .....	24
Figuur 9: Schematische voorstelling van de verwerkingstijd .....	25
Figuur 10: Voorbeeld van een naamkaartje op de transportband .....	25
Figuur 11: XML-structuur voor verzenden.....	27
Figuur 12: XML-structuur voor verzenden van de alternatieve configuratie.....	28
Figuur 13: Schematische voorstelling KUKA-router.....	29
Figuur 14: XML-structuur van de XmlApiConfig.xml.....	30
Figuur 15: XML-structuur van de Cognex7010.xml .....	30
Figuur 16: XML-structuur van de Cognex7010.xml voor de alternatieve methode .....	31
Figuur 17: Resolver voor ConveyorTech (a) zijaanzicht [11] (b) gemonteerd op de transportband [11].....	31
Figuur 18: RDC2-kaart met aansluitingen (a) algemeen [11] (b) aangesloten [11] .....	32
Figuur 19: Verende pen als TCP.....	33
Figuur 20: Opmeten van de conveyorbasis.....	33
Figuur 21: Achterlopen robot t.g.v. verwerkingstijd .....	34
Figuur 22: flowchart snelheidsmeting sps.sub .....	35
Figuur 23: flowchart opmeten snelheidsvariabele voor hoofdprogramma .....	36
Figuur 24: Flow-chart hoofdprogramma.....	37
Figuur 25: Grafische weergave van het camera- en robotassenstelsel.....	39
Figuur 26: Aansluitschema visiesysteem Cognex7010 voor inbedrijfname .....	41
Figuur 27: Aansluitschema visiesysteem Cognex7010 voor bedrijf.....	42
Figuur 28: Visiesysteem Cognex 7010 [14] .....	43
Figuur 29: (a) optimale configuratie (b) alternatieve configuratie .....	43
Figuur 30: Intervaltijd voor het nemen van een beeld .....	44
Figuur 31: Logische functies om de coördinaten van de modellen in te vullen .....	46
Figuur 32: Invullen van de parameter k.....	47
Figuur 33: Berekenen van de parameters voor de te verzenden XML-string.....	47
Figuur 34: Totale XML-structuur .....	47
Figuur 35: Job-pass triggering.....	48
Figuur 36: Current sourcing aansluitprincipe [13] .....	48
Figuur 37: Verschillende coördinaatstelsels binnen de opstelling [16] .....	50
Figuur 38: Aansluitschema opstelling IDS uEye-RE2330-C.....	51
Figuur 39: IDS uEye-2230RE-C camera [17].....	52
Figuur 40: Flowchart voor het aanleren van een object .....	53
Figuur 41: Flowchart voor het zoeken van voorwerpen op de transportband .....	55
Figuur 42: Beelden nemen op de standaard manier [18] .....	57
Figuur 43: Open emitter type aansluiting van de I/O-aansluiting [19].....	58

Figuur 44: Image acquisition met asynchrone reset [20] .....	59
Figuur 45: Resultaten voor het bepalen van de onbekende bij te regelen bandafstand .....	62
Figuur 46: Voorbeeld van een meetresultaat .....	63
Figuur 47: Schematische voorstelling van de meetprocedure .....	63
Figuur 48: Meetresultaat (a) normaal (b) met apart getekende lijn .....	66
Figuur 49: Schematische weergave hoeknauwkeurigheid .....	67
Figuur 50: Hoogte van het naamkaartje voor verschillende hoeken .....	71
Figuur 51: Hoogst haalbare bandsnelheid in functie van het hoekbereik voor beide systemen...	72

# Verklarende woordenlijst

<b><i>Acquisition time</i></b>	De <i>acquisition time</i> is de totale tijd die een camera nodig heeft om zichzelf te triggeren en een beeld te nemen.
<b><i>Base</i></b>	Een <i>base</i> of basis is een coördinaatstelsel waarin de industriële robot zijn punten berekent.
<b>CCD</b>	CCD is de afkorting voor Charged-Coupled Device. Deze optische chip zet invallende lichtstralen om in een gedigitaliseerd beeld.
<b>CMOS</b>	CMOS is de afkorting voor Complementary Metal Oxide Semiconductor. Deze halfgeleider vormt de basis voor een optische chip met CMOS-technologie.
<b><i>Depth of field</i></b>	De <i>depth of field</i> of dof is een maat voor de hoogte van voorwerpen binnen het camerabeeld.
<b><i>Field of view</i></b>	De <i>field of view</i> is een andere benaming voor het volledige camerabeeld.
<b><i>Fixtures</i></b>	De eenheid <i>fixtures</i> is een andere benaming voor variabele afstand in pixels vermenigvuldigd met de maximum grootte in mm.
<b><i>Focal length</i></b>	De <i>focal length</i> is een andere benaming voor de brandpuntsafstand van een lens.
<b><i>Frame-rate</i></b>	Het door een camera maximaal te verwerken aantal beelden per tijdseenheid noemt <i>frame-rate</i> . Deze grootheid is meestal uitgedrukt in <i>frames per second</i> of fps.
<b><i>Loop</i></b>	Een <i>loop</i> is een oneindige lus binnen een applicatie.
<b>Module</b>	Een module is de benaming voor een applicatie binnen de KUKA Robot Language of KRL.
<b>Nulpositie</b>	De nulpositie van een industriële manipulator is de positie waarin de robot in rust staat.
<b>PLC</b>	PLC is de afkorting voor Programmable Logic Controller. PLC's zijn in de industrie gebruikt als aansturing van verschillende machines.
<b><i>Pointer</i></b>	Een pointer is een voorwerp om een bepaalde richting aan te duiden.
<b><i>Real-time</i></b>	Een <i>real-time</i> werkingsprincipe garandeert een directe ingreep op de omgeving zonder tijdsvertraging.
<b><i>Region of interest</i></b>	De <i>region of interest</i> of ROI is een deel van het beeld waarop beeldverwerking is toegepast.

<b>Sluittijd</b>	De sluitertijd is de tijd waaraan de optische chip van de camera blootgesteld is aan het omgevingslicht.
<b>Smart-camera</b>	Een smart-camera is een stand-alone visiesysteem met geïntegreerde processor voor beeldverwerking.
<b>String</b>	Een <i>string</i> is een bepaald type data dat bestaat uit een willekeurige reeks van tekens.
<b>Submit interpreter</b>	De <i>submit interpreter</i> is een KRL-applicatie die parallel naast de robotapplicatie draait en zichzelf continu herhaalt tegen snelle doorlooptijd.
<b>Tag</b>	Een <i>tag</i> is een benaming voor een bepaalde parameter binnen de XML-structuur.
<b>TCP/IP-protocol</b>	Het TCP/IP-protocol is bepaald protocol om datapakketten te versturen over een ethernetnetwerk.
<b>Tool</b>	Een <i>tool</i> is enerzijds een gereedschap dat is gemonteerd op de robotflens om bewerkingen mee uit te voeren en anderzijds een functie voor het herkennen van vormen, kleuren en patronen binnen een visiesysteem.
<b>Tool center point</b>	Het <i>tool center point</i> of TCP van een industriële robot is het werkpunt op de eindeffector.
<b>Working distance</b>	De <i>working distance</i> WD is een term voor de afstand tussen het werkoppervlak en de camerachip.
<b>XML</b>	XML is de afkorting voor Extensible Markup Language. Deze codetaal dient om gegevens gestructureerd op te slaan en te verwerken.

# Abstract

KUKA te Houthalen wil kennis verwerven in visiesystemen die bruikbaar zijn voor het volgen van vlakke delen op een transportband m.b.v. een industriële robot. Het doel van deze masterproef is om twee 2D-visiesystemen te implementeren, te testen en te vergelijken.

De twee te vergelijken visiesystemen zijn enerzijds de Cognex 7010 en anderzijds de uEye-2230RE-C camera in combinatie met de Halcon-software. De te testen aspecten zijn nauwkeurigheid, verwerkingssnelheid, robuustheid, flexibiliteit en gebruiksgemak. Er is een meetprocedure voor de nauwkeurigheid te kwantificeren. De robot tekent lijnen ter hoogte van het gevonden voorwerp. De spreiding op deze lijnen is een maat voor de nauwkeurigheid. De verwerkingssnelheid is te kwantificeren door de maximale bandsnelheid die het totale systeem bereikt. De overige parameters zijn subjectief beoordeeld.

Het systeem met Cognex-camera behaalt bij lage bandsnelheden (0,1 – 0,5 m/s) en hoge bandsnelheden (0,6 – 1 m/s) een spreiding van respectievelijk 2 mm en 3,5 mm. Voor de opstelling met de IDS-camera is deze spreiding respectievelijk 1 mm en 3 mm. De verwerkingssnelheid bij het Cognex-systeem is te laag bij verdraaide voorwerpen. Het systeem van IDS presteert hier beter. Qua robuustheid valt er weinig te onderscheiden in beide opstellingen. De flexibiliteit ligt beduidend hoger binnen het systeem van IDS. Ten slotte is het Cognex-visiesysteem gebruiksvriendelijker.



# Abstract English

KUKA Automation + Robots JSC at Houthalen wants to obtain knowledge in 2D-vision systems which are suitable for an application involving tracking flat parts on a conveyor with an industrial robot. The aim of this master's thesis is to implement, test and compare these vision systems at different aspects.

In this report, two vision systems are compared. On one hand the Cognex 7010-vision system, on the other hand an IDS uEye-2230RE-C -vision system. The aspects that are investigated are: precision, execution time, robustness, flexibility and ease of use. The aspect precision is quantified by a measurement procedure. The robot draws lines at the height of the found object. The spread of these lines is a measure for the precision. The execution time is quantified by the maximum conveyor speed. The other aspects are subjectively reviewed.

The system with the Cognex-vision system scores at lower conveyor speeds (0,1 m/s – 0,5 m/s) and for higher conveyor speeds a spread of respectively 2 mm and 3,5 mm. The system with the IDS-camera scores a spread of respectively 1 mm and 3 mm. The first system has limited execution times and is therefore not useful for every application. However the second system can reach high execution times. In robustness is less difference noticeable. On one hand, flexibility is higher for the IDS-system. On the other hand it is less comfortable in usage than the Cognex-system.





# Inleiding

## Situering

KUKA Automatisering + Robots N.V. te Houthalen behoort tot de KUKA-groep met het hoofdkantoor gevestigd in Augsburg. Deze firma is één van de voornaamste fabrikanten van industriële robots en complete automatiseringssystemen voor verscheidene industrieën wereldwijd.

## Probleemstelling

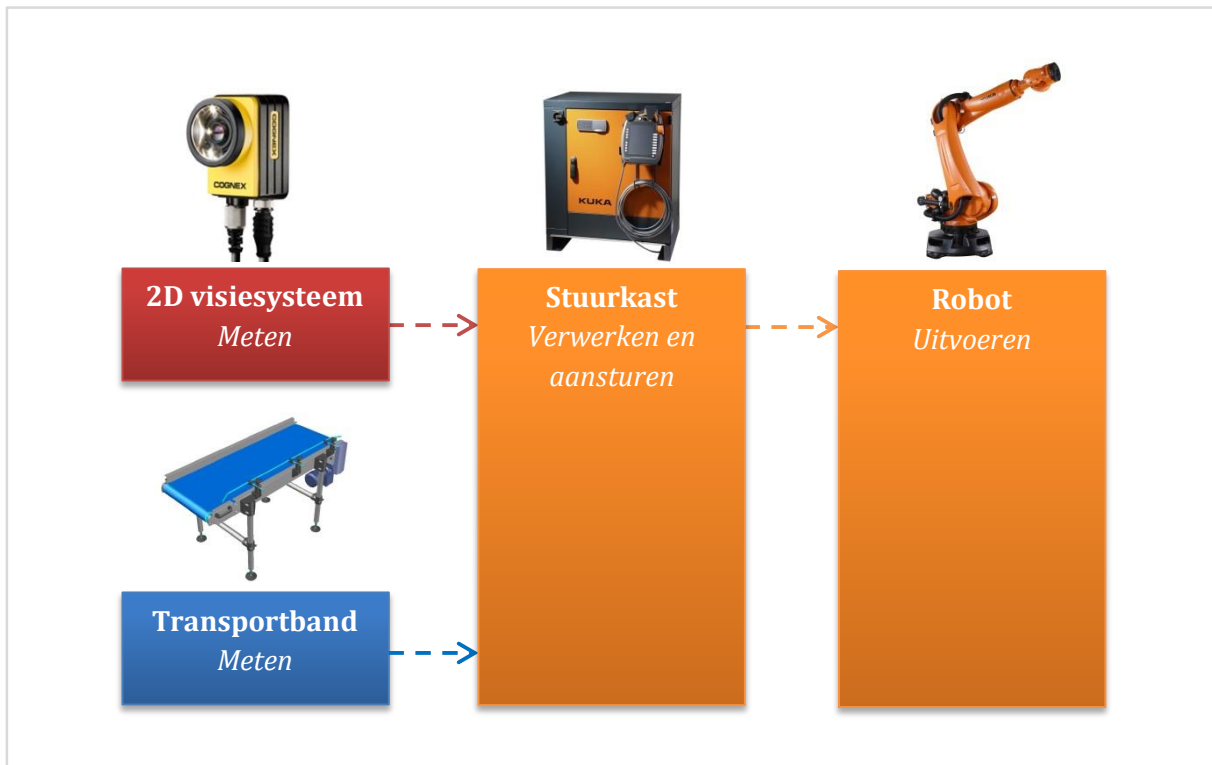
KUKA Houthalen wil doorgroeien op het gebied van visie gecontroleerde systemen. De firma is tot de conclusie gekomen dat het implementeren van robotinstallaties voor het grijpen van delen m.b.v. visiesystemen omslachtig is. Een eenvoudige oplossing dringt zich op. De markt biedt verschillende soorten visiesystemen aan met hun eigen capaciteiten en beperkingen. De vraag bestaat erin welk visiesysteem het meest geschikt is voor het grijpen van delen op een transportband met een industriële robot. Classificatie van de visiesystemen gebeurt aan de hand van enkele parameters.

## Doelstelling

De concrete doelstelling bestaat eruit om drie verschillende 2D-visiesystemen te testen en te vergelijken op verschillende aspecten. Het testen omvat ten eerste het herkennen van delen met een visiesysteem. Ten tweede het volgen van deze delen gedurende 100 mm op een aanvoerband. Ten derde het grijpen ervan met een industriële robot. Een vergelijkingsmatrix plaatst de verkregen testgegevens om zo het juiste visiesysteem voor een bepaalde toepassing beter te kunnen bepalen. Dit aan de hand van enkele cruciale parameters:

- nauwkeurigheid,
- verwerkingssnelheid,
- robuustheid,
- flexibiliteit,
- gebruiksgemak,
- kostprijs.

Een oplossing bevat een 2D-visiesysteem, een transportband met afstandmeting en een robot met stuurkast. Figuur 1 weergeeft dergelijke oplossing.



**Figuur 1:** Schematische voorstelling van de opstelling [1][2][3][4]

De **nauwkeurigheid** is het verschil in afstand tussen de tool van de robot en het voorwerp op de transportband. Deze afstand vergroot bij toenemende bandsnelheden. Enerzijds hangt de nauwkeurigheid af van het visiesysteem en anderzijds van de precisie van het robot-TCP.

De **verwerkingsnelheid** van het systeem hangt af van de uit te voeren bewerking(en). Indien de bewerking vele functies sequentieel moet volbrengen zal het proces logischerwijs een grotere tijd in beslag nemen. De gebruikte componenten in de installatie bepalen de verwerkingsnelheid. Alle gebruikte onderdelen hebben afzonderlijk een beperking in uitvoersnelheid waardoor de maximumsnelheid van het proces is bepaald.

**Robuustheid** of bedrijfszekerheid is binnen de industrie een belangrijk gegeven. Een robuuste installatie zal weinig storingen met zich meebrengen. Factoren die bijdragen tot een grote bedrijfszekerheid zijn onder andere een beperkte complexiteit van de installatie en het gebruik van een real-time aansturingssysteem dat kan inspelen op gewijzigde situaties.

**Flexibiliteit** is de mogelijkheid tot het eenvoudig aanpassen van het systeem. Verdere wensen om de flexibiliteit te vergroten zijn: mogelijkheden om het visiesysteem in te stellen met JPG- of DXF-bestanden en mogelijkheden om het visiesysteem in bedrijf te nemen met extern triggersignaal en/of scanfrequentie.

Het **gebruiksgemak** hangt af van de bediening van de installatie. Dit is een vooropgestelde eis aangezien de opstelling behoort in een productie omgeving waar operators de machine bedienen. Hieronder valt een quasi-automatische instelling van zowel het visiesysteem als robotprogramma. Daarnaast valt ook een eenvoudige definieerbaarheid van de aanvoerband, de te grijpen delen en de hoeveelheid te grijpen delen onder de titel gebruiksgemak.

Ten slotte is de implementatie van deze visietechnologieën in een werkende robottoepassing vereist samen met de rapportering van het eindverslag.

## **Materiaal en methode**

Aangezien de opdracht bestaat uit het vergelijken van twee visiesystemen moet dan ook de juiste software en hardware voor deze systemen aanwezig zijn. Afhankelijk van het te testen visiesysteem verschillen de gebruikte materialen en methoden.

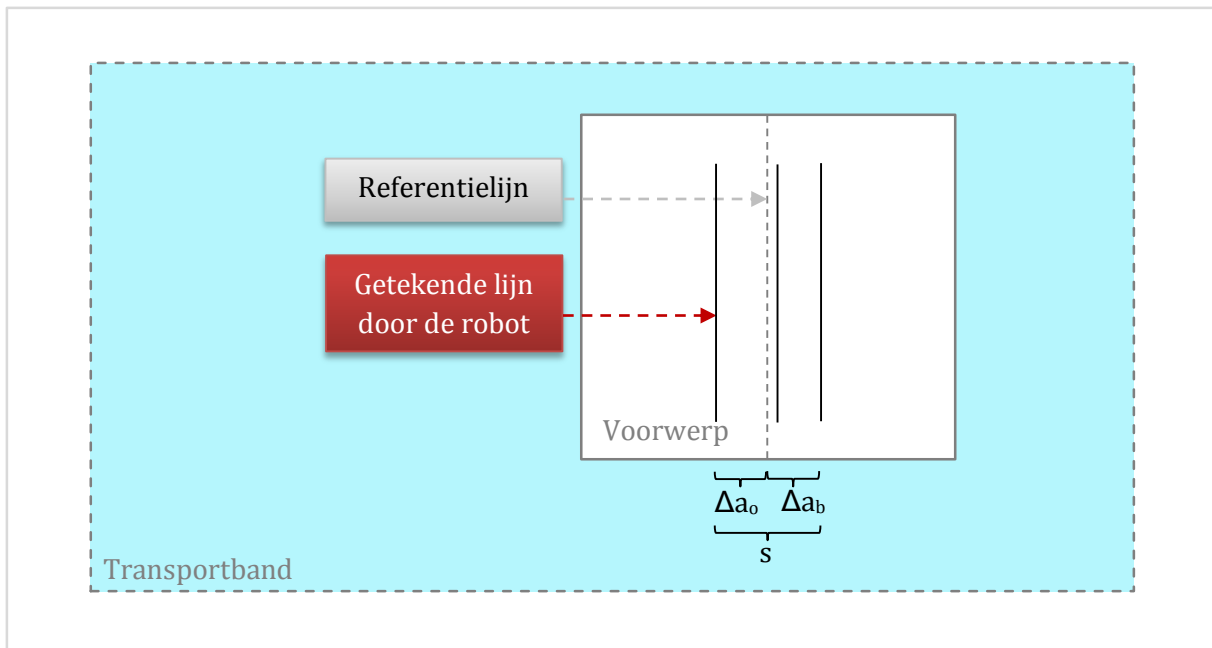
De **Cognex 7010** camera maakt gebruik van een 16 mm f/2,5 lens. De voeding van de camera bestaat uit een 24 VDC-bron. De ondersteunende software voor dit type camera is In-Sight Explorer 4.9.1. De KRC2-sturing van KUKA zorgt voor de aansturing van de robot. Het KUKA technologiepakket EthernetKRLXML, EKX genaamd, aan de robotzijde verzorgt de communicatie tussen de robot en de camera. De software aan de camerazijde biedt de mogelijkheid data te versturen in XML-structuur.

De **IDS uEye UI-2230RE-C-HQ** camera gebruikt het Halcon softwarepakket aangezien dit pakket samen met de licentie en een opleiding beschikbaar is gesteld door het onderzoekscentrum ACRO. Daarnaast is dit een veel gebruikt pakket binnen de visiewereld en biedt het volgens 'experts' veel mogelijkheden. De lens gemonteerd op de camera is verstelbaar. De robotsturing is wederom de KRC2-sturing. De communicatie tussen robot en camera is identiek als in de voorgaande opstelling.

De algemene hardware bestaat voor de twee opstellingen uit de transportband KK XL4 Bandsynchro van het merk Unata samen met een KUKA-resolver. Deze resolver is rechtstreeks aangesloten op de RDC2-kaart van de robot. Op de stuurkast is het technologiepakket ConveyorTech geïnstalleerd om synchronisatie met de transportband mogelijk te maken.

De aansluitschema's in bijlagen 1, 2 en 3 geven een gedetailleerder overzicht van de hardware.

Naast de gebruikte materialen is er een gangbare **meetprocedure**. De totale nauwkeurigheid die de installatie bereikt is van belang. Het visiesysteem maakt een beeld en verwerkt dit. Vervolgens stuurt het systeem de coördinaten van het gevonden object door naar de robot en zal deze robot naar het voorwerp bewegen. Ter hoogte van het te zoeken object bevindt zich een referentielijn. De robot is zodanig ingesteld dat hij ter hoogte van het gevonden object een lijn tekent. Deze getekende lijn ligt niet altijd exact op de referentielijn maar kan een bepaalde afstand voor of achter de referentielijn liggen. De afstand tussen de referentielijn en de getekende lijn is een maat voor de nauwkeurigheid. Indien dezelfde meetprocedure meerdere malen is doorlopen, ontstaat er een spreiding  $s$  rond de referentielijn als maat voor de nauwkeurigheid. Een mogelijk meetresultaat van dergelijk lijnenpaar is weergegeven in figuur 2.



**Figuur 2: Schematische voorstelling meetprocedure**

De verwerkingssnelheid is bepaald uit de maximaal haalbare bandsnelheid van het systeem. Deze bandsnelheid is dan ook de eenheid waarin de verwerkingssnelheid is uitgedrukt. De aspecten flexibiliteit en gebruiksgemak zijn moeilijk objectief te bepalen. Daarom zijn deze parameters bepaald op basis van de eigen ervaringen.

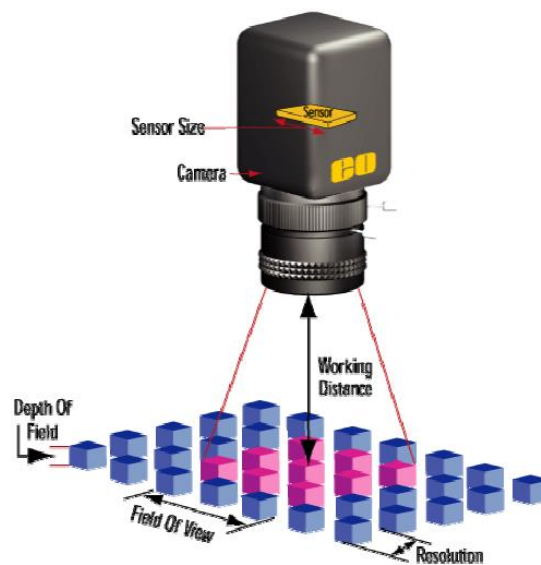
## Opbouw van de scriptie

De scriptie is als volgt opgedeeld. In de literatuurstudie zijn enkele basisbegrippen in verband met visie en beeldverwerking uitgelegd. Hoofdstuk 2 beschrijft de principiële werking van de testopstelling. Hoofdstuk 3 beschrijft de algemene componenten voor beide opstellingen. Hoofdstuk 4 behandelt het Cognex 7010 visiesysteem. Hoofdstuk 5 beschrijft het visiesysteem bestaande uit de IDS uEye-2230RE-C camera samen met de Halcon beeldverwerkingssoftware. In hoofdstuk 6 zijn vervolgens de metingen, de meetresultaten en de vergelijkingsmatrix uitvoerig besproken. Hoofdstuk 7, tevens het laatste hoofdstuk, vat de conclusies samen en blikt vooruit naar de mogelijkheid tot verder onderzoek.

# 1 Literatuurstudie

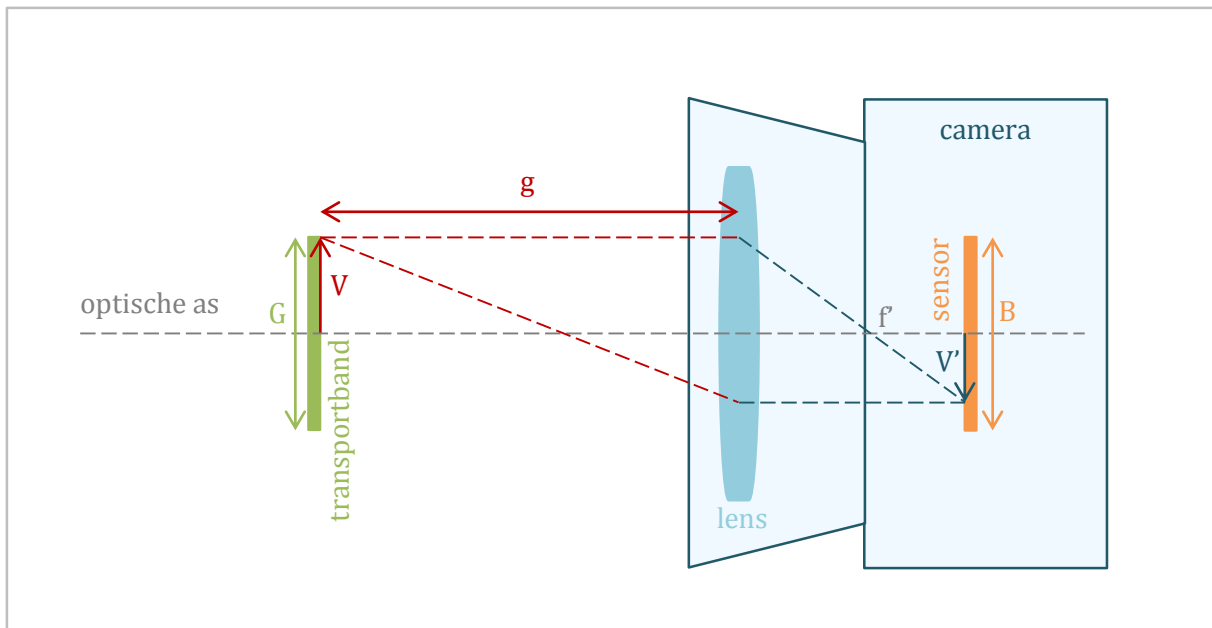
## 1.1 Optische eigenschappen

Op de industriële camera is een lens gemonteerd. De keuze van de lens hangt af van de plaatsing van de camera. De *field of view* en de *working distance* zijn hierin belangrijke parameters. De *field of view* valt op te delen in een hoogte en een breedte. De verhouding tussen deze afstanden noemt de beeldverhouding. Binnen de *machinevision* zijn er enkele gestandaardiseerde verhoudingen waarvan 4:3 de meest gekende is. De *depth of field* dient voor het scherpstellen van het beeld en is bepaald door het diafragma van de lens. Deze drie parameters zijn weergegeven in figuur 3.



**Figuur 3: Optische parameters voor het ophangen van een camerasysteem [5]**

De gepaste lens is bepaald door twee parameters. De eerste is de *field of view*. Deze parameter is afhankelijk naargelang de toepassing. In de thesis is de *field of view* zodanig dat de breedte van de transportband net in een beeld past. Door de beeldverhouding van de camera is de hoogte van het beeld ook vastgelegd. De tweede parameter is de gewenste *working distance*. Eenmaal de twee parameters gekend zijn is het mogelijk om de *focal length* van de lens te bepalen [5]. De optische parameters zijn terug te vinden in het vereenvoudigd lensmodel. Figuur 4 is een schematische voorstelling van dergelijk lensmodel.



**Figuur 4: Vereenvoudigd lensmodel voor een convexe lens**

Uit formule (1) is de brandpuntsafstand af te leiden [6]. De meeste lenzen zijn gestandaardiseerd, het afronden van de uitkomst is daarom vereist. Het is ook mogelijk een lens te nemen met een variabel instelbare brandpuntsafstand.

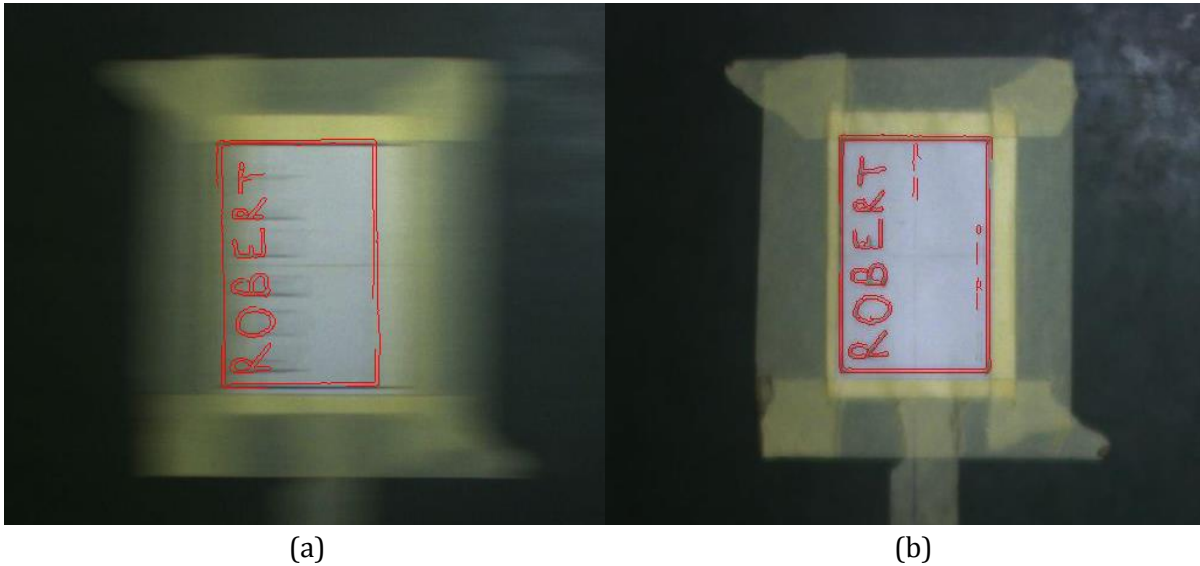
$$f' = \frac{g}{\left(\frac{G}{B} + 1\right)} \quad (1)$$

Hierin is:

- $f'$  de gewenste *focal length*
- $g$  de *working distance*
- $G$  de breedte van de transportband
- $B$  de breedte van de camerasensor

## 1.2 Sluittijd

Om correcte beelden te nemen is de camera-instelling belangrijk. Sluittijd en belichting zijn hierin cruciale elementen. Aangezien de transportband snelheden tot 1 m/s bereikt is het belangrijk een korte sluitertijd in te stellen. Dit blijkt aan de hand van volgend voorbeeld: de sluitertijd bedraagt 50 ms en de transportband beweegt aan 1 m/s. Tijdens het nemen van een beeld is het voorwerp inmiddels 5 cm verschoven. Hierdoor ontstaan ongewenste vegen in het beeld waardoor het voorwerp moeilijk te herkennen is, zie figuur 5 (a). Een korte sluitertijd is dus essentieel voor het duidelijk in beeld brengen van de voorwerpen. Voor beide opstellingen is een sluitertijd van orde grootte 2 ms gebruikt.



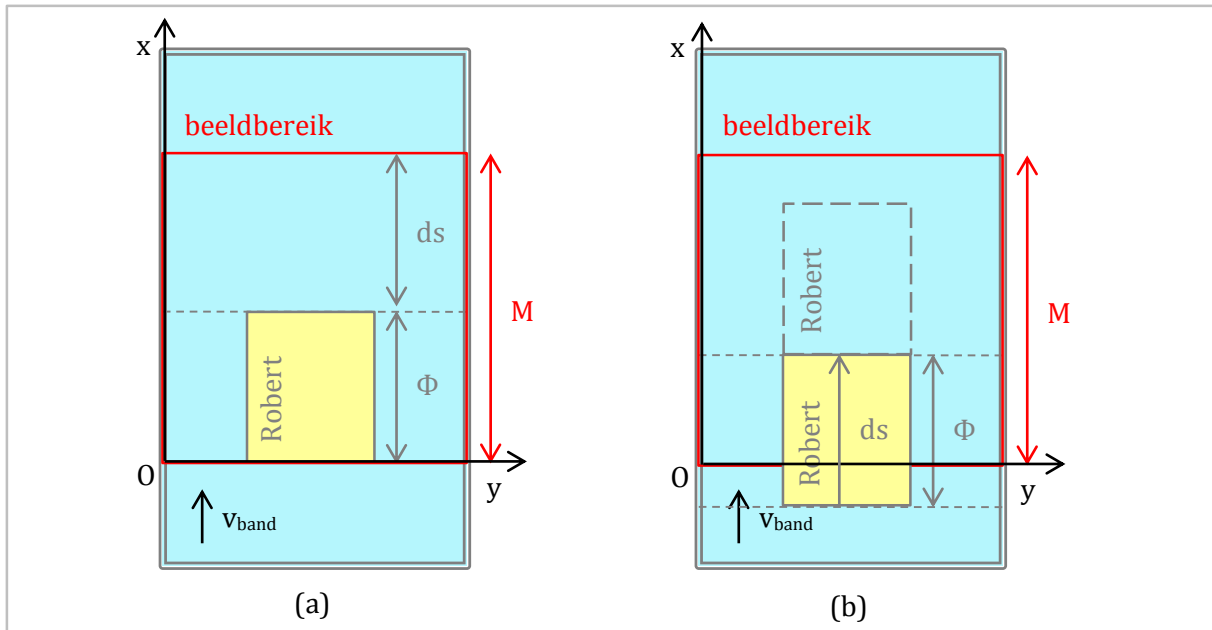
**Figuur 5: Beeld met: (a) een te hoge sluitertijd (b) een lage sluitertijd**

Daarnaast is het belangrijk een juiste belichting in te stellen. Deze belichting is afhankelijk van de sluitertijd. Bij een hoge sluitertijd heeft de optische sensor meer tijd om licht op te vangen. In veel gevallen is extra belichting niet nodig. Is de sluitertijd klein zoals in de beoogde opstelling, dan heeft dezelfde optische sensor minder tijd voor evenveel licht op te vangen. Meer lichtinval is dus vereist. Een externe belichtingsbron kan dienen als extra belichting.

### 1.3 Herkennen van voorwerpen

Voor het herkennen van voorwerpen op een transportband met een visiesysteem ligt er een beperking op de bandsnelheid. Voor het bepalen van deze maximale bandsnelheid gelden enkele randvoorwaarden. Ten eerste neemt de camera met een vaste intervaltijd beelden van de transportband. Binnen deze tijd heeft het visiesysteem dit beeld genomen en verwerkt. Deze methode noemt *continuous framegrabbing* [7]. Ten tweede mag de lengte van het voorwerp logischerwijs nooit groter zijn dan de hoogte van het beeld.

Om elk voorwerp op de transportband te herkennen dient dit voorwerp minimaal een keer volledig in beeld te zijn geweest. De maximale tijd die de camera nodig heeft om steeds een correct beeld van het voorwerp te krijgen is beperkt. Overschrijdt de verwerkingstijd van het visiesysteem deze maximale tijdgrens dan zal de camera het voorwerp niet altijd detecteren en is de bandsnelheid te hoog voor de camera. Deze maximale tijd is afhankelijk van enkele parameters. Dit is verduidelijkt aan de hand van figuur 6.



**Figuur 6: Boven-aanzicht transportband met het voorwerp: (a) binnen het beeldbereik (b) buiten het beeldbereik**

In het geval zoals weergegeven in figuur 6 (a), bevindt het voorwerp zich binnen het beeld van de camera. De camera vindt het voorwerp dus binnen het beeld<sup>1</sup>. Hier is  $\Phi$  de lengte van het voorwerp,  $M$  de hoogte van het beeld en  $ds$  het verschil tussen beide. Het voorwerp ligt echter niet altijd op deze positie binnen het beeld zoals aangegeven op Figuur 6 (b). Het voorwerp is in eerste instantie, tijdens het nemen van het eerste beeld, niet volledig in beeld. De camera herkent het voorwerp in dit geval dus niet. Bij het nemen van het volgende beeld is het voorwerp verplaatst over de afstand  $ds$ . Voor deze afstand  $ds$  zijn er drie situaties mogelijk:

- (1) Indien  $ds < M - \Phi$  is het voorwerp steeds zichtbaar.
- (2) Indien  $M - \Phi \leq ds \leq M + \Phi$  is het voorwerp soms zichtbaar.
- (3) Indien  $ds > M + \Phi$  is het voorwerp nooit zichtbaar.

Enkel situatie (1) voldoet aan de vooropgestelde eis. De verwerkingstijd van de camera mag dus maximaal gelijk zijn aan de tijd die het voorwerp nodig heeft om de afstand  $ds$  af te leggen. Deze tijd valt te bepalen aan de hand van volgende formule:

$$t_{ds} = \frac{ds}{v_{band}} = \frac{M - \Phi}{v_{band}} \quad (2)$$

Uit formule (2) blijkt dat de rekentijd afhangt van drie parameters. Ten eerste hangt de maximale rekentijd af van de lengte van het voorwerp  $\Phi$ . Ten tweede hangt deze tijd af van de breedte van het beeld  $M$ , deze is vast in de opstelling aangezien de camera op een vaste hoogte hangt. Ten slotte hangt deze tijd af van de snelheid van de transportband  $v_{band}$  [8].

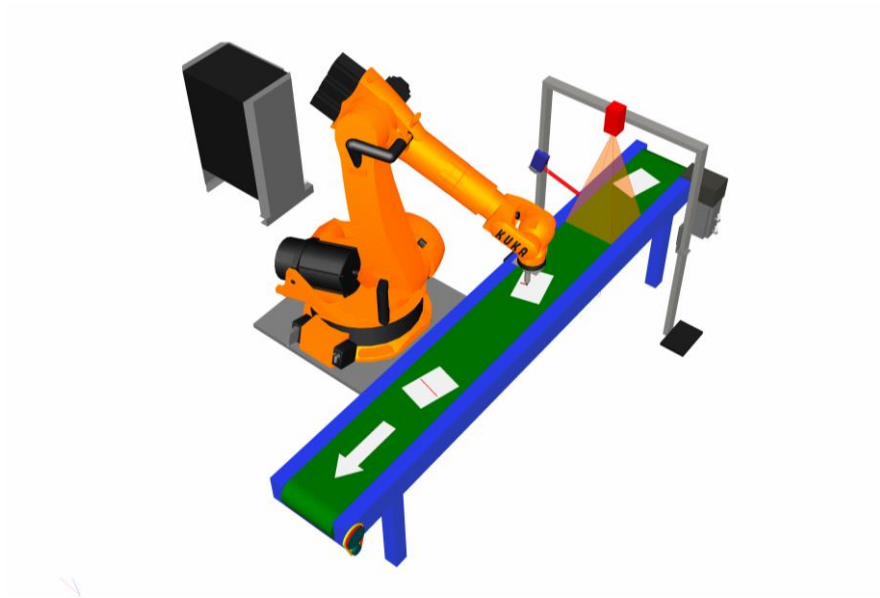
<sup>1</sup> Dit is enkel theoretisch zo, in de praktijk moet het voorwerp een klein stukje verwijderd zijn van de rand van het beeld.



## 2 Algemeen

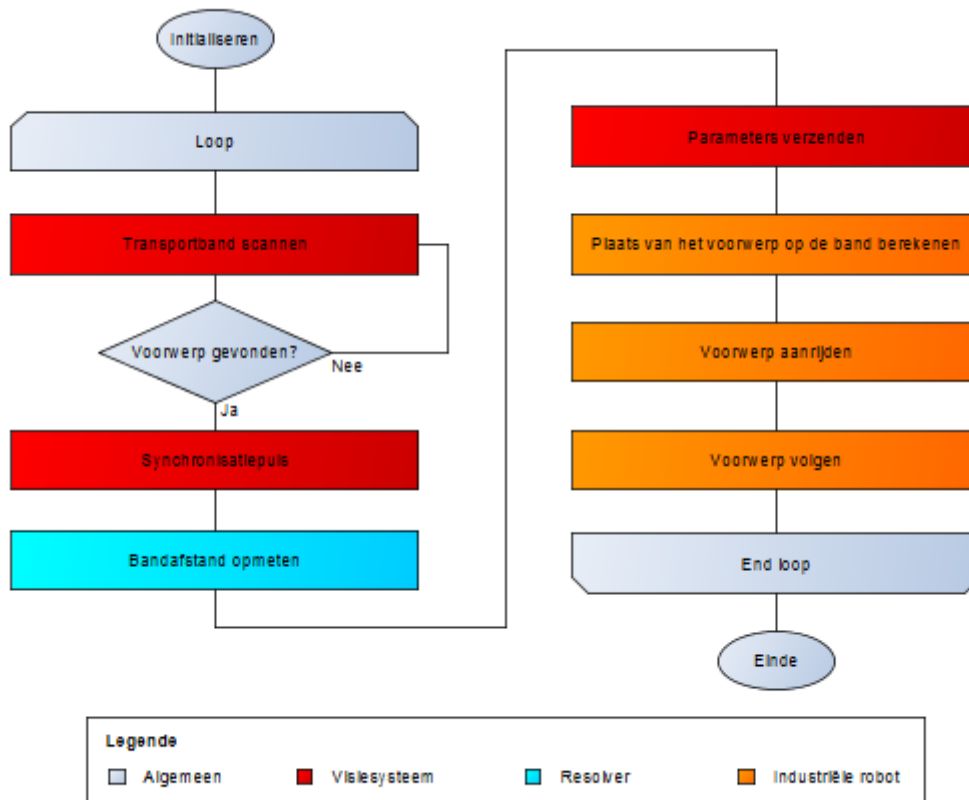
### 2.1 Principiële werking

De beoogde opstelling bestaat uit een 2D-visiesysteem, een transportband met resolver en een industriële robot met sturing. Dergelijke opstelling is weergegeven in figuur 7.



***Figuur 7: Principiële opbouw van de opstelling***

De werking van het systeem valt het best te beschrijven aan de hand van een *flowchart*. Deze *flowchart* beschrijft het sequentieel uit te voeren proces voor alle componenten binnen de opstelling. Deze *flowchart* is afgebeeld in figuur 8. De legende in deze figuur beschrijft welke deelcomponent een bepaalde handeling in het proces uitvoert.

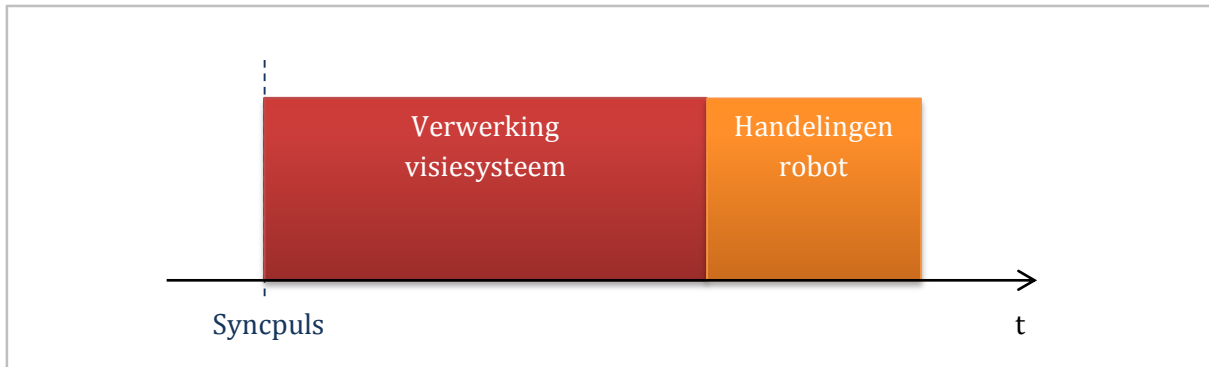


**Figuur 8: Flowchart principiële werking**

De eerste stap in de flowchart is initialisatie. Deze stap bevat het selecteren van de hoofdapplicaties op de robotsturing en het visiesysteem. Vervolgens gaat het programma in een *loop*. De tweede stap bevat het scannen van de transportband m.b.v. het visiesysteem. De voorwaarden voor bandscanning zijn reeds beschreven in paragraaf 1.3. Per genomen beeld gaat het visiesysteem controleren op de aanwezigheid van een voorwerp. Enkel voorwerpen die volledig in beeld zijn kan het visiesysteem herkennen en vergelijken<sup>2</sup>. Indien het visiesysteem het aangeleerde voorwerp niet vindt, herhaalt stap twee zich. Als het gewenste voorwerp daarentegen is gevonden gaat het systeem naar stap drie. Op de achtergrond meet een resolver, gemonteerd op de transportband, de bandafstand continu op. Vervolgens *set* het visiesysteem een synchronisatiepuls. De robot ontvangt dit signaal via zijn daarvoor voorziene aansluiting. Vanaf dit moment is de robot gesynchroniseerd met de transportband. De robot *reset* de verplaatste bandafstand naar nul. Vervolgens, in stap vier, gaat het visiesysteem zowel de coördinaten als de hoekverdraaiing van het voorwerp berekenen en versturen. Vanaf het versturen van de puls tot het versturen van de informatie over het voorwerp heeft het visiesysteem bepaalde handelingen uitgevoerd die rekentijd vereisen. Na het verstrijken van deze rekentijd is de band al een bepaalde afstand verplaatst afhankelijk van de bandsnelheid waardoor de coördinaten van een moment eerder niet meer overeenkomen met de actuele positie van het voorwerp. Om deze fout weg te regelen meet de visieapplicatie de verwerkingstijd op en verstuurt deze parameter samen met de

<sup>2</sup> Deze voorwaarde is zo ingesteld om problemen i.v.m. de coördinaten te voorkomen. Indien het middelpunt van het voorwerp niet in beeld is, zijn de coördinaten van het voorwerp niet te berekenen door het visiesysteem.

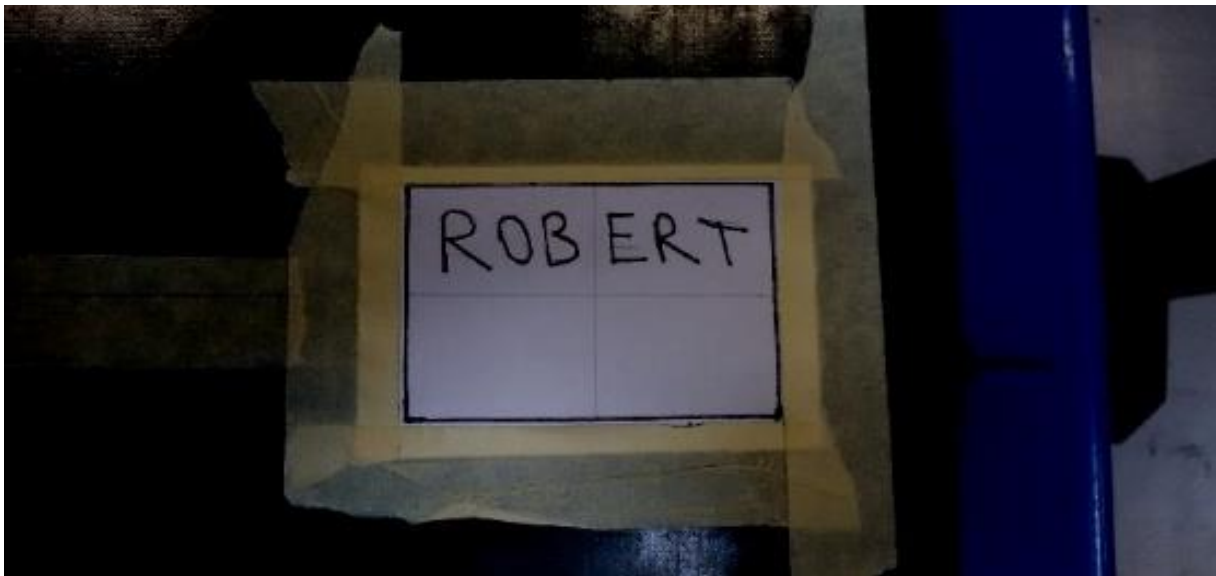
eigenschappen van het gevonden voorwerp naar de robotsturing. Een schematische voorstelling van deze verwerkingstijd is weergegeven in figuur 9.



**Figuur 9: Schematische voorstelling van de verwerkingstijd**

In de vijfde stap gaat de robot de positie van het voorwerp op de transportband berekenen. Hier houdt de robotapplicatie rekening met de verplaatste bandafstand t.g.v. de verwerkingstijd van het visiesysteem. Deze verplaatste bandafstand is gecompenseerd door deze tijd te vermenigvuldigen met de bandsnelheid. In de laatste stap gaat de robot de berekende punten aanrijden en het voorwerp gedurende een bepaalde afstand volgen.

De te herkennen voorwerpen zijn naamkaartjes. Het gebruik van naamkaartjes biedt enkele voordelen. Enerzijds zijn deze voorwerpen vast te maken aan de transportband, dit is praktisch voor de metingen. Anderzijds bevinden de voorwerpen zich op dezelfde hoogte als de transportband, hierdoor is de *depth of field* gelijk aan nul. Op deze manier is er geen schaduwvorming. Figuur 10 geeft een voorbeeld van dergelijk naamkaartje.



**Figuur 10: Voorbeeld van een naamkaartje op de transportband**



## 3 Algemene componenten

Dit hoofdstuk behandelt alle componenten in het systeem die voor beide opstellingen gelijk zijn. Deze cruciale deelcomponenten zijn de ethernetcommunicatie, de bandsynchronisatie en het robotprogramma.

### 3.1 Ethernetcommunicatie

De communicatie heeft als doel de door de camera opgemeten informatie te verzenden naar de robotsturing. Deze verbinding bestaat uit een ethernetnetwerk. De communicatie is te verdelen in twee onderdelen. Enerzijds de communicatie aan de visiesysteemzijde, anderzijds aan de robotzijde.

#### 3.1.1 Visiesysteemzijde

De communicatie aan de zijde van het visiesysteem heeft als doel de opgemeten data in XML-structuur op te bouwen en te versturen. De structuuropbouw is hetzelfde voor beide visiesystemen [9]. Het gehanteerde protocol is TCP/IP. Dit protocol werkt niet volgens het *real-time* principe. Het visiesysteem bezit hierdoor een variabele zendtijd. De datapakketten binnen dit protocol hebben een lage prioriteit. Deze pakketten moeten m.a.w. voorrang verlenen aan datapakketten met hoge prioriteit. Hierdoor kan een variabele tijdsvertraging ontstaan op de verzendtijd. Dit risico is te beperken door te werken met zo weinig mogelijk dataverkeer binnen het netwerk. In de meest gunstige opstelling is het visiesysteem rechtstreeks aangesloten op de robotsturing. In de beoogde opstelling is *real-time* datacommunicatie echter niet vereist. Enkel de synchronisatiepuls moet *real-time* zijn, dit is verduidelijkt in paragraaf 3.2.

Binnen het netwerk is het visiesysteem een TCP/IP-*client*. Elke *client* heeft een uniek IP-adres en een poortnummer. Voor het visiesysteem zijn deze parameters als volgt ingesteld.

- IP-adres: 10.32.3.19
- Poortnummer: 3000

Het visiesysteem verstuurt de data enkel indien het een voorwerp vindt. Dergelijke data is gestructureerd in XML-code. Figuur 11 geeft de opbouw voor dergelijk XML-structuur. Indien het visiesysteem niets vindt, verstuurt het de string “#ERR” die de robot niet kan lezen.

```
<Data>
  <X>value</X>
  <Y>value</Y>
  <A>value</A>
  <T>value</T>
  <K>value</K>
</Data>
```

**Figuur 11: XML-structuur voor verzenden**

Het visiesysteem zendt vijf verschillende parameters met behulp van zes XML-tags. De betekenis van deze tags staan weergegeven in tabel 1.

**Tabel 1: Verklaring XML-tags uit de te verzenden data**

Naam	Type	Betekenis
Data	Structtag	Structurele tag voor de XML-structuur.
X	Real/Floating point	De x-coördinaat van het middelpunt in fixtures.
Y	Real/Floating point	De y-coördinaat van het middelpunt in fixtures.
A	Real/Floating point	De hoek van het voorwerp rond de z-as in graden.
T	Real/Floating point	De tijd van de beeldverwerking in milliseconden.
K	Integer	Het id-nummer van het gevonden voorwerp.

De waarde K dient om verschillende voorwerpen te onderscheiden. Alle mogelijke waarden voor K behoren tot de verzameling  $Z^+$ . Enkel in het geval waarbij K gelijk is aan nul, is er geen voorwerp gevonden. Alle andere discrete positieve waarden voor K geven aan dat er een voorwerp is gevonden. De K-waarde is dus de *identifier* of id van het voorwerp. Dit id-nummer is gekoppeld aan het overeenkomstig model binnen de software van het visiesysteem. Op deze manier is de instelling van het visiesysteem toegankelijk voor het zoeken van één of meerdere voorwerpen zonder deze instelling te moeten wijzigen.

Een belangrijke kanttekening binnen voorgaande methode is dat er nooit twee voorwerpen binnen één beeld kunnen liggen. Indien dit toch het geval is zal de robot geen enkel voorwerp van de band nemen. Is dit echter gewenst dan kan een alternatieve methode hiervoor een oplossing bieden. De XML-structuur uit figuur 11 is dan aangepast en krijgt de vorm zoals in figuur 12. Het voorbeeld binnen deze figuur is enkel geldig voor het nemen van twee voorwerpen.

```

<Data>
  <X1>value</X1>
  <Y1>value</Y1>
  <A1>value</A1>
  <X2>value</X2>
  <Y2>value</Y2>
  <A2>value</A2>
  <T>value</T>
</Data>

```

**Figuur 12: XML-structuur voor verzenden van de alternatieve configuratie**

De X- en Y-coördinaat en de rotatie A van elk voorwerp krijgen hier een aparte XML-tag. De tijd omvat de totale rekentijd van de camera voor beide voorwerpen. In dit voorbeeld is de XML-structuur enkel gangbaar voor één of twee voorwerpen. Voor een uitbreiding naar meerdere voorwerpen is een wijziging van zowel deze XML-structuur alsook het robotprogramma vereist. Vandaar dat deze methode niet is gehanteerd binnen het systeem.

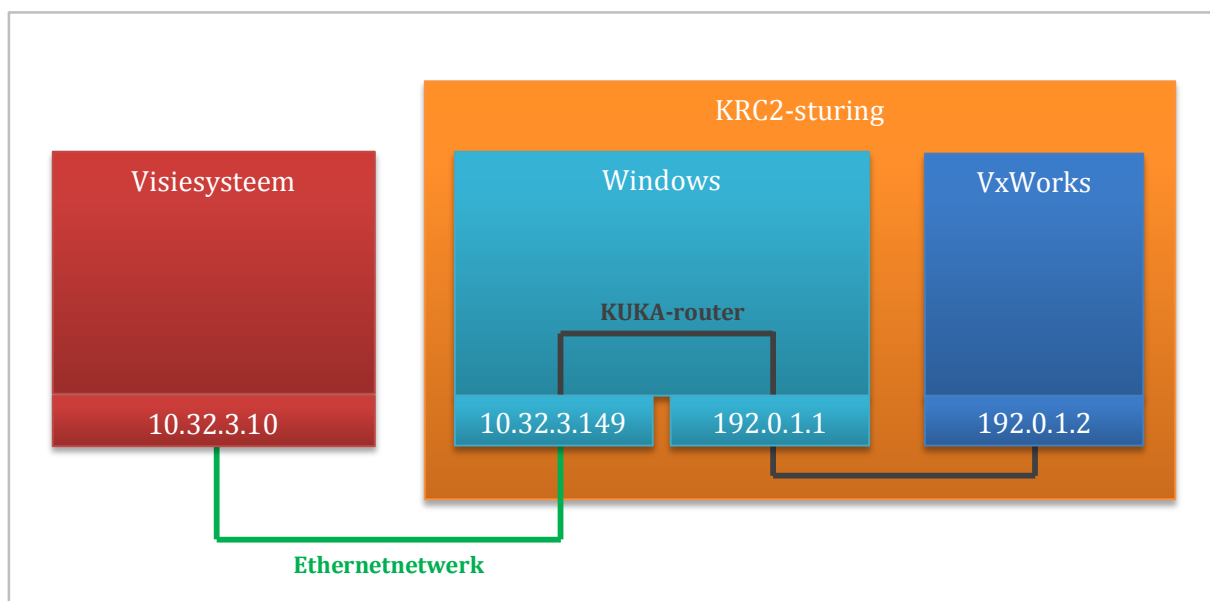
### 3.1.2 Robotzijde

De robotsturing is de ontvanger in het netwerk en heeft net zoals het visiesysteem een IP-adres. De sturing fungeert als server in het netwerk en heeft om deze reden geen poortnummer.

- IP-adres: 10.32.3.149

Om de te ontvangen data te kunnen inlezen gebruikt de robot het EthernetKRLXML-technologiepakket. Dit pakket maakt het mogelijk de data in te lezen in VxWorks, het *real-time* besturingssysteem van de robot. Het IP-adres van VxWorks verschilt met het IP-adres van het netwerk waardoor rechtstreekse communicatie onmogelijk is. Het technologiepakket voorziet hiervoor twee oplossingen.

De eerste optie is het gebruik van een tweede netwerkkaart waardoor het rechtstreeks communiceren met VxWorks wel mogelijk is. Deze configuratie is niet beschikbaar voor deze thesis. De tweede optie is KUKA-router, een virtuele router binnen het technologiepakket. De instelling van deze router is terug te vinden in bijlage 4. De verstuurde data maakt dan eerst een omweg via het Windows besturingssysteem dat parallel draait langs VxWorks. Deze router verandert het IP-adres binnen het ethernetnetwerk naar een compatibel IP-adres voor de verbinding met VxWorks, weergegeven in figuur 13. Deze configuratie is gehanteerd in deze thesis.



**Figuur 13:** Schematische voorstelling KUKA-router

De robot moet nu enkel nog verbinding maken met de gewenste sensor, in dit geval het visiesysteem. Dit kan door de XmlApiConfig in te stellen. De inhoud van XmlApiConfig.xml is weergegeven in figuur 14 en terug te vinden in bijlage 5. Tabel 2 geeft de belangrijkste parameters binnen deze *file* samen met hun betekenis.

```
<XmlApiConfig>
  <XmlApiParam InitOnce="false"/>
  <Channel SensorName="Cognex7010" SensorType="2D">
```

```

    <TCP_IP IP="10.32.3.19" Port="3000" Route="true" MapPort="3000" />
  </Channel>
</XmlApiConfig>

```

**Figuur 14: XML-structuur van de XmlApiConfig.xml**

**Tabel 2: Verklaring belangrijkste XML-tags en hun parameters uit de XmlApiConfig.xml**

Naam	Betekenis
XmlApiConfig	Structurele tag voor de XML-structuur.
Channel	Structurele tag voor de XML-structuur.
Channel.SensorName	De naam van het bestand met voorgedefinieerde XML-structuur.
TCP_IP	Structurele tag voor de XML-structuur.
TCP_IP.IP	Het IP-adres van de sensor.
TCP_IP.Port	De netwerkpoort van de sensor.
TCP_IP.Route	Gebruik van de KUKA-router.

Om de communicatie aan de robotzijde te vervolledigen moet de robotsturing de XML-structuuropbouw voor zenden en ontvangen kennen alvorens er gecommuniceerd is. De opbouw van deze structuur is gedefinieerd in twee *files*: Cognex7010.xml voor ontvangen en Cognex7010+.xml voor verzenden. Beide files zitten respectievelijk in bijlage 6 en 7. Deze laatste *file* is echter leeg aangezien de robot enkel data ontvangt. Het Cognex7010.xml bestand is afgebeeld in figuur 15.

```

<Elements>
  <Element Tag="Data" Type="STRUCTTAG" Stacksize="5" />
  <Element Tag="Data.X" Type="REAL" Stacksize="5" />
  <Element Tag="Data.Y" Type="REAL" Stacksize="5" />
  <Element Tag="Data.A" Type="REAL" Stacksize="5" />
  <Element Tag="Data.T" Type="REAL" Stacksize="5" />
  <Element Tag="Data.K" Type="INTEGER" Stacksize="5" />
</Elements>

```

**Figuur 15: XML-structuur van de Cognex7010.xml**

Voor de alternatieve methode met de twee voorwerpen is de Cognex7010.xml *file* anders geconfigureerd. Aangezien het visiesysteem de coördinaten en rotatie voor elk voorwerp apart verstuurt, dienen deze parameters afzonderlijk gedeclareerd te zijn. Dergelijk bestand is dan opgebouwd volgens figuur 16. Ook in dit geval is de *file* opgebouwd voor slechts twee voorwerpen [10].

```

<Elements>
  <Element Tag="Data" Type="STRUCTTAG" Stacksize="5" />
  <Element Tag="Data.X1" Type="REAL" Stacksize="5" />
  <Element Tag="Data.Y1" Type="REAL" Stacksize="5" />
  <Element Tag="Data.A1" Type="REAL" Stacksize="5" />

```



```
<Element Tag="Data.X2" Type="REAL" Stacksize="5" />
<Element Tag="Data.Y2" Type="REAL" Stacksize="5" />
<Element Tag="Data.A2" Type="REAL" Stacksize="5" />
<Element Tag="Data.T" Type="REAL" Stacksize="5" />
</Elements>
```

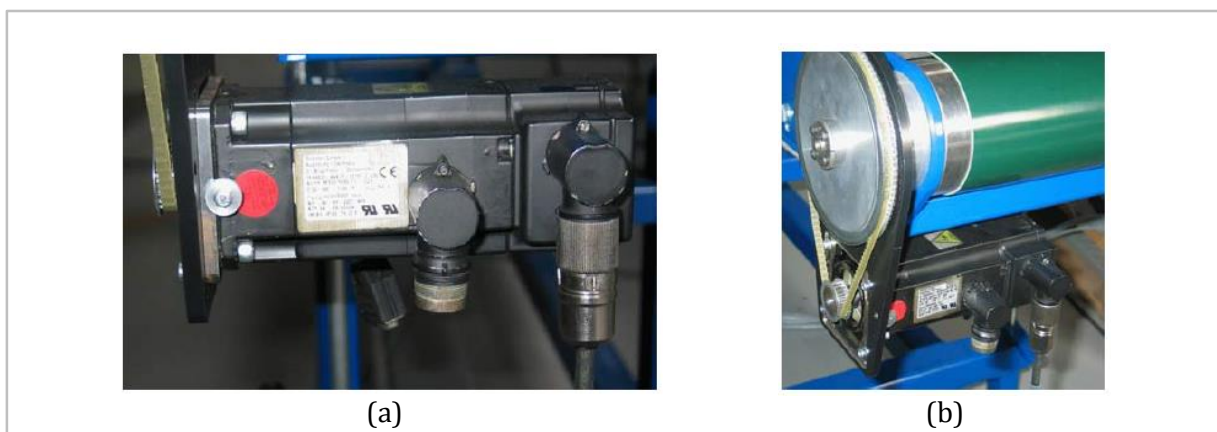
**Figuur 16:** XML-structuur van de *Cognex7010.xml* voor de *alternatieve methode*

## 3.2 Bandsynchronisatie

Door het technologiepakket ConveyorTech is het mogelijk om de robot te synchroniseren met een lopende transportband. Deze synchronisatie uit zich in het synchroon lopen van het tool center point (TCP) met de band. Het technologiepakket verwezenlijkt dit door een verplaatsende *base* te berekenen die synchroon loopt met de transportband.

Synchronisatie is verwezenlijkt door twee elementen: enerzijds een synchronisatiepuls of startpuls en anderzijds een resolver voor het opmeten van de afgelegde bandafstand. Deze resolver staat afgebeeld in figuur 17. De RDC2-kaart in de robot zet de sinusodiële resolversignalen om naar pulsen. De verplaatste bandafstand per puls is ongekend. Daarom is de configuratie van ConveyorTech noodzakelijk.

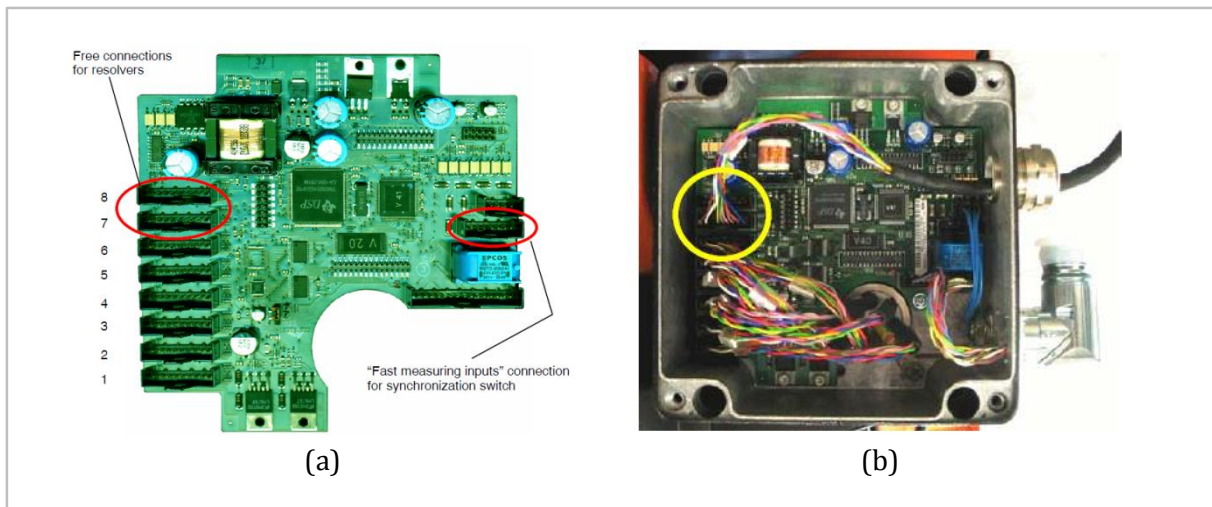
Tijdens het configureren telt de robot het aantal pulsen vanaf het synchronisatiesignaal. De robot rijdt het vertrekpunt, waar de synchronisatiepuls is gegeven, aan nadat dit over een bepaalde afstand is verplaatst. Met deze gegevens schat de robot de verplaatste afstand over de transportband. Na controle van deze afstand is ConveyorTech aangeleerd. De robotsturing berekent na een synchronisatiepuls de verplaatste bandafstand. Daarnaast is het mogelijk de transportband te volgen met de robot voor elke bandsnelheid.



**Figuur 17:** Resolver voor ConveyorTech (a) zijaanzicht [11] (b) gemonteerd op de transportband [11]

Een RDC2-kaart leest de resolversignalen en de synchronisatiepuls in. Dergelijke kaart is afgebeeld in figuur 18 (a). De synchronisatiepuls gaat van het visiesysteem via de X33-aansluiting naar de robot. In de robot is deze X33-aansluiting rechtstreeks verbonden met de snelle meetingang van de RDC2-kaart. De acht connecties op de kaart dienen om de resolver op de transportband in te lezen. De sloten één tot zes gebruikt de robot voor zijn eigen assen, daarom

moet de resolver op slot zeven of acht aangesloten zijn. De robot kan deze sloten ook gebruiken om externe assen op in te lezen [11].

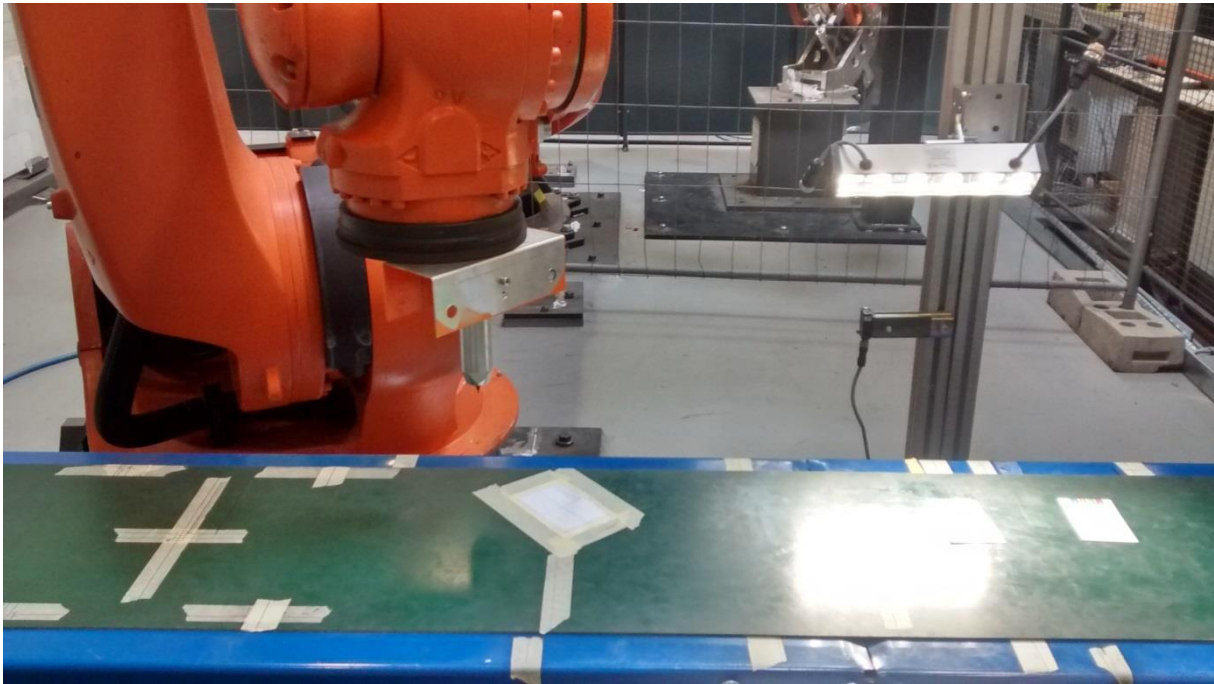


**Figuur 18: RDC2-kaart met aansluitingen (a) algemeen [11] (b) aangesloten [11]**

### 3.3 Industriële robot

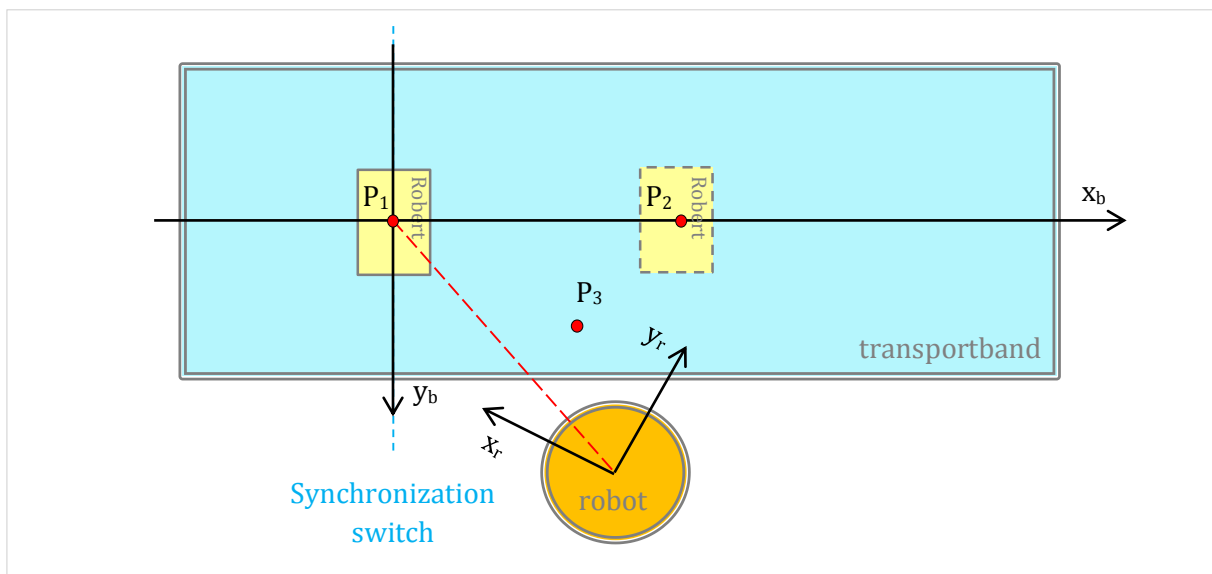
#### 3.3.1 Inbedrijfname

Een industriële robot is enkel bruikbaar nadat hij is geconfigureerd. De eerste stap is het aanleren van de eindeffector. De tweede stap is het opmeten van het *base*-coördinatenstelsel. De gebruikte *tool* in de thesis is een *pointer* die bestaat uit een stift. De *pointer* is rechtstreeks bevestigd op de flens van de robot. Dit is weergegeven in figuur 19. Het *tool center point* (TCP) van de *tool* is aangeleerd aan de robot volgens de XYZ-4-punt-methode. Hierbij rijdt de robot een referentiepunt in de ruimte aan uit vier verschillende richtingen waardoor de robot het TCP kan bepalen. Uit de eerste drie meetpunten berekent de robotsturing het middelpunt van een bol. Dit middelpunt is gelijk aan het referentiepunt en dus het uiteinde van de *tool*. Het vierde punt dient om de meetfout te berekenen. Deze fout is een maat voor de nauwkeurigheid van de meting. De nauwkeurigheid van het aanrijden van alle vier de punten bepaalt dus de uiteindelijke nauwkeurigheid van het TCP [12]. Als vuistregel voor een aanneembare nauwkeurigheid is een resultaat onder 0,5 mm gangbaar.



**Figuur 19: Verende pen als TCP**

De tweede stap is het opmeten van de *base*. Er zijn twee verschillende *bases* opgemeten. De eerste is een vaste base, de tweede een base die zich met de transportband voortbeweegt. De eerste base is opgemeten met de 2-puntsmethode. Hierbij is het XY-vlak bepaald m.b.v. de oorsprong en een tweede punt. De laatste stap bestaat erin de richting van de z-as op te meten [12]. Het opmeten van het assenstelsel van de transportband gebeurt m.b.v. ConveyorTech. Het voorwerp is hierbij in het midden van het beeld geplaatst. De camera stuurt bijgevolg de synchronisatiepuls uit. Hierdoor ligt het eerste punt  $P_1$  in het midden van het camerabeeld. Vervolgens start de band waardoor het voorwerp verplaatst naar  $P_2$ . ConveyorTech slaat de afstand  $P_{1,x}$  tot  $P_{2,x}$  op. Zo kan de robotsturing de resolversignalen rechtstreeks omzetten naar een afstand in mm. Vervolgens rijdt de robot de punten  $P_2$  en  $P_3$  aan om het XY-vlak op te meten. Figuur 20 geeft een schematische voorstelling voor het opmeten van de bewegende *base* met het ConveyorTech technologiepakket.



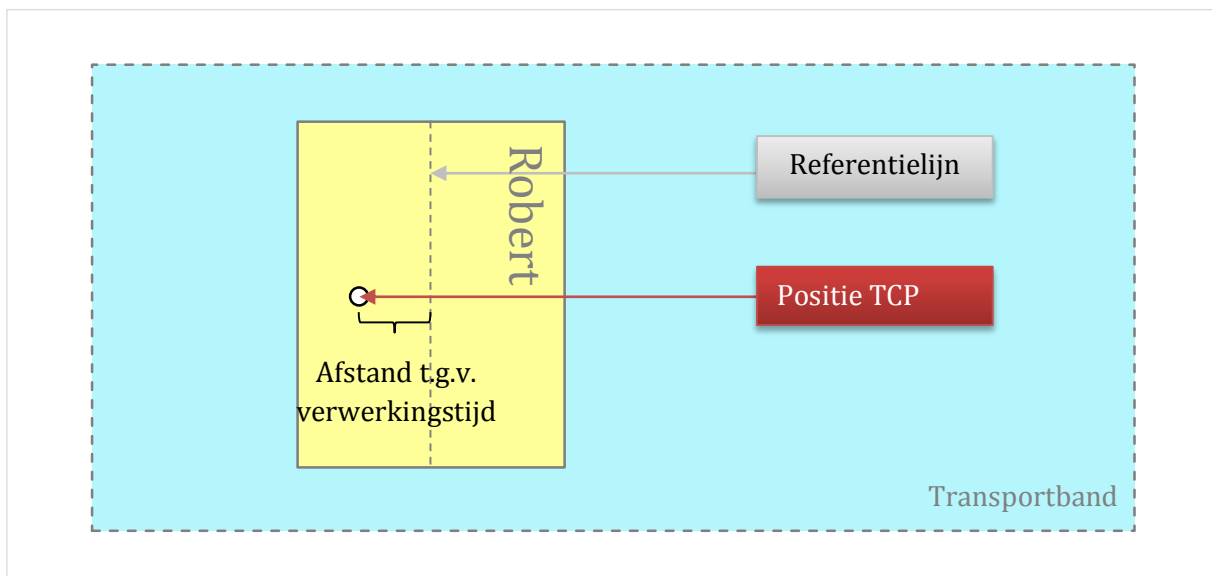
**Figuur 20: Opmeten van de conveyorbase**

ConveyorTech ondersteunt naast het opmeten van de *conveyorbase* ook een meting voor het voorwerp. Het voorwerp op de transportband krijgt dan zelf een assenstelsel. De gehanteerde voorwerpen binnen deze thesis liggen echter willekeurig verdraait op de transportband. Hierdoor zou de *base* van het voorwerp steeds veranderen. In de huidige oplossing verandert de gebruikte *base* niet. De verdraaiing van het voorwerp is gecompenseerd door de verdraaiing rond de z-as, de coördinaat A, aan te passen [11].

### 3.3.2 Snelheidsmeting

Er zijn twee robotprogramma's nodig om het systeem te laten werken. Het eerste programma meet de snelheid van de band en slaat deze snelheid op. Indien de bandsnelheid wijzigt is het noodzakelijk dit programma opnieuw te doorlopen. Het tweede programma is het hoofdprogramma van de robot en dient om de voorwerpen te volgen op de band.

De snelheid van de transportband is vereist om de benodigde verwerkingstijd van het visiesysteem te compenseren. De eerste robotapplicatie meet deze snelheid op. Het visiesysteem heeft een zekere tijd nodig om een beeld te nemen, dit beeld te verwerken en vervolgens de resultaten door te geven aan de robot. Terwijl het visiesysteem het beeld verwerkt verschuift het oppervlak waar een beeld van genomen is aangezien de transportband blijft draaien. De robot krijgt de resultaten binnen van het verwerkte beeld, maar dit beeld is genomen van een beeldoppervlak dat inmiddels is verschoven. De doorgegeven coördinaten komen niet meer overeen met de plaats van het voorwerp. De coördinaten lopen achter met een afstand gelijk aan de bandsnelheid vermenigvuldigd met de tijd die de camera nodig heeft vanaf dat het beeld genomen is totdat de resultaten berekend zijn en de synchronisatiepuls gegeven is. Indien de snelheid is opgemeten, is het dus mogelijk om deze afstand weg te werken. Figuur 21 geeft een schematische voorstelling van de te compenseren afstand.



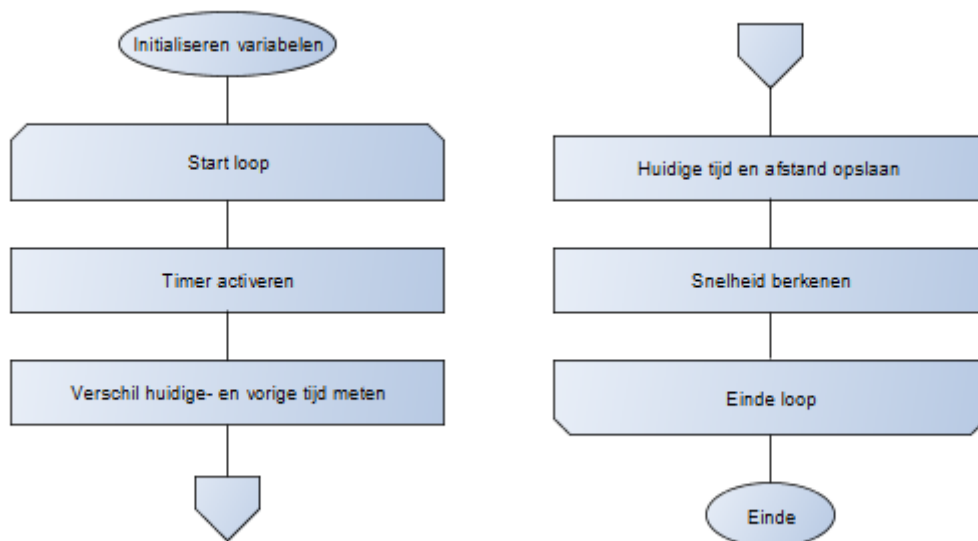
**Figuur 21: Achterlopen robot t.g.v. verwerkingstijd**

De robot meet de bandsnelheid continu op. Dit is mogelijk door de snelheidsmeting te programmeren in de *submit interpreter* (*sps.sub-file*) van de robot. Het *sps*-programma toont enkele gelijkenissen met de principiële werking van een PLC. Het programma draait continu op de achtergrond met een cyclustijd van  $\pm 12$  ms. Deze korte cyclustijd is vereist om een *real-time* werking te garanderen. Figuur 22 geeft de flowchart van het *sps*-programma weer. De eerste stap

dient om de variabelen te initialiseren. De robot doorloopt de initialisatiestap enkel bij het opstarten. Vervolgens stapt het programma in een lus die het zal blijven doorlopen zolang de *submit interpreter* actief is. Binnen deze lus is de volgende en tevens tweede stap een timer starten. In de derde stap berekent het programma het verschil tussen de huidige timerwaarde en een opgeslagen timerwaarde. Dergelijke berekening is analoog voor de afstand. De mogelijkheid om de bandafstand op te vragen is er enkel indien het technologiepakket ConveyorTech is geïnstalleerd. Nadat het verschil in tijd en afstand is berekend, slaat het programma deze waarden op in het geheugen. In het programma is er echter een voorwaarde opgelegd bij het opslaan van de huidige tijd en afstand. Deze voorwaarde zegt dat het opslaan van de waarden enkel is toegelaten indien het tijdsverschil groter is dan 100 ms. Dit moet aangezien de *submit interpreter* anders zodanig snel is doorlopen, dat het tijdsverschil te kort is om een representatieve snelheidsmeting te verkrijgen. De laatste stap binnen de applicatie berekent de snelheid uit het verschil in tijd en afstand met formule (3).

$$R_{Snelheid} = \frac{\Delta S}{\Delta t} \left[ \frac{m}{s} \right] \quad (3)$$

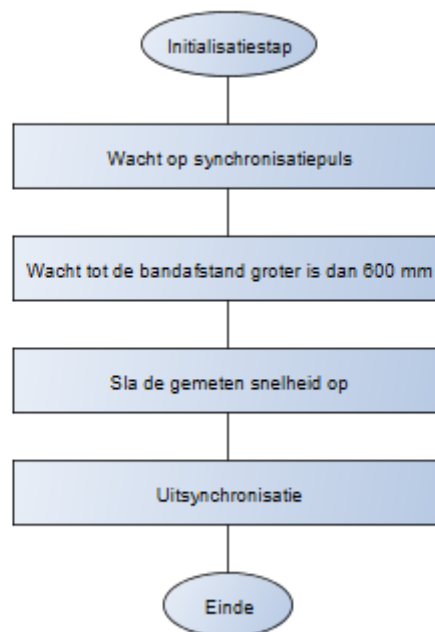
De waarde voor de snelheid is weggeschreven in een globale variabele zodat elk programma deze waarde kan opvragen.



**Figuur 22: flowchart snelheidsmeting sps.sub**

Aangezien ConveyorTech de afgelegde bandafstand enkel kan berekenen nadat de band met de robot is gesynchroniseerd, is er een extra programma geschreven om de robot te synchroniseren en de bandsnelheid op te slaan. Figuur 23 geeft de flowchart van het extra programma weer. Binnen de initialisatiestap desynchroniseert de robot met de transportband. Deze stap is vereist indien er een applicatie vroegtijdig onderbroken werd waardoor de robot nog steeds was gesynchroniseerd. Vervolgens geeft de applicatie aan ConveyorTech het commando om te wachten op een puls. Bij de derde stap wacht ConveyorTech tot de bandafstand meer dan 600 mm bedraagt. Deze stap dient om er zeker van te zijn dat de band op de ingestelde snelheid draait en niet meer aan het versnellen is na opstart. Indien de band meer als 600 mm heeft afgelegd moet het programma de snelheid opslaan in een globale variabele. Het hoofdprogramma,

waarin de robot de naamkaartjes volgt, gebruikt deze variabele om de verwerkingstijd van de camera weg te werken. De snelheid is eenmalig opgeslagen tijdens de snelheidsmeting. Deze variabele bevat dus steeds een constante waarde voor de snelheid. Bij een wijziging van de bandsnelheid is het dus vereist de snelheidsmeting opnieuw te doorlopen. De laatste stap dient om de band te desynchroniseren. Zowel het sps-programma als het programma voor het opslaan van de snelheid zijn terug te vinden in de bijlage 8 en 9.

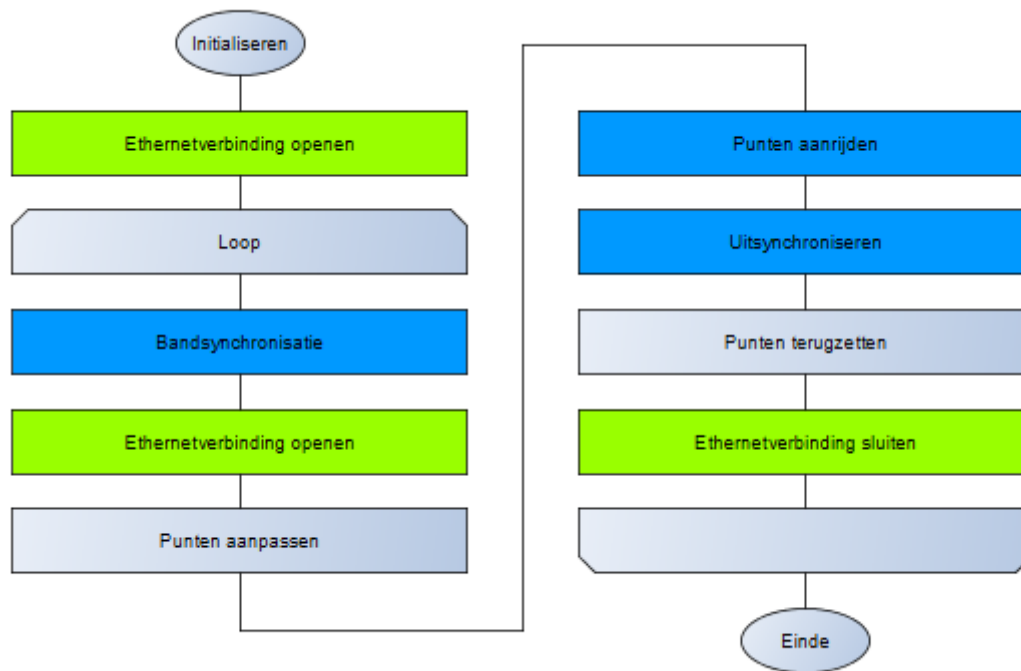


**Figuur 23: flowchart opmeten snelheidsvariabele voor hoofdprogramma**

### 3.3.3 Hoofdprogramma

Het robotprogramma is geschreven in KRL-code ofwel *KUKA Robot Language*. Het programma is van het type module en bestaat typisch uit twee *files*: een *source-* en *data-file*. Het volledige robotprogramma is te vinden in de bijlagen 10, 11, 12 en 13.

De ***source-file*** bevat het programma met de door de robot te ondernemen acties. Het programma is sequentieel opgebouwd. De te doorlopen stappen binnen het programma zijn weergegeven in figuur 24. Binnen elke stap is het bijbehorende technologiepakket weergegeven.



**Figuur 24: Flow-chart hoofdprogramma**

Het programma roept enkele vooraf gedefinieerde functies op uit de technologiepakketten EthernetKRLXML en ConveyorTech. Deze functies staan respectievelijk in tabel 3 en tabel 4. De functies vereisen invulling van enkele parameters, de bijbehorende technologiepakketten en de robotapplicatie gaan hier dieper op in.

**Tabel 3: Gebruikte functies binnen het technologiepakket EthernetKRLXML [10]**

Funcienaam in KRL	Beschrijving
EKX_Open()	Opent het opgegeven sensorkanaal via de XmlApiConfig.
EKX_Close()	Sluit het sensorkanaal.
EKX_WaitForSensorData()	Wacht tot de sturing connectie heeft met de sensor.
EKX_GetRealElement()	Neemt de gewenste real-waarde uit de XML-structuur.
EKX_GetIntegerElement()	Neemt de gewenste integer-waarde uit de XML-structuur.

**Tabel 4: gebruikte functies binnen het technologiepakket ConveyorTech [10]**

Funcienaam in KRL	Beschrijving
CONV_INI_OFF()	Initialiseert de conveyor, m.a.w. <i>reset</i> de verplaatste bandafstand naar nul.
CONV_ON()	Na het actief stellen van deze functie wacht het programma op de achtergrond op zijn synchronisatiesignaal.
CONV_FOLLOW()	Laat de robot aanrijden naar de te volgen punten op de conveyor.
CONV_QUIT()	Dient om de robot te laten stoppen bij een abnormale situatie zoals een noodstop of het overschrijden van de maximaal te volgen afstand.

De eerste stap binnen het programma is een initialisatiestap. Deze stap vervult drie verschillende functies: de programmaparameters resetten naar hun nul waarde, het definiëren van de XML-structuur voor het ontvangen van data en een initiële sluiting van de ethernetverbinding. De tweede stap bestaat uit het aanrijden van de home-positie. Vanuit deze

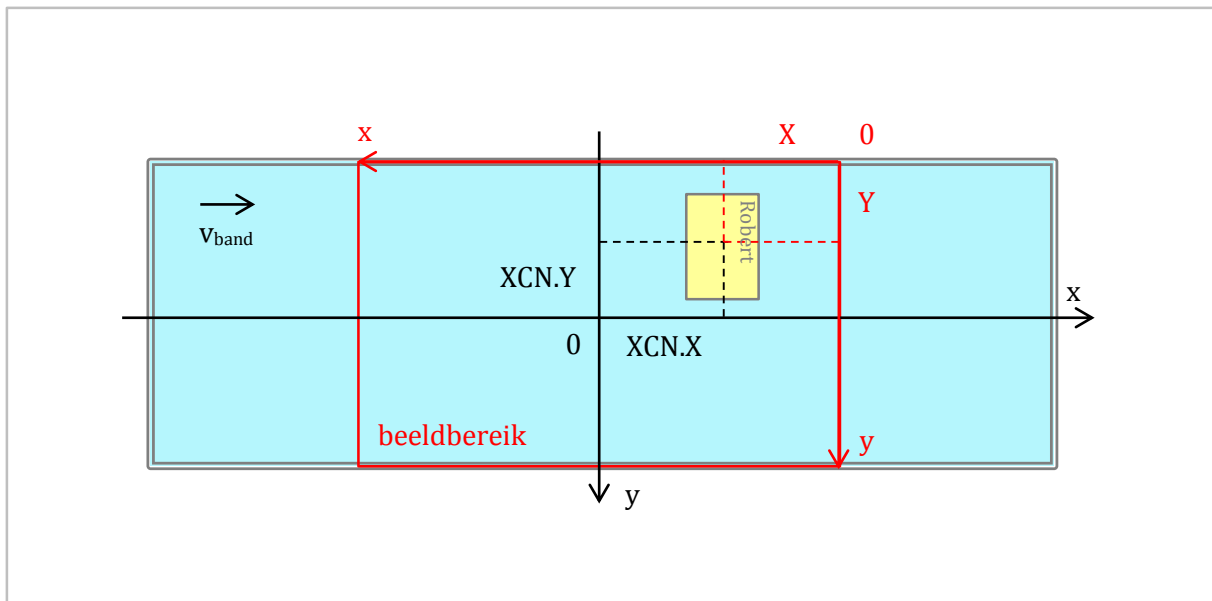
positie zal de robot naar het naamkaartje op de band bewegen. Deze positie is zo gekozen dat de afstand van home naar het punt op de band zo klein mogelijk is. In stap drie opent de robotapplicatie de dataverbinding. Hiervoor gebruikt het programma de functie `EKX_Open()` uit tabel 3. Bij het openen van de datalijn geeft het programma mee met welk type sensor hij een verbinding wil maken, in dit geval het visiesysteem. De functie `EKX_Open()` verwijst naar de `XmlApiConfig.xml`. Hier staan de gegevens van de sensor in zoals beschreven in paragraaf 3.1.1. De volgende stap, stap vier, omvat bandsynchronisatie door middel van `ConveyorTech`. Het programma activeert de bandsynchronisatie. Hiervoor zijn twee functies uit tabel 4 gedefinieerd binnen `ConveyorTech`. De eerste functie `CONV_INI_OFF()` reset de verplaatste bandafstand naar nul. De tweede functie `CONV_ON()` zet bandsynschronisatie actief, dit wil zeggen dat het programma op de achtergrond wacht op zijn synchronisatiepuls. Het visiesysteem *set* dit signaal via zijn digitale uitgang nadat een voorwerp is gedetecteerd. Binnen stap vijf gaat het programma wachten tot er een voorwerp gevonden is. De puls geeft aan dat een voorwerp is getraceerd. De `EKX`-functies uit tabel 3 lezen de gegevens in. De functie `EKX_WaitForSensorData()` leest de gehele structuur in. Daarnaast is de functie `EKX_GetRealElement()` gebruikt om `X`, `Y`, `A` en `T` in te lezen. De functie `EKX_GetIntegerElement()` leest ten slotte `K` in. Het programma slaat de ontvangen meetgegevens op in de bijbehorende variabelen met dezelfde naam, weergegeven in tabel 5. Indien de camera niets vindt binnen de ingestelde tijd zal de robot in *halt* gaan, m.a.w. het programma onderbreken.

**Tabel 5: Variabelen met de ontvangen meetgegevens in KRL**

Naam	Type	Betekenis
X	Real	De x-coördinaat van het middelpunt van het gevonden voorwerp in fixtures.
Y	Real	De y-coördinaat van het middelpunt van het gevonden voorwerp in fixtures.
A	Real	De hoek van het voorwerp rond de z-as in graden.
T	Real	De benodigde tijd voor beeldverwerking in ms.
K	Integer	Nummer van het gevonden voorwerp.

In stap zes berekent het programma de aan te rijden punten. De verstuurde gegevens uit tabel 5 staan in cameracoördinaten. Deze meetwaarden komen niet overeen met de coördinaten in het robotcoördinaatstelsel. De camera gebruikt de linkerhoek van het beeld als nulpunt. De robot heeft als nulpunt echter het midden van het beeld. De reden hiervoor is dat de camera geen voorwerpen kan herkennen in het hoekpunt van zijn beeld en bijgevolg deze coördinaten niet kan doorsturen. Ook is het midden nauwkeurig op te meten (minder distortie). Daarnaast is de richting van de x-as in het camera- en robotcoördinaatsysteem tegengesteld. Beide coördinaatstelsels zijn samengevat in figuur 25. Het camera-assenstelsel heeft geen z-as omdat het een 2D-beeld is. Het robotassenstelsel heeft een z-as gelegen loodrecht op de transportband met zijn richting positief in de transportband, de robot werkt dus in een rechtsdraaiend assenstelsel.





**Figuur 25: Grafische weergave van het camera- en robotassenstelsel**

De coördinaten X en Y beschrijven het middelpunt van het gevonden voorwerp in camera-coördinaten, met eenheden in *fixtures*. De robotsturing zet deze coördinaat om van *fixtures* naar mm. Tabel 6 geeft de afstanden in mm, pixels en *fixtures* voor de volledige grootte van het beeld en voor beide visiesystemen. De grootte van dit beeld in mm hangt af van de hoogte waarop de camera zich bevindt. Bij een verstelling en/of kalibratie van de camera veranderen deze waarden.

**Tabel 6: Afstanden voor de grootte van het beeld in verschillende eenheden**

Afstand	[mm]	[pixels]	[fixtures]
<b>Cognex 7010</b>			
X <sub>max</sub>	209,0	600	125400,0
Y <sub>max</sub>	278,0	800	222400,0
<b>IDS uEye-2230RE-C</b>			
X <sub>max</sub>	435,0	1024	445440,0
Y <sub>max</sub>	326,0	768	250368,0

De coördinaten van het gevonden voorwerp t.o.v. het robotcoördinaatsysteem zijn XC.N.X en XC.N.Y. Met behulp van de formules (4), (5) en (6) zijn deze coördinaten omgerekend voor beide visiesystemen. Ook de verdraaiing van het voorwerp is omgezet in robotcoördinaten. Na herkalibratie is een aanpassing van de formules vereist.

$$XC.N.X [mm] = 0,5 * X_{max} [mm] - \frac{X [fixtures]}{X_{max} [pixels]} \quad (4)$$

$$XC.N.Y [mm] = 0,5 * Y_{max} [mm] - \frac{Y [fixtures]}{Y_{max} [pixels]} \quad (5)$$

$$XC.N.A [^\circ] = XC.N.A [^\circ] + A [^\circ] \quad (6)$$

Vervolgens berekent het programma de aan te rijden punten ten opzichte van zijn nulpunt. De aan te passen coördinaten verschillen voor zowel de x- als y-richting. In de x-richting speelt er een afstand mee ten gevolge van de tijd bepaald door het verwerkingsproces van het visiesysteem.

De oorzaak van deze afstand is beschreven in paragraaf 2.1. In stap zes gaat de robot deze punten aanrijden met behulp van de geprogrammeerde *conveyor-base* die synchroon meeloopt met de transportband. Stap zeven bevat het desynchroniseren van de transportband. Hiervoor gebruikt het programma de functie CONV\_QUIT() uit tabel 4. De afgelegde bandafstand loopt hierdoor niet meer op. De robot gebruikt vanaf nu zijn vaste *base* voor het aanrijden van punten. In stap acht dienen de aangepaste punten uit stap zes terug aangepast naar de nul- of startpositie, namelijk het midden van het beeld. Bij het opnieuw doorlopen van het programma zal het programma nu steeds terug vanaf de nulpositie rekenen. De negende en laatste stap dient om de ethernetverbinding af te sluiten. Hiervoor gebruikt het programma de functie EKX\_Close() uit tabel 3. Over stap twee tot en met negen staat een loop geprogrammeerd. Deze stappen herhalen zich continu.

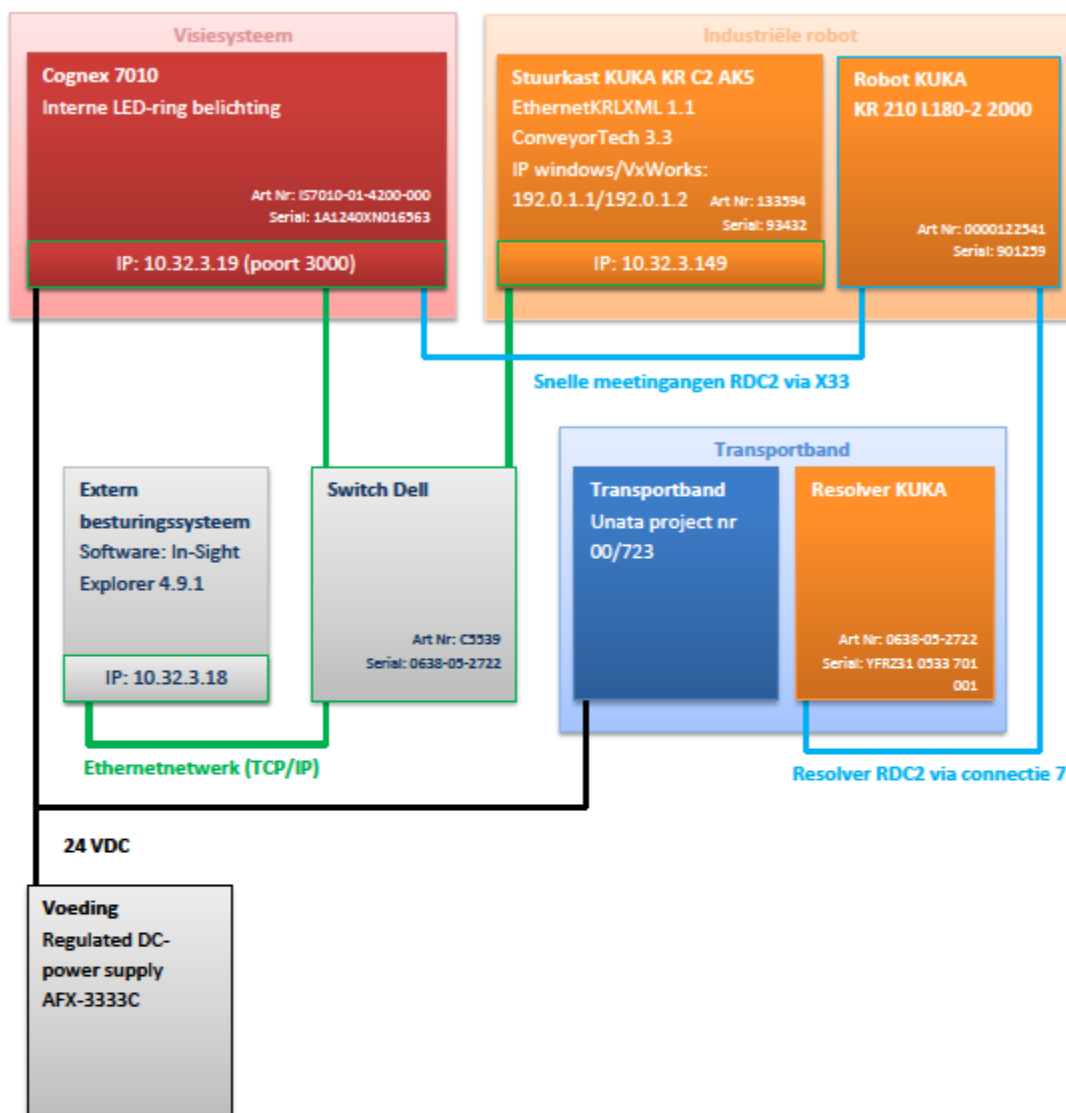
De ***data-file*** bevat alle lokale variabelen en aan te rijden punten. Bij elk punt staat zowel de bewegingssnelheid, de gebruikte *tool* en de gebruikte *base*. Tussentijdse resultaten van berekeningen zijn ook in deze *file* opgeslagen.

# 4 Visiesysteem Cognex 7010

De opstelling met het visiesysteem van Cognex is op te delen in twee bedrijfsmodi. De eerste modus geldt enkel tijdens de inbedrijfname of service. De tweede modus geldt indien de opstelling in bedrijf is genomen. Naar gelang de modus verandert de algemene opstelling. De volgende paragrafen beschrijven voor beide modi de componenten binnen de opstelling.

## 4.1 Inbedrijfnamemodus

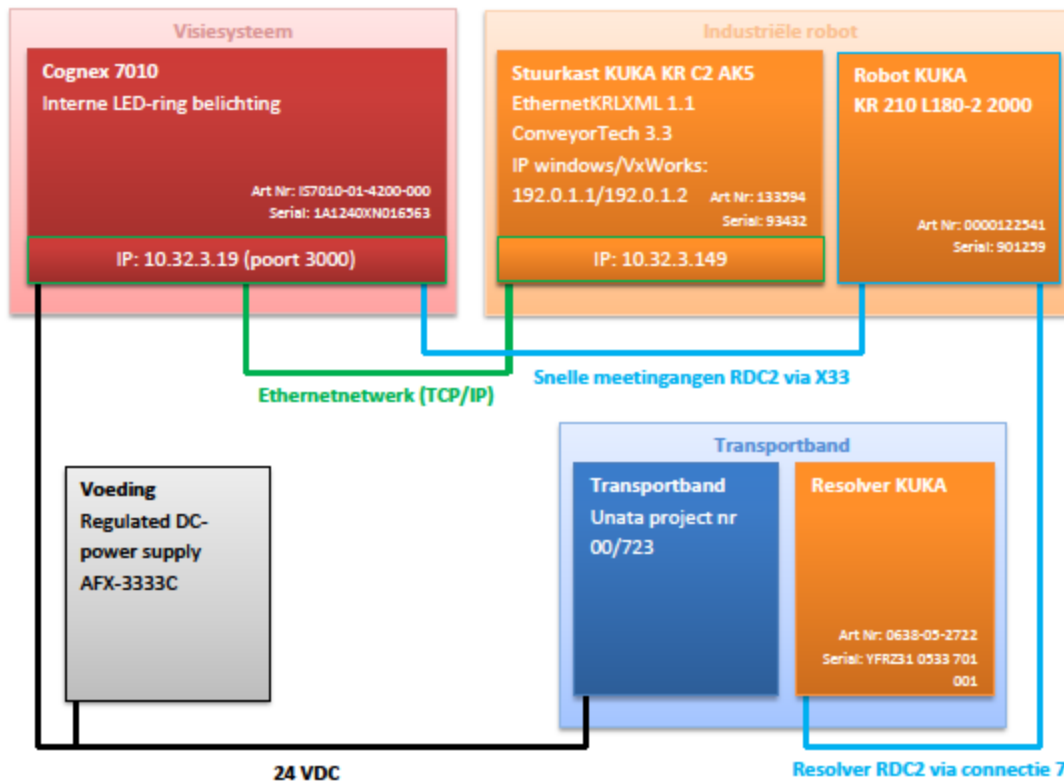
De opstelling bestaat uit het visiesysteem van Cognex type 7010, een extern besturingssysteem, de synchroband, de industriële robot met KRC2-sturing en de communicatie tussen de verschillende onderdelen. Dit is samengevat in het aansluitschema tijdens inbedrijfname, zie figuur 26.



Figuur 26: Aansluitschema visiesysteem Cognex7010 voor inbedrijfname

## 4.2 Bedrijfsmodus

Deze opstelling is nagenoeg gelijk aan de voorgaande opstelling. Enkel het extern besturingssysteem is hier weg genomen. Dit is te zien in het aansluitschema voor bedrijfsmodus, weergegeven in figuur 27.



*Figuur 27: Aansluitschema visiesysteem Cognex7010 voor bedrijf*

Zowel het visiesysteem als de synchroband zijn gevoed door een regelbare spanningsbron van 24 VDC. Daarnaast is de transportband aangesloten op 230 V.

Het ethernetnetwerk bevat volgende componenten zoals besproken in paragraaf 3.1: het visiesysteem, de KRC2-stuurkast, een externe PC en een switch. De computer in deze opstelling dient voor camera- en robotprogrammatie. De gebruiker kan deze PC in automatische bedrijfsmodus ontkoppelen van het systeem. In dit laatste geval is de camera rechtstreeks aangesloten op de stuurkast.

De RDC2-kaart als hoofdcomponent voor ConveryorTech is zoals besproken in paragraaf 3.2. De camera is via zijn I/O-connectie aangesloten op de RDC2-kaart via de snelle meetgangen. De resolver is aangesloten op de RDC2-kaart via de voorziene resolver-connectie.

### 4.3 Specificaties Cognex 7010

Het visiesysteem Cognex 7010 is een 2D-visiesysteem met geïntegreerde CPU voor beeldverwerking. Het visiesysteem heeft een *on-board* LED-belichting. Het visiesysteem is afgebeeld in figuur 28. De belangrijkste specificaties staan weergegeven in tabel 7. Verdere specificaties staan in bijlage 14.



Tabel 7: Specificaties Cognex 7010 [13]

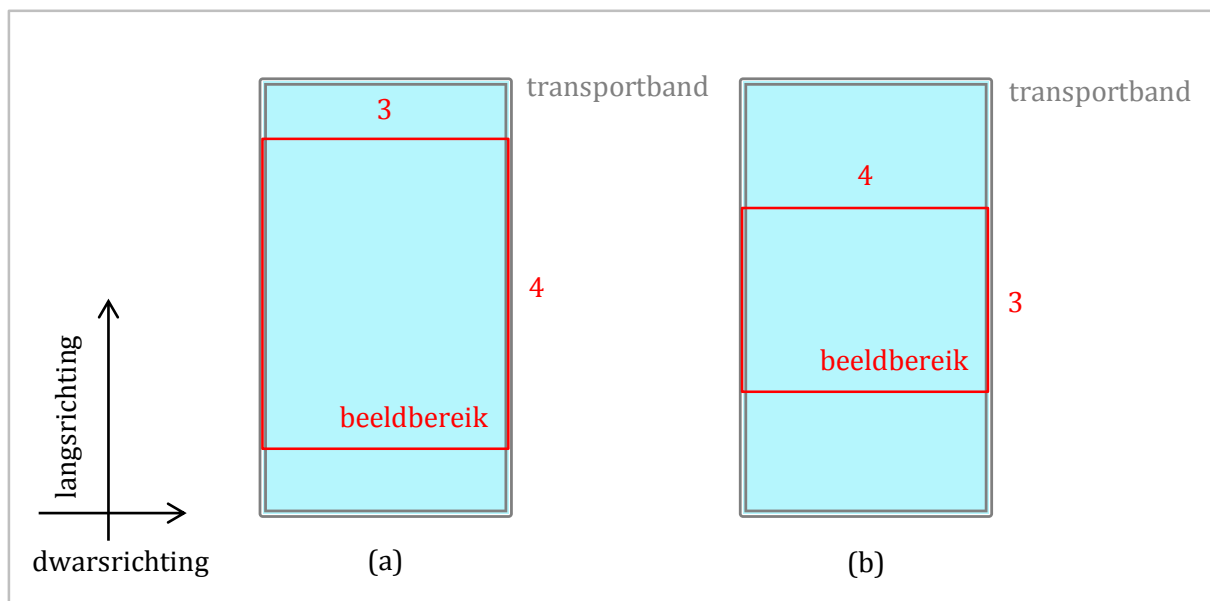
Beeldverhouding	4:3
Resolutie	800 x 600
Technologie	CMOS zwart-wit
Max framerate	102
Sensor grootte diagonaal	5,3 mm
Digitale ingangen	3
Digitale uitgangen	4

Figuur 28: Visiesysteem Cognex 7010 [14]

De camera heeft drie aansluitingen. De eerste is een gecombineerde aansluiting. Deze connector voorziet de camera van I/O en spanning. De tweede is een ethernet aansluiting voor netwerkcommunicatie en de derde is een aansluiting voor externe belichting. Deze laatste aansluiting is in de opstelling niet gebruikt aangezien de LED-ringbelichting voldoet voor deze toepassing.

#### 4.3.1 Cameramontage

De juiste montage van de camera is bepaald door het beeld van de camera. De montage bepaalt namelijk de *field of view*. Deze oppervlakte is onderverdeeld in een *horizontal*- en *vertical field of view* met een beeldverhouding van 4:3. Daarnaast heeft de transportband een langs- en dwarsrichting. Hierdoor zijn er twee plaatsingsmogelijkheden: een optimale en een alternatieve wijze die is bepaald door de montage van de camera, weergegeven in figuur 29.



Figuur 29: (a) optimale configuratie (b) alternatieve configuratie

De optimale configuratie omvat een groter oppervlak op de transportband. Om dit te verwezenlijken zal de benodigde lens ten eerste een grotere *focal length* nodig hebben. Ten tweede heeft de lens een klein diafragma nodig om een scherp gesteld beeld te verkrijgen.

Voor de lens in de optimale configuratie is de formule uit paragraaf 1.1. van toepassing. De *field of view* van de transportband is opgemeten op 400 x 300 mm. De gewenste *working distance* bedraagt 1,15 m. De specificaties van de camera geven aan dat de camerasensor een diagonaal heeft van 5,3 mm, zie Tabel 7. De *focal length* van de lens bedraagt dan:

$$f' = \frac{g}{\left(\frac{G}{B} + 1\right)} = \frac{1100}{\left(\frac{400}{5,3 * \frac{4}{5}} + 1\right)} = 11,54 \text{ mm}$$

Een lens met *focal length* 11,54 mm is vereist. Bij de camera van Cognex zijn enkele standaard lenzen geleverd. De dichtstbijzijnde lens heeft een *focal length* van 12 mm. Tijdens het gebruik van deze configuratie blijkt dat de camera geen scherpstelling kan bekomen. Deze lens voldoet niet omdat het diafragma F/2 te groot is.

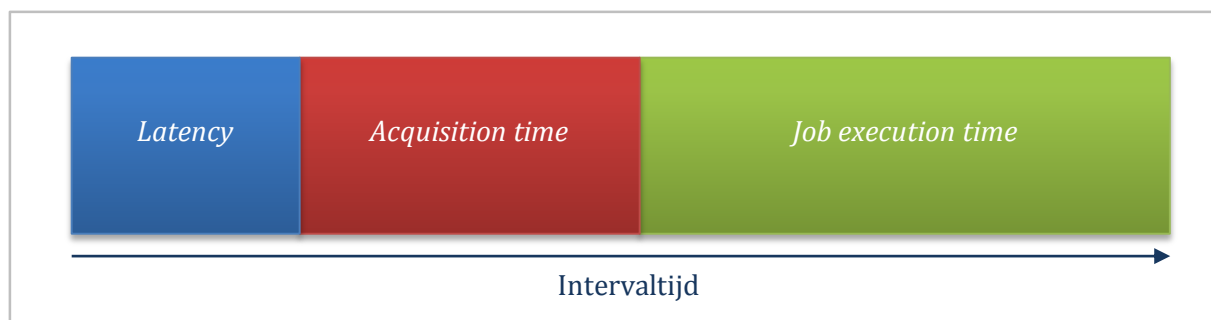
Aangezien er geen andere lens is met een kleiner diafragma is de alternatieve configuratie gehanteerd. De *field of view* van de transportband is opgemeten op 278 x 209 mm. De *working distance* en breedte van de sensor blijven identiek aan voorgaande configuratie en bedragen respectievelijk 1,15 m en 5,3 mm. De *focal length* bedraagt dan:

$$f' = \frac{g}{\left(\frac{G}{B} + 1\right)} = \frac{1100}{\left(\frac{278}{5,3 * \frac{4}{5}} + 1\right)} = 16,52 \text{ mm}$$

De dichtstbijzijnde beschikbare lens heeft een *focal length* van 16 mm en een diafragma van F/2. Met deze lens is het beeld goed scherp te stellen. De alternatieve configuratie is dus gehanteerd voor de opstelling met het visiesysteem van Cognex.

### 4.3.2 Software

Het visiesysteem maakt gebruik van het softwarepakket In-Sight Explorer 4.9.1. Deze software ondersteunt beeldverwerking, I/O-aansturing en netwerkcommunicatie met de robot. De programma's zijn terug te vinden in bijlage 15 en 16. In bedrijf zal de camera beelden nemen met een vaste intervaltijd. Deze vaste intervaltijd is op te delen in een aantal deeltijden. Dit is weergegeven in figuur 30.



**Figuur 30: Intervaltijd voor het nemen van een beeld**

De totale tijd bestaat uit een *latency*, een *acquisition time* en een *job execution time*. De *latency* is de tijd nodig om het visiesysteem te triggeren. Aangezien de camera intern is getriggerd beperkt deze tijd zich tot 250  $\mu$ s. De *acquisition time* is de tijd voor de belichting en de tijd voor het transfereren van de afbeelding naar het verwerkingsproces. Deze tijd is dus afhankelijk van het omgevingslicht aangezien er in een donkere omgeving een langere of sterkere belichting nodig is. Daarnaast is de tijd afhankelijk van de *region of interest* (ROI). Bij een grotere ROI is de te transfereren data groter waardoor deze tijd stijgt. De *job execution time* bestaat uit de tijd voor de beeldverwerking. Deze rekentijd hangt binnen de beeldverwerking af van verschillende elementen [15]:

- de *tool* voor het zoeken van het voorwerp binnen het beeld;
- de manier voor het aanleren van voorwerpen;
- de uitvoering van een aantal logische functies.

De gehanteerde *tool* in deze opstelling is de *pattern-tool*. Deze *tool* heeft twee functies. De eerste is een *teach-in*-functie om een patroon aan te leren. De tweede is een functie om dit aangeleerde patroon binnen een beeld te herkennen. De *tool* werkt *edge-based* [15]. Dit wil zeggen dat hij randen herkent en vergelijkt. Op basis daarvan is bepaald of het voorwerp overeen komt met het aangeleerde voorwerp. Binnen de *pattern-tool* zijn er drie hoofdparameters waarvan de eerste de *accept threshold* is. Deze parameter beschrijft hoeveel beide patronen minimum overeen moeten komen. De *accept threshold* dient zo ingesteld te zijn dat enerzijds het visiesysteem het voorwerp steeds herkent, maar anderzijds andere voorwerpen niet aanziet voor het aangeleerde voorwerp. Hoe lager de *accept threshold* staat ingesteld, hoe meer rekentijd dit vergt voor de camera. Ten tweede is er de *rotation tolerance*. Deze parameter beschrijft de minimale en maximale hoek waaronder het voorwerp kan liggen. Het totale bereik van de verdraaiing van het voorwerp is dus steeds twee keer de *rotation tolerance*. De rekentijd van de camera vergroot recht evenredig met een vergroting van deze parameter. De derde en laatste parameter is een schaalfactor. Indien het voorwerp onder een andere grootte is aangeleerd is het nuttig de schaalfactor aan te vinken. Het schalen van voorwerpen vergt meer rekentijd voor het visiesysteem. De totale tijd die nodig is voor het uitvoeren van de tool is bovendien ook nog afhankelijk van de *search region* [15]. De breedte van het beeld moet steeds de volledige transportband omvatten waardoor enkel de hoogte van het beeld aanpasbaar is. Een verminderde hoogte heeft als voordeel dat de tool niet op elke pixel is toegepast waardoor de rekentijd voor deze tool verkleint. Tegenstrijdig hieraan is het moeilijker om voorwerpen te herkennen omdat de hoogte van het beeld wijzigt, zie paragraaf 1.3. Proefondervindelijk blijkt dat het visiesysteem de beste prestaties levert indien de hoogte van het beeld niet gereduceerd is. De *search region* is dus in het beste geval even groot als het beeld.

Het aanleren van het voorwerp d.m.v. de *pattern-tool* kan op twee manieren. Enerzijds door het echte voorwerp te gebruiken. Bij de instelling m.b.v. het echte voorwerp is dit voorwerp op de transportband geplaatst en neemt het visiesysteem hiervan een beeld met de camera. Het voordeel van deze methode is dat het naamkaartje steeds dezelfde grootte heeft op het beeld waardoor de schaalfactor niet meespeelt. De schaalfactor vergroot de rekentijd dus niet. Anderzijds is het mogelijk het voorwerp aan te leren m.b.v. een foto van het voorwerp in JPG-formaat. In dit geval is het voorwerp anders geschaald dan de voorwerpen die voorbijkomen op de transportband. De schaalfactor vergroot hier de rekentijd echter wel. Daarboven is het misschien noodzakelijk de *accept tolerance* te verlagen aangezien de belichting tijdens het nemen van de foto niet altijd overeenkomt met de belichting in de opstelling. Hierdoor bestaat het gevaar

dat het visiesysteem foutieve voorwerpen herkent. Het inladen van technische tekeningen in DXF-formaat is niet mogelijk met de Cognex-smartcamera.

Logische functies zijn gebruikt voor het opbouwen van de XML-structuur uit paragraaf 3.1.1. De XML-*string* bestaat uit de coördinaten van de te zoeken voorwerpen. Indien de software de coördinaten rechtstreeks uit de *patterns* haalt, krijgt de software enkel een waarde wanneer het voorwerp in beeld ligt. Bij het zoeken naar een enkel voorwerp levert dit geen probleem op. Indien er daarentegen is gezocht naar meerdere voorwerpen is dit echter wel een probleem. In veel gevallen ligt er slechts één van de deze voorwerpen in beeld. De software vult dan slechts voor één model de coördinaten in. De coördinaten van de andere modellen zijn leeg<sup>3</sup>. Hierdoor ontstaat er een ongewenste *error*. Om dit probleem te verhelpen gebruikt de visieapplicatie de parameter *k* uit paragraaf 3.1.1. Deze parameter is bepaald door logische functies. Figuur 31 tot en met figuur 35 geven deze logische functies weer voor twee *patterns*. Binnen In-Sight Explorer noemen deze twee *patterns* Pattern\_1 en Pattern\_2. De eerste drie parameters Pattern1X, Pattern1Y en Pattern1A zijn bepaald door een *if*-lus. Figuur 31 geeft hier de logische functies voor weer. Indien Pattern\_1 is gevonden krijgen deze parameters de coördinaten van het voorwerp. Is het voorwerp echter niet gevonden, dan krijgen ze de waarde nul. Op deze manier is de parameter ingevuld en kan er geen *error* optreden. Voor Pattern\_2 zijn gelijkaardige functies opgebouwd.

```
Pattern1X = If(Pattern_1.Fail, 0, Pattern_1.Fixture.X)
Pattern1Y = If(Pattern_1.Fail, 0, Pattern_1.Fixture.Y)
Pattern1A = If(Pattern_1.Fail, 0, Pattern_1.Fixture.Angle)
Pattern2X = If(Pattern_2.Fail, 0, Pattern_2.Fixture.X)
Pattern2Y = If(Pattern_2.Fail, 0, Pattern_2.Fixture.Y)
Pattern2A = If(Pattern_2.Fail, 0, Pattern_2.Fixture.Angle)
```

**Figuur 31: Logische functies om de coördinaten van de modellen in te vullen**

Vervolgens moet de parameter *k* een waarde krijgen naar gelang het gevonden voorwerp. Figuur 32 illustreert hoe deze parameter is bepaald. De eerste *if*-lus stelt *k* gelijk aan nul indien beide voorwerpen zijn gevonden. De robot zal in dit geval geen voorwerp volgen<sup>4</sup>. De eerste *elseif*-lus stelt dat *k* nul is indien geen enkel voorwerp is gevonden. De tweede *elseif*-lus stelt dat *k* gelijk is aan twee omdat Pattern\_1 niet is gevonden en niet voldaan is aan de twee bovenstaande voorwaarden. De *else*-lus stelt *k* gelijk aan één omdat aan geen enkele bovenstaande voorwaarde is voldaan.

```
If(Not(Pattern_1.Fail) And Not(Pattern_2.Fail)) Then
    K = 0
ElseIf(Pattern_1.Fail And Pattern_2.Fail) Then
    K = 0
ElseIf(Pattern_1.Fail) Then
```

<sup>3</sup> Een lege parameter is niet gelijk aan de waarde nul.

<sup>4</sup> In dit geval liggen de voorwerpen zo dicht bij elkaar dat de robot mogelijk te traag is om beide objecten van de transportband te volgen. Vandaar de keuze om geen van beide voorwerpen te volgen.



```

        K = 2
    Else
        K = 1
    EndIf

```

**Figuur 32: Invullen van de parameter k**

De volgende stap dient om de coördinaten van het overeenkomstig voorwerp te selecteren. Deze stap is vereist omdat de software slechts één coördinatenpaar kan doorsturen. Indien de waarde k gelijk is aan nul, stelt de *if*-lus XTotaal, YTotaal en ATotaal ook gelijk aan nul. Heeft k de waarde één of twee dan zijn dergelijke parameters gelijk aan de som van de overeenkomstige coördinaten van Pattern\_1 en Pattern\_2. Een van beide coördinaten is dan gelijk aan nul waardoor de variabele de coördinaat van het gevonden voorwerp bevat. Een vierde bijkomende parameter is de verwerkingstijd. Dit is de som van alle deelverwerkingstijden aangezien iedere logische functie een extra rekentijd vergt. Figuur 33 geeft de berekening voor deze vier parameters.

```

XTotaal = If(K.Result = 0, 0, Sum(Pattern1X.Result, Pattern2X.Result))
YTotaal = If(K.Result = 0, 0, Sum(Pattern1Y.Result, Pattern2Y.Result))
ATotaal = If(K.Result = 0, 0, Sum(Pattern1A.Result, Pattern2A.Result))
Tijd = Sum(Pattern_1.Execution_Time, Pattern_2.Execution_Time,
Pattern1A.Execution_Time, Pattern1X.Execution_Time, Pattern1Y.Execution_Time,
Pattern2A.Execution_Time, Pattern2X.Execution_Time, Pattern2Y.Execution_Time,
Xtotaal.Execution_Time, Ytotaal.Execution_Time, Atotaal.Execution_Time,
K.Execution_Time)

```

**Figuur 33: Berekenen van de parameters voor de te verzenden XML-string**

De XML-string neemt vervolgens deze vier parameters op voor verzenden. De totale doorgestuurde file ziet er dan uit zoals figuur 34.

```

<Data>
    <X>Xtotaal.Result</X>
    <Y>Ytotaal.Result</Y>
    <A>Atotaal.Result</A>
    <T>Tijd.Result</T>
    <K>k.Result</K>
</Data>

```

**Figuur 34: Totale XML-structuur**

De laatste stap is het uitsturen van de triggerpuls indien er een voorwerp is gedetecteerd. Een *or*-functie zorgt voor deze stap zoals weergegeven in figuur 35. De output van de camera moet triggeren telkens er een voorwerp is gevonden. Door de waarde van pass hoog te maken indien voorwerp één of twee is gevonden, set het visiesysteem de puls hoog. Een belangrijke opmerking bij deze logische functie is dat de *range limits* moeten staan op minimum één en maximum één. Bij een foutieve instelling van de *range limits* zal de variabele pass steeds *true* zijn.

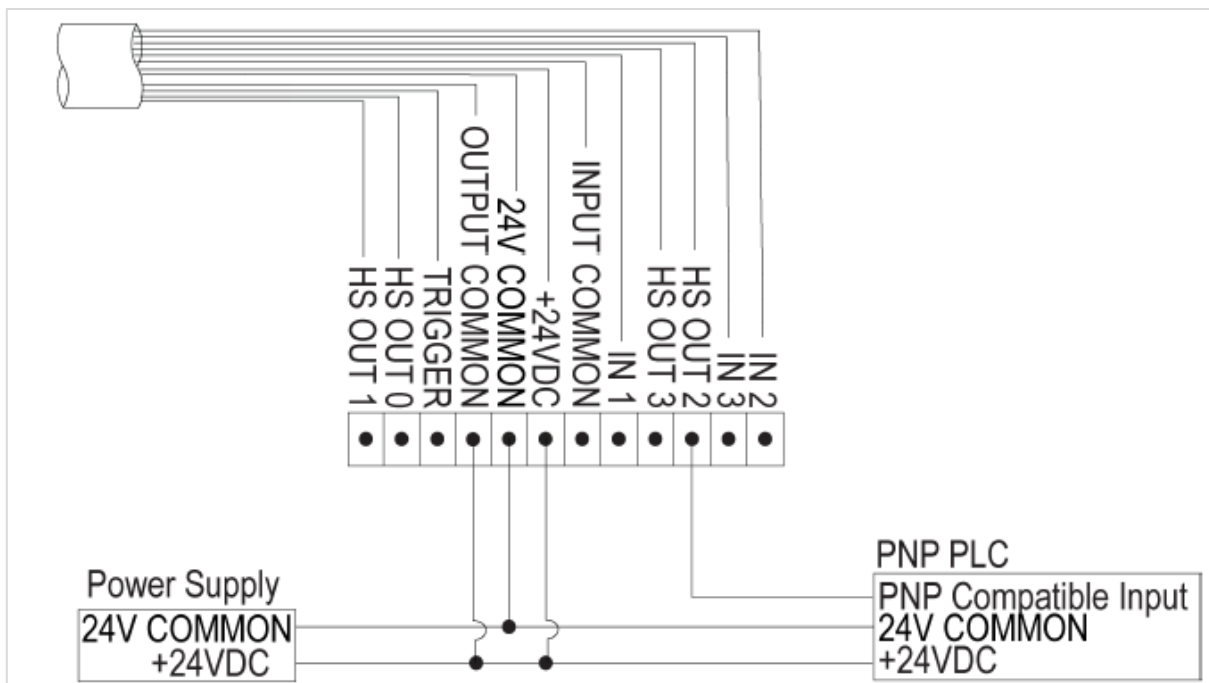
Pass = Or(Pattern\_1.Pass, Pattern\_2.Pass)

**Figuur 35: Job-pass triggering**

Zoals eerder vermeld bevat de *job execution time* de beeldverwerkingstijd en de tijd voor het uitvoeren van de logische functies. Ten slotte bevat deze *job execution time* ook de tijd nodig voor het uitsturen van de triggerpuls.

### 4.3.3 I/O-configuratie

Het Cognex-visiesysteem voorziet dus de 24 VDC-synchronisatiepuls indien de camera een voorwerp herkent. Het visiesysteem zet de puls hoog via zijn uitgang aangesloten op de gecombineerde aansluiting naar de RDC2-kaart van de robot. Deze snelle uitgang is op twee manieren instelbaar, enerzijds als *current sinking* (NPN-mode) of anderzijds als *current sourcing* (PNP-mode). De instelling is volgens het PNP-type aangezien de robot wacht op een positieve puls. De instelling gebeurt door de output common van de I/O-kabel te verbinden met de +24 VDC-ingang van de spanningsbron en de +24 VDC-ingang van de RDC2-kaart [13]. Deze laatste verbinding gaat via de X33-aansluiting op de robot naar deze kaart. De onderlinge verbinding tussen het visiesysteem en de robot is schematisch weergegeven in figuur 36.



**Figuur 36: Current sourcing aansluitprincipe [13]**

Vervolgens is een instelling van de software noodzakelijk. Deze instelling gebeurt via de In-Sight Explorer onder het tabblad *output*. In totaal bevat het visiesysteem vier high-speed outputs. De geselecteerde *output* is HS OUT 1. De lengte van de puls is instelbaar, de standaard waarde van 10 ms volstaat.

## 4.4 Verbetermogelijkheden

### 4.4.1 Lenskeuze

De eerste verbetering bestaat erin de alternatieve configuratie te vervangen door de optimale configuratie zoals besproken in paragraaf 4.3.1. Om dit te verwezenlijken moet de camera 90° draaien op de statief. De huidige lens met een *focal length* van 16 mm voldoet niet. Deze lens is het beste te vervangen door een lens van 12 mm samen met een kleiner diafragma dan de huidige F/2. Een kleiner diafragma is aangeduid met een groter getal. Een lens met diafragma F/2.5 of F/4 is dus aangewezen voor een goede scherpstelling. Door gebruik te maken van de optimale configuratie vergroot de hoogte van het beeld ongeveer met factor twee. Dezelfde camera kan in dit geval hogere bandsnelheden verwerken, dit principe is beschreven in paragraaf 1.3.

### 4.4.2 Externe triggermogelijkheid

De huidige werking met een continue trigger voor het nemen van beelden brengt een nadeel met zich mee. De maximale verwerkingsnelheid is afhankelijk van de bandsnelheid. Stijgt de bandsnelheid, dan is de huidige verwerkingstijd van het visiesysteem misschien te groot waardoor mogelijk niet heel de transportband is gescand. Daarnaast moet de robotsturing voor elke bandsnelheid eerst de module snelheidsmeting doorlopen zoals eerder vermeld. Bij een wijzigende bandsnelheid komt de werkelijke snelheid niet meer overeen met de gemeten snelheid. Uit deze twee redenen volgt dat de band tijdens bedrijf niet kan versnellen of vertragen.

Dit probleem valt te verhelpen door enerzijds het Cognex-systeem via een externe ingang te *triggeren*. Een resolver en uitleesmodule kunnen het visiesysteem voorzien van dergelijke pulsen. De KUKA-resolver die reeds op de transportband bevestigd is, is echter niet bruikbaar. De gegenereerde pulsen uit de resolversignalen zitten te diep in de robothardware verwerkt waardoor deze signalen niet toegankelijk zijn. Een tweede sensor is dus aangewezen. Dergelijke sensor kan eventueel een tweede resolver zijn. Anderzijds moet de robotsturing de bandsnelheid tijdens bedrijf continu opmeten.

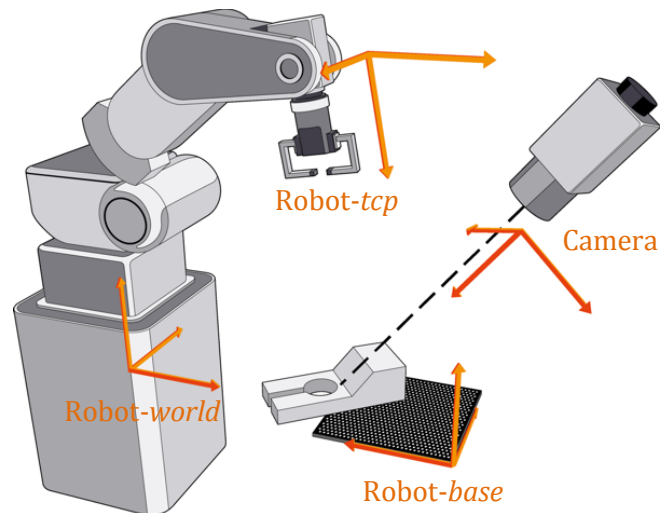
### 4.4.3 Robot – camera kalibratie

Robot – camera kalibratie dient om de coördinaten van het visiesysteem rechtstreeks om te vormen naar de coördinaten in robot-*world* zoals in Figuur 37. De oriëntatie van het assenstelsel van de camera ligt vast. Het is enkel mogelijk om de x- en y-as om te wisselen en om de zinnen van deze assen te kiezen. De eerste stap in de kalibratie van de camera is het loodrecht opstellen van de camera t.o.v. de transportband waarbij het beeld evenwijdig ligt met de transportband. De tweede stap is het omzetten van de eenheid waarin de camera werkt. De robot werkt in de eenheid mm en het visiesysteem werkt standaard in de eenheid pixels. De kalibratie van de camera moet ervoor zorgen dat de coördinaten van het voorwerp gevonden door de camera in dezelfde eenheid als de eenheid waarin de robot werkt staan. De omzetting van het robot-*base*-coördinatenstelsel naar het robot-*world*-coördinatenstelsel gebeurt automatisch in de robotsturing<sup>5</sup>. In de thesis-opstelling zijn er twee basissen opgemeten: de eerste is een vaste *base* van de transportband en de tweede is een bewegende *base* opgemeten door ConveyorTech, dit

---

<sup>5</sup> Dit is enkel zo indien er een *base* is opgemeten.

principe is beschreven in paragraaf 3.3.1. De omzetting van camera-coördinaten naar robot-*base*-coördinaten gebeurt nu echter nog handmatig met de formules uit paragraaf 3.3.3. Figuur 37 illustreert al deze coördinaatstelsels.



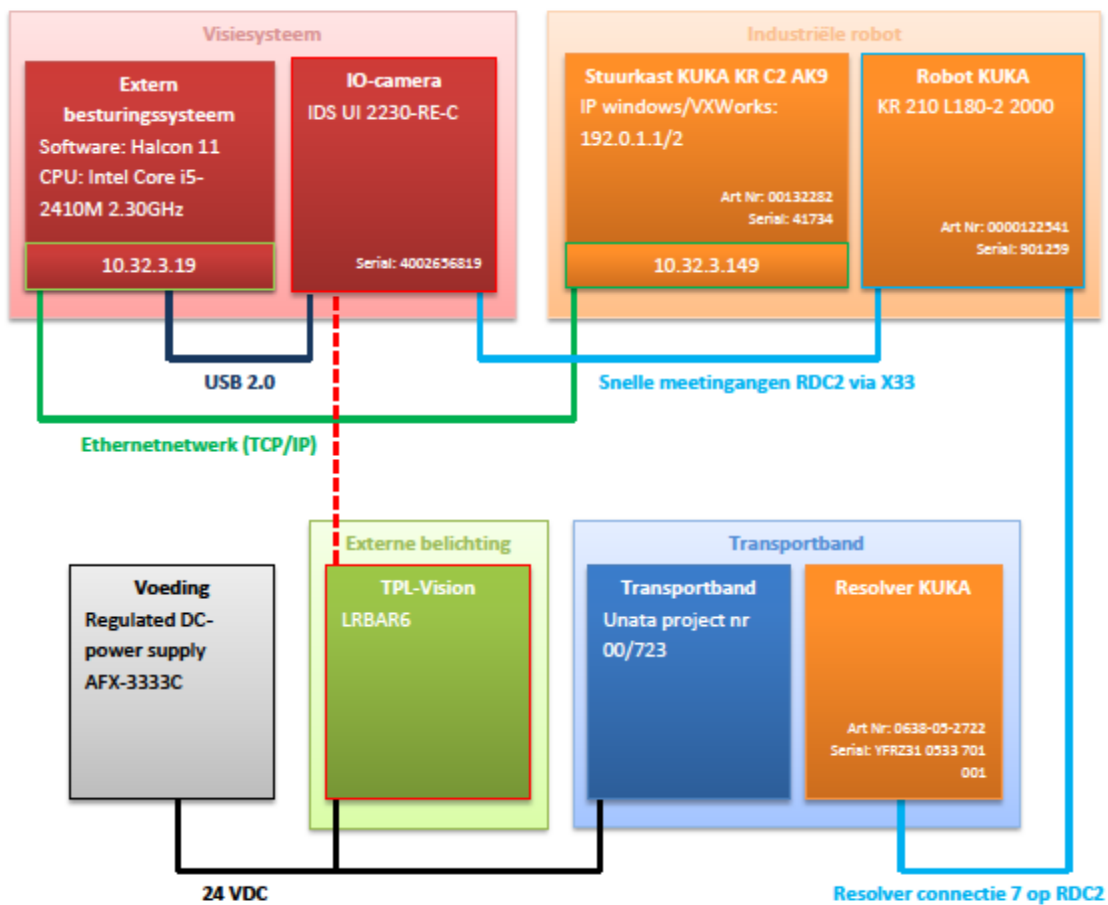
***Figuur 37: Verschillende coördinaatstelsels binnen de opstelling [16]***

Het nadeel van deze handmatige omzetting tussen robot-*base* en camera is dat de configuratie van ConveyorTech apart gebeurt vanuit het middelpunt van het camerabeeld. Het is daarom nuttig om een procedure uit te schrijven die zowel de robot-camera kalibratie als ook ConveyorTech configuratie ineens uitvoert. De procedure kan dan bestaan uit een methode waarbij enkele punten binnen het camerabeeld zijn opgemeten en vervolgens verwerkt.

# 5 Visiesysteem IDS uEye-2230RE-C en Halcon

## 5.1 Algemeen

De tweede opstelling bestaat uit het visiesysteem IDS uEye-2230RE-C, een computer met daarop de beeldverwerkingssoftware Halcon 11, de synchroband en de industriële robot met sturing. De opstelling staat afgebeeld in figuur 38.



**Figuur 38: Aansluitschema opstelling IDS uEye-RE2330-C**

In tegenstelling tot het visiesysteem van Cognex uit voorgaande opstelling heeft dit systeem van IDS geen interne CPU. Het visiesysteem bestaat dus niet enkel uit een camera maar bevat ook een extern besturingssysteem voor beeldverwerking. Het gehanteerde software-pakket voor de beeldverwerking is Halcon 11 van MVTec. De connectie tussen de computer en de camera bestaat uit een USB 2.0 connectie die eveneens de camera voedt. Deze connectie is niet *real-time* waardoor het mogelijk beperkingen legt op het systeem. Daarnaast heeft deze camera geen belichting. Een externe belichting is dus vereist. Hiervoor is een LED-bar met zes *power-LED's* gebruikt. De camera bevat I/O voor het versturen van de synchronisatiepuls. De robotsturing, transportband en voeding zijn gelijk aan de voorgaande opstelling. Alsook de connecties tussen deze componenten.

## 5.2 Specificaties visiesysteem IDS RE2330-C en Halcon

Het visiesysteem IDS uEye-2230RE-C is een 2D-visiesysteem. De beeldverwerking vindt plaats op een extern besturingssysteem, binnen de thesis een PC. De camera heeft een connectie voor rechtstreekse I/O. De communicatie over ethernet loopt via het extern besturingssysteem. De camera is afgebeeld in figuur 39. De belangrijkste specificaties staan weergegeven in tabel 8. Verdere specificaties staan in bijlage 17.



Figuur 39: IDS uEye-2230RE-C camera [17]

Tabel 8: Specificaties IDS uEye-2230RE-C camera en extern besturingssysteem [17]

Camera	
Beeldverhouding	3:4
Resolutie	768 x 1024
Technologie	CCD Color
frame-rate	40 fps
Sensor grootte diagonaal	5,95 mm
Digitale ingangen	1
Digitale uitgangen	1
Extern besturingssysteem	
CPU	Intel Core i5-2410M 2.30GHz
Besturingssysteem	Windows 7
Software	Halcon 11

### 5.2.1 Cameramontage

Voor de optimale configuratie, waarbij de langste richting van het beeld gelijk ligt met de langs richting van de transportband, is de vereiste lens berekend met de formule uit paragraaf 1.1. De *field of view* van de transportband is opgemeten op 435 x 326 mm. De gewenste *working distance* bedraagt 1,1 m. De specificaties van de camera geven aan dat de camerasensor een diagonaal heeft van 5,95 mm, zie tabel 8. De *focal length* van de lens is dan als volgt:

$$f' = \frac{g}{\left(\frac{G}{B} + 1\right)} = \frac{1100}{\left(\frac{435}{5,95 * \frac{4}{5}} + 1\right)} = 18,49 \text{ mm}$$

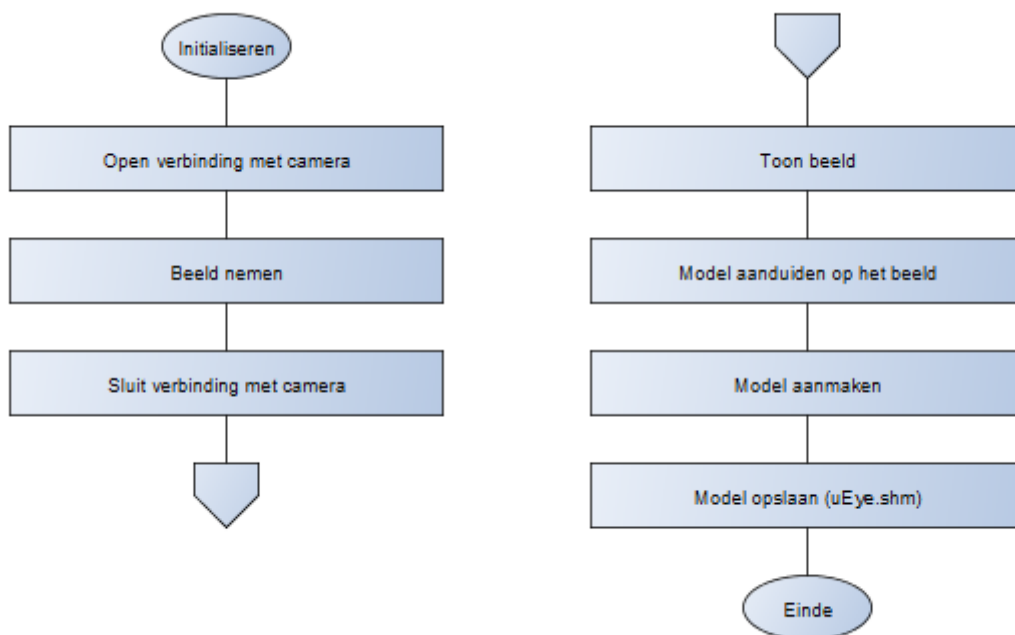
De gebruikte lens is een verstelbaar model van Pentax. Het bereik voor de *focal length* en het diafragma liggen respectievelijk tussen 8 - 48 mm en F/1.2 - F/22. Met deze lens verkrijgt de camera scherpe beelden. De optimale configuratie is dus gehanteerd voor de opstelling met het visiesysteem van IDS.

## 5.2.2 Software

Het visiesysteem is geprogrammeerd met de software Halcon 11 van MvTec. Zoals eerder aangegeven beschikt de camera zelf niet over een CPU waardoor de beeldverwerking continu gebeurt op een externe pc die met de camera verbonden is via USB 2.0.

De flowchart van het programma om het object aan te leren is weergegeven in figuur 40. De Halcon-applicatie is terug te vinden in de bijlagen 18, 19, 20 en 21. De eerste stap geeft aan dat het aan te maken model van het type *border shape* is. Dit model bestaat uit de contouren van het voorwerp. Voor naamkaartjes is dit de beste keuze aangezien de vormen van de tekst/afbeeldingen op het naamkaartje ook in het model zijn opgenomen. Het visiesysteem kan zo twee naamkaartjes met dezelfde grootte onderscheiden.

De tweede stap opent de verbinding met de camera en geeft bij de verbinding enkele parameters van de camera op. Bij stap drie neemt de camera een afbeelding en bij stap vier sluit de computer de verbinding met de camera. Het programma laat nu de afbeelding zien zodat de gebruiker het model kan selecteren op het beeld. Vervolgens slaat het programma het model op in de *file* uEye.shm.



**Figuur 40:** Flowchart voor het aanleren van een object

De volgende stap is de tweede applicatie starten. Deze applicatie bevat het continu zoeken van het aangeleerde model. De flowchart van dit programma is weergegeven in figuur 41. De code van dit programma is terug te vinden in bijlage 22, 23, 24 en 25.

Het tweede programma start met de initialisatiestap om de parameters voor de communicatie in te lezen. In de tweede stap gaat het programma na of het modelbestand uEye.shm bestaat. Indien dit niet zo is springt de code door naar het einde en krijgt de gebruiker een melding met als boodschap: "Could not find the model file!". Als het model wel aanwezig is leest het programma dit in. Vervolgens opent het programma de USB-verbinding met de camera

en de ethernetverbinding met de robot. Nu de verbindingen allemaal geopend zijn kan het cyclisch zoeken naar het model beginnen.

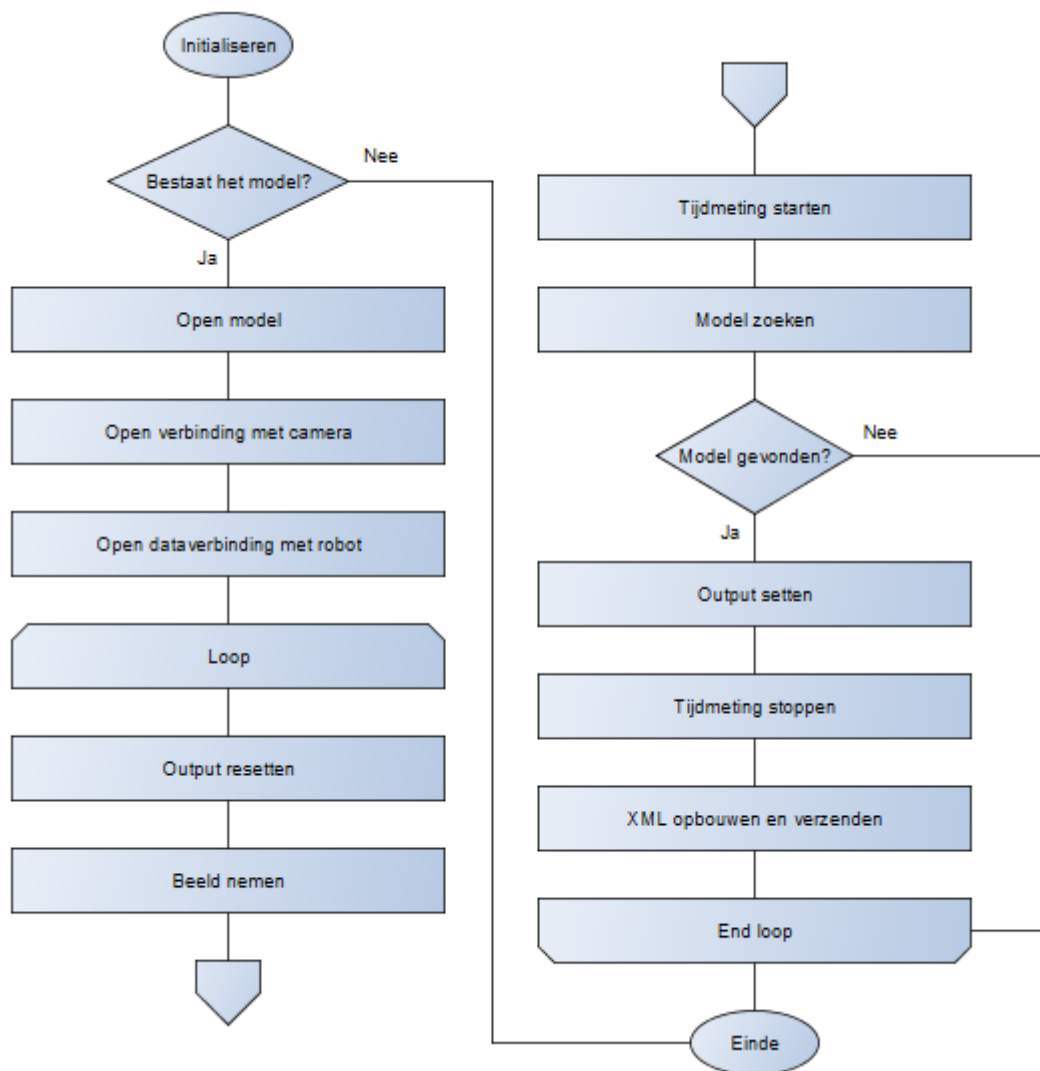
De eerste stap van het cyclisch zoeken *reset* de output zodanig dat de robot geen synchronisatiesignaal ontvangt. De *loop* opent en de camera neemt een afbeelding. Hierna start een tijdmeting. Deze tijdmeting is nodig om de verwerkingstijd weg te kunnen regelen zodanig dat de robot niet achter gaat lopen op het naamkaartje. Vervolgens zoekt de software het model in het beeld. Als het model niet gevonden is begint de loop terug vanaf het begin. Indien het model daarentegen wel gevonden is, geeft de instructie de parameters  $x$ ,  $y$  en *angle* als resultaat terug. Vervolgens *set* de instructie de output voor de puls hoog. Er gebeurt een tweede tijdmeting om de eindtijd van het proces vast te stellen. Door de begintijd van de eindtijd af te trekken is de totale verwerkingstijd van het visiesysteem bekend.

De camera kan slechts een nieuw beeld nemen indien voorgaande stappen zijn doorlopen. Deze verwerkingssnelheid bedraagt ongeveer 100 ms per programmacclus. Hierdoor bestaat de mogelijkheid dat de camera het object meerdere malen vindt terwijl dit object slechts eenmaal voorbij de camera passeert<sup>6</sup>. ConveyorTech aan de robotzijde reageert enkel op de eerste synchronisatiepuls. Het is dus belangrijk dat de camera de coördinaten van het voorwerp doorstuurt die horen bij het eerste beeld waarin het voorwerp is herkend. Om deze reden houdt het visiesysteem de tijd tussen twee herkende voorwerpen bij. Indien deze tijd minder dan twee seconden bedraagt moet het visiesysteem geen parameters naar de robot verzenden. Als de tijd tussen de twee gevonden voorwerpen wel hoger is als twee seconden moet het visiesysteem de coördinaten wel verzenden naar de robot. Op deze manier verstuurt het visiesysteem niet opnieuw coördinaten van het voorwerp op een verder gelegen plaats op de transportband. Na deze stap is het verwerken van de genomen afbeelding afgerond en herhaalt het proces zich.

---

<sup>6</sup> Dit is vooral zo bij relatief lage bandsnelheden.





**Figuur 41: Flowchart voor het zoeken van voorwerpen op de transportband**

Het aanleren van het model kan met behulp van dit visiesysteem op drie verschillende manieren. Een eerste mogelijkheid is om het echte voorwerp te gebruiken zoals reeds is besproken bij de Cognex-camera. In dit geval moet de camera het programma voor het aanmaken van het object gebruiken zoals eerder besproken.

Een tweede mogelijkheid is om een foto van het voorwerp in JPG-formaat te gebruiken om het model aan te leren. In plaats dat de camera een beeld van het voorwerp maakt laadt de software een beeld in JPG-formaat in. Deze procedure is uitgevoerd in een extra applicatie. De *teach-in* methode m.b.v. JPG-bestanden is gelijkaardig aan deze bij het Cognex-systeem. Het visiesysteem zoekt naar geschaalde voorwerpen omdat de foto nooit vanaf exact dezelfde hoogte is genomen. Het programma vraagt de minimum en maximum schaal van het voorwerp op. De grenswaarden voor de schaling zijn op voorhand nauwkeurig bepaald. Hoe groter het verschil is tussen de grenswaarden, hoe groter de verwerkingstijd van het visiesysteem. Voor elke schaal tussen deze grenswaarden genereert het systeem een model. Hoe groter het verschil, hoe meer modellen er zijn aangemaakt. Het programma overloopt al deze modellen waardoor de verwerkingstijd stijgt. Het verschil tussen de minimum- en maximumgrens dient dus zo klein mogelijk te zijn om onnodig hoge verwerkingstijden te vermijden.

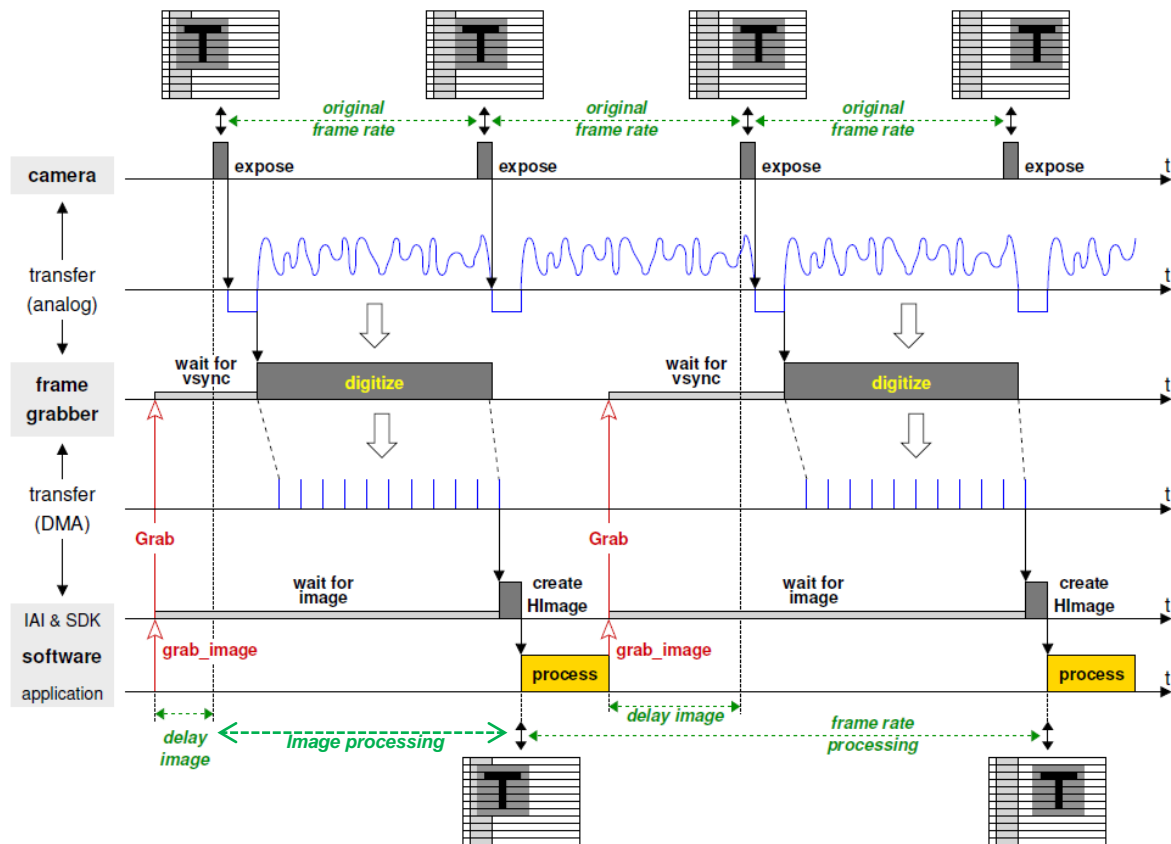
Bij het nemen van de afbeelding in JPG-formaat is het belangrijk dat dit onder dezelfde hoek gebeurt als de camera is opgesteld. In het geval van het naamkaartje in de testopstelling zal de verhouding breedte op hoogte niet meer gelijk zijn. Dit heeft als gevolg dat het visiesysteem het te zoeken voorwerp niet meer herkent aan de hand van het aangeleerde model. Aangezien de contouren van het JPG-bestand nooit exact hetzelfde zijn als de contouren van het naamkaartje op de band door belichting, kreuken in het kaartje en de hoek waaronder afbeelding is genomen moet het zoekcommando zodanig zijn ingesteld dat er vervorming van het model mag optreden t.o.v. het te zoeken voorwerp. De parameter die de toegelaten vervorming aangeeft noemt *max deformation*. Deze parameter staat bij de andere zoekopdrachten altijd op nul, wat het nauwkeurigste resultaat oplevert. Maar aangezien hier wel vervorming optreedt mag deze parameter niet langer gelijk aan nul zijn. De nieuwe waarde in de testopstelling is *max deformation 2*, wat betekent dat de contouren van het gevonden object met twee pixels mogen verschillen t.o.v. de contouren van het model.

Een derde manier om het model aan te leren is door gebruik te maken van DXF-bestanden. De eerste stap is het maken van een DXF-*file* van het te zoeken voorwerp. Het is belangrijk dat het voorwerp in de *file* gelijkvormig is aan het te zoeken voorwerp. Daarnaast kan Halcon geen tekst-entiteiten herkennen binnen een DXF-*file*<sup>7</sup>. Halcon haalt de contouren voor het model rechtstreeks uit het DXF-bestand. De schaalfactor is hier van belang. De camera-kalibratie bepaalt immers de hoogte en breedte van het naamkaartje. Door de handmatige camera-kalibratie speelt er steeds een meetfout mee. De afmetingen van het voorwerp binnen de DXF-*file* zijn nooit perfect gelijk aan deze afmetingen van het voorwerp binnen een beeld. Een schatting van de minimum en maximum grenswaarden voor de schaling is tevens vereist. Gelijkaardig aan de *teach-in* mogelijkheid via JPG-formaat is het schaalbereik zo klein mogelijk om onnodig hoge rekentijden te vermijden.

---

<sup>7</sup> Dit probleem is op te lossen door elk symbool apart te tekenen in de CAD-*file*.

Het visiesysteem kan op verschillende manieren beelden nemen. Binnen de thesis opstelling is geopteerd voor *synchronous image grabbing*. Dit is de meest eenvoudige methode en is geïllustreerd aan de hand van figuur 42.



**Figuur 42: Beelden nemen op de standaard manier [18]**

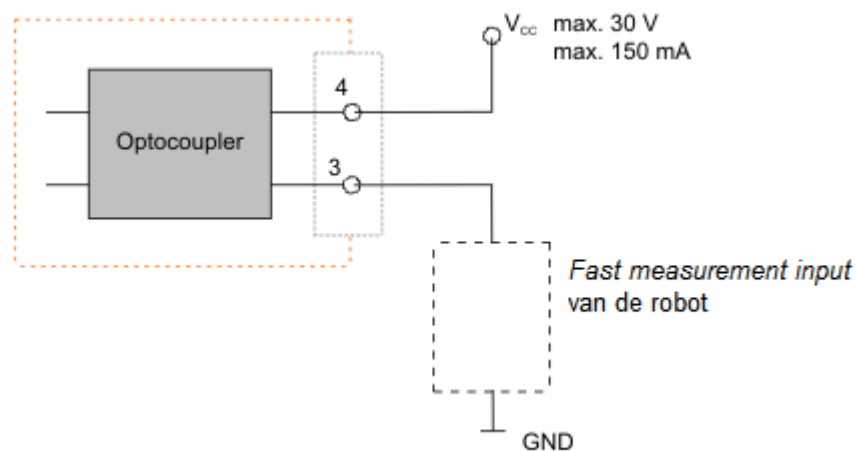
De werking van dergelijk visiesysteem bestaat uit drie lagen. De eerste laag is de camera-laag die de optische chip aan- en uitschakelt. Dit aan- en uitschakelen is verwezenlijkt door een *exposure*- of belichtingspuls. De instelbare breedte van deze puls stelt de sluitertijd van de camera voor. De camera werkt in *freerun-mode*, in deze mode neemt hij om een vaste tijd een beeld. Deze vaste tijd is benoemd als *frame rate*. De grootte van een *frame rate* ligt vast door het type camera en is voor dit toestel gelijk aan 40 fps of 25 ms. De tweede laag is de framegrabber-laag die analoge signalen omvormt naar gedigitaliseerde beelden met behulp van een ADC. De overgang van de framegrabber-laag naar de software-laag gaat via het *Direct Access Memory (DMA)*, dit is het tijdelijk geheugen waarin de gedigitaliseerde beelden zijn opgeslagen. Ten slotte is er de software-laag die de Halcon-applicaties bevat.

Bij *synchronous framegrabbing* geeft Halcon binnen de software-laag het commando *grab\_image*. Halcon kan dit commando op elk moment aanspreken. De tijd tussen dit commando en de eerst volgende belichtingspuls binnen de camera-laag varieert. Deze variërende tijd noemt de *delay image*. De *delay image* is in het ideale geval gelijk aan nul en in het meest nadelige geval een *frame rate*. De tijd om vervolgens het genomen beeld te digitaliseren, in het geheugen te plaatsen en in te lezen in de software-laag is altijd een vaste waarde. Deze tijd noemt de *image processing time*. Vanaf dit punt gaat Halcon het opgeslagen beeld vanuit het DMA verwerking in zijn applicatie. De tijd die Halcon meegeeft voor het commando *grab\_image* bestaat uit de *delay image time* samen met de *image processing time* [18]. Aangezien de *delay image time* varieert zorgt

deze tijd voor ongewenste spreiding. Het Halcon programma moet enkel de *image processing time* in rekening brengen. De applicatie kan deze tijd echter niet apart berekenen. Hierdoor is de tijd voor het commando *grab-image* niet opgenomen in de tijdsmeting. Dit tekort aan tijd is beschouwd als een 'dode tijd'. De oplossing voor dit probleem staat beschreven in paragraaf 6.1.

### 5.2.3 I/O-configuratie

Uit tabel 8 volgt dat de uEye-camera beschikt over één input en één output. De input is ongebruikt en de output dient om de synchronisatiepuls te geven aan ConveyorTech. De output kan op twee manieren verbonden zijn zodat het visiesysteem triggert met een 0 VDC puls (open collector type) of met een 24 VDC puls (open emitter type). De robot leest in op een positieve puls waardoor het open emitter type gehanteerd is. Dergelijke aansluiting is weergegeven in figuur 43.

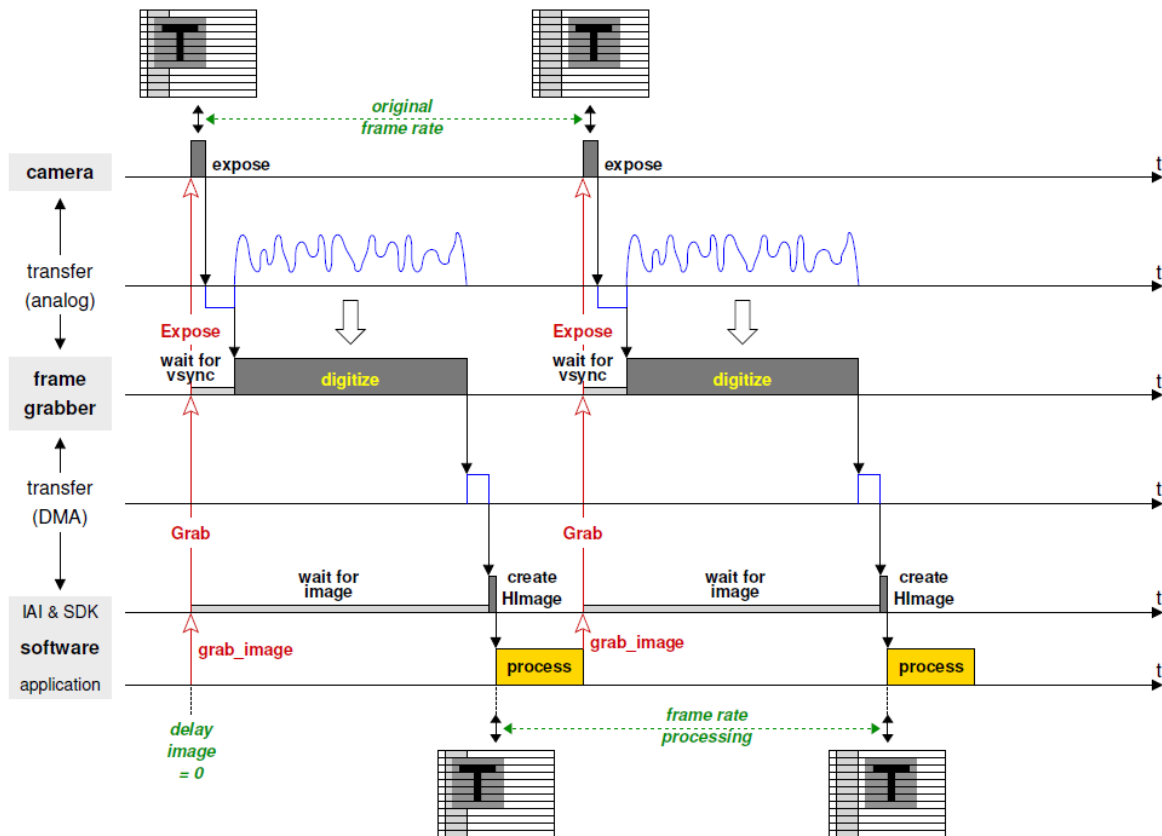


*Figuur 43: Open emitter type aansluiting van de I/O-aansluiting [19]*

## 5.3 Verbetermogelijkheden

### 5.3.1 Image acquisition

De huidige methode voor image acquisition is *synchronous image grabbing*. Het nadeel van deze methode is de aanwezigheid van een *delay image time* zoals beschreven in paragraaf 5.2.2. Er bestaat een methode waarbij deze tijd gelijk is aan nul. Deze methode bestaat uit een asynchrone reset. Niet alle camera's, zoals de gehanteerde camera in de thesis, ondersteunen dergelijke reset. Het principe is geïllustreerd aan de hand van figuur 44.



Figuur 44: Image acquisition met asynchrone reset [20]

### 5.3.2 Externe triggermogelijkheid

De continue trigger heeft de nadelen zoals eerder besproken in paragraaf 4.4.2. Aangezien de camera voorzien is van een *input* bestaat hier de mogelijkheid om externe triggering toe te passen met behulp van een resolver en de nodige elektronica. De voordelen zijn gelijkaardig als bij het Cognex-visiesysteem.

### 5.3.3 Robot - camera kalibratie

In de huidige methode is de camera loodrecht boven de transportband geplaatst met een beeld evenwijdig aan de bewegende band. Vervolgens is de grootte van het beeld in wereldcoördinaten opgemeten om zo het aantal mm/pixel te bepalen. De nadelen van deze methode zijn de onnauwkeurigheid van de kalibratie. Het opmeten van de grootte van het beeld gebeurt met een meetlat die slechts nauwkeurig is tot 0,5 mm. Daarnaast houdt deze kalibratiemethode geen rekening met lensonnauwkeurigheden, bijvoorbeeld distorsie.

Een verbetering bestaat erin de camera te kalibreren met een kalibratieplaat. Halcon voorziet een mogelijkheid om de camera te kalibreren met behulp van een kalibratieplaat en een standaard *calibration assistant* programma. Deze kalibratieplaat met gekende afmetingen is op de robotflens bevestigd en bevindt zich onder de camera. Door dan tien tot twintig beelden te nemen onder verschillende posities en oriëntaties kan Halcon de camera- en lensparameters afleiden. Na het kalibreren is het systeem in staat nauwkeurige metingen uit te voeren waarbij de

lensafwijkingen gecorrigeerd zijn. Daarnaast kan het visiesysteem beeldcoördinaten omzetten naar wereldcoördinaten en de relatie tussen de camera en de robot bepalen.

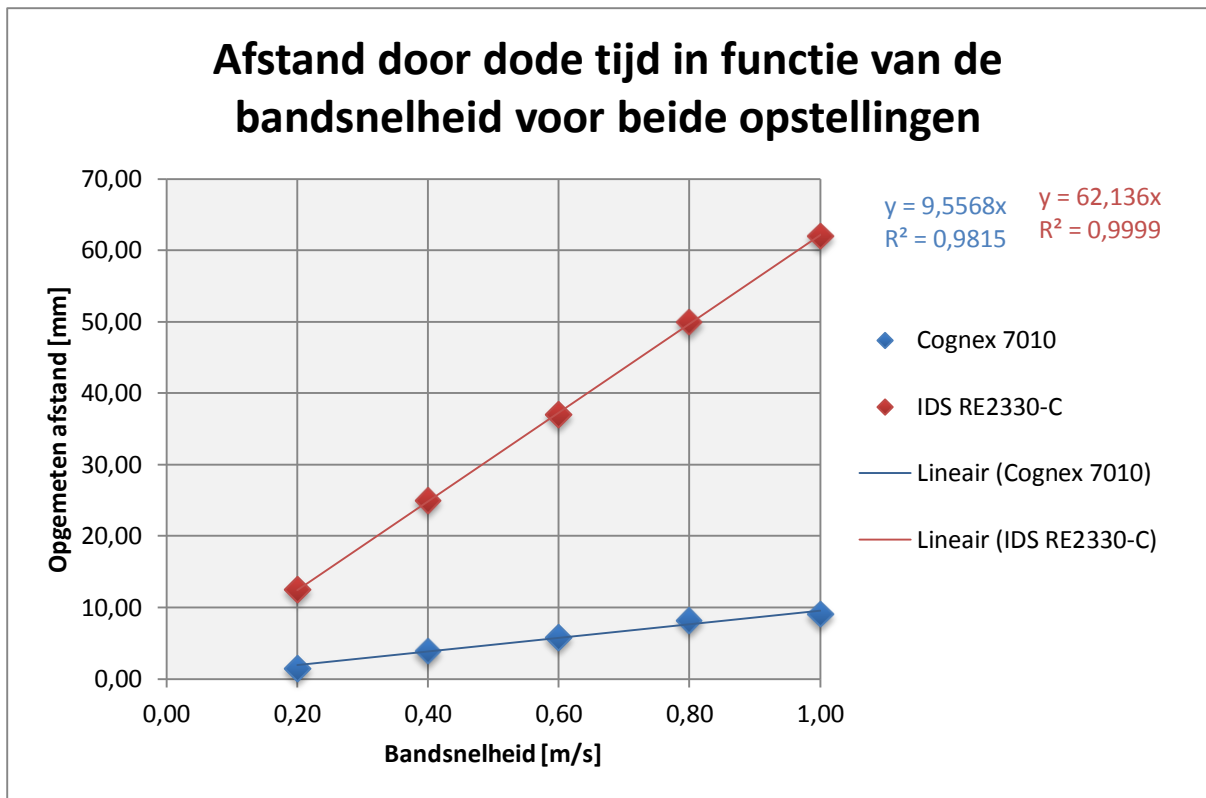
# 6 Metingen

## 6.1 Nauwkeurigheid

De meetprocedure dient om de nauwkeurigheid van het totale systeem te bepalen en is voor elke opstelling identiek. Voor het meten van de nauwkeurigheid is er een verende pen op de robot gemonteerd. Ter hoogte van het voorwerp ligt een horizontale referentielijn. De camera zal de x-coördinaat van deze lijn doorsturen naar de robot. Op basis van deze coördinaat tekent de robot een horizontale lijn. De verplaatsing van het voorwerp over de y-as is niet van belang tijdens de metingen. De nauwkeurigheid is immers veel groter in de y-richting aangezien de verwerkingstijd van het visiesysteem hier geen rol speelt.

In paragraaf 3.3.2 is reeds aangegeven dat de robot de verwerkingstijd van de camera in rekening brengt door deze tijd te vermenigvuldigen met de bandsnelheid. Het is dus vereist dat de totale vereiste rekestijd per beeld samen met de coördinaten zijn verstuurd. De Cognex-camera kan de volledige tijd per beeld niet doorgeven maar enkel de tijd die nodig is voor de beeldverwerking. Hierdoor blijft de robot nog een bepaalde afstand achterlopen op de referentielijn. De extra tijd, die de camera niet kan doorsturen, bestaat uit de *latency time* en de *acquisition time* zoals besproken in paragraaf 4.3.2 voor de Cognex-camera. Het IDS-systeem heeft een gelijkaardig probleem. Halcon kan enkel de tijd voor totale *image acquisition* berekenen. Deze bestaat uit een *image processing time* en een *delay image time*, zie paragraaf 5.2.2. De *image processing time* is echter enkel gevraagd maar deze tijd kan de software niet berekenen. De onbekende tijden voor het Cognex-systeem en het Halcon-systeem zijn op een andere manier bepaald.

De robot tekent lijnen zonder deze tijden in rekening te brengen voor beide systemen. Door het ontbreken van deze tijd ligt de getekende lijn steeds achter de referentielijn. Het verschil in afstand tussen de door de robot getekende lijn en de referentielijn is een maat voor de onbekende tijd. Om een representatieve waarde voor deze tijd te bekomen tekent de robot twintig lijnen per proef. Daarnaast zijn er meerdere proeven uitgevoerd voor verschillende bandsnelheden. Vervolgens is elke afstand uitgezet in functie van de bandsnelheid. Dit resulteert in een puntenwolk voor beide opstellingen. Lineaire regressie toegepast op beide puntenwolken geeft voor elk systeem de bij te regelen bandafstand in functie van de bandsnelheid. Dit is geïllustreerd in figuur 45. Deze vergelijkingen zijn ten slotte geïmplementeerd in het robotprogramma.



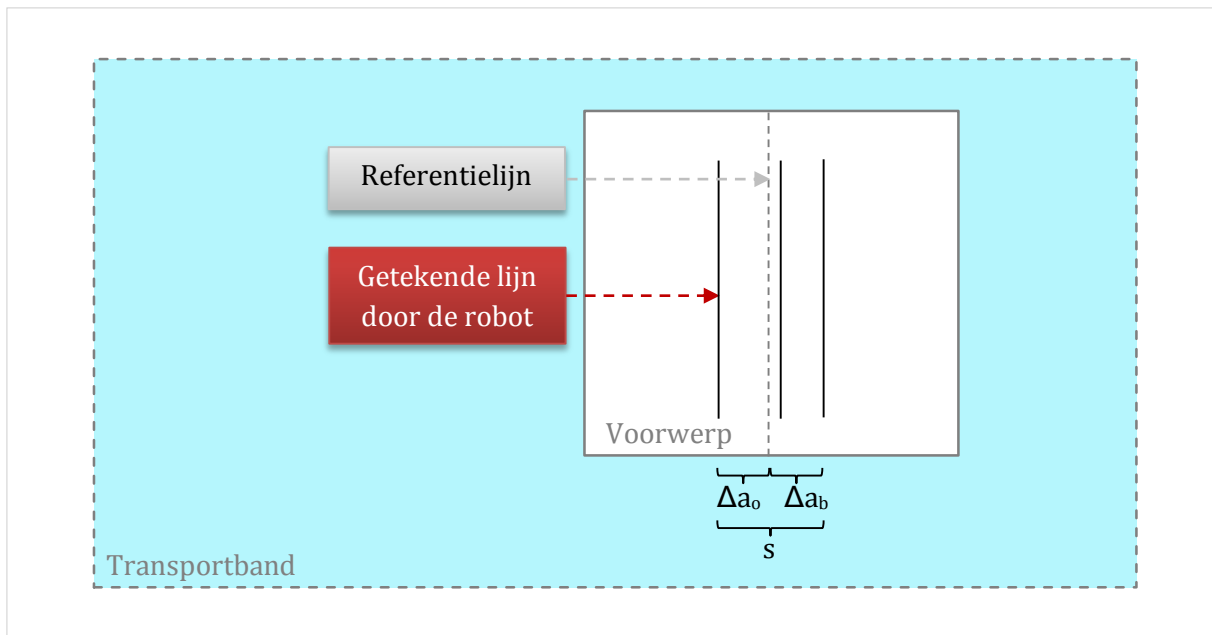
**Figuur 45: Resultaten voor het bepalen van de onbekende bij te regelen bandafstand**

Nu bepaalt de robot de juiste coördinaten van de referentielijn. In theorie tekent de robot nu steeds een lijn ter hoogte van de referentielijn. In werkelijkheid ontstaat er een spreiding in de buurt van deze referentielijn. Deze spreiding is een maat voor de nauwkeurigheid van de gehele opstelling. Om een representatieve meting te verkrijgen is de proef minimum twintig keer herhaald. De oorzaak van deze spreiding kan liggen aan verschillende factoren:

- nauwkeurigheid van het *tool center point*,
- nauwkeurigheid van ConveyorTech,
- dikte van de stift,
- nauwkeurigheid van het visiesysteem,
- nauwkeurigheid van de offset waarmee het blad is gehangen (zie later),
- nauwkeurigheid waarmee de referentielijn manueel is getekend.

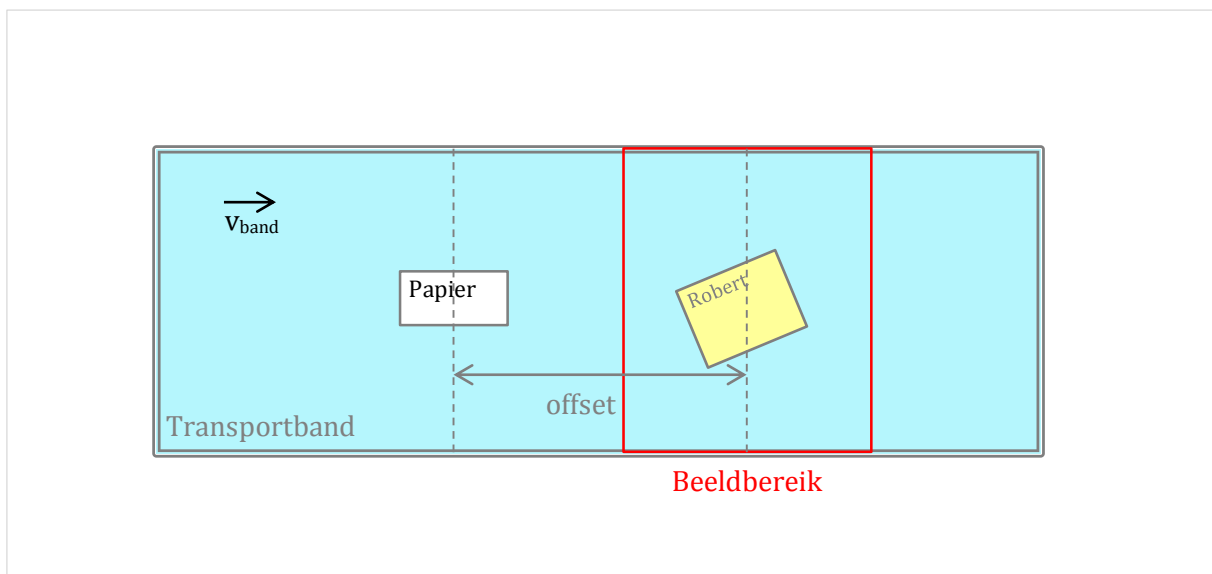
Een mogelijk meetresultaat met drie lijnen is weergegeven in figuur 46. De op te meten parameters zijn de spreiding  $s$  van het lijnenstel, de grootste afwijking onder de referentielijn  $\Delta_{a_0}$  en de grootste afwijking boven de referentielijn  $\Delta_{a_b}$ .





**Figuur 46: Voorbeeld van een meetresultaat**

Het stuk papier waarop de robot lijnen tekent mag niet in hetzelfde beeld vallen als het voorwerp. Omdat het papier wit gekleurd is, vergroot de aanwezigheid van het papier de vereiste rekentijd onnodig. Om deze fout te elimineren is het blad papier met een *offset* van minstens de hoogte van het beeld naar onder verschoven. Dit is te zien in figuur 47.



**Figuur 47: Schematische voorstelling van de meetprocedure**

Voor de twee visiesystemen zijn er vijf reeksen van metingen om de nauwkeurigheid te bepalen. Elke reeks is uitgevoerd voor de minimum bandsnelheid van 0,1 m/s en de maximale bandsnelheid die de opstelling kan verwerken. De maximum bandsnelheid mag nooit hoger dan 1 m/s zijn ook al kan het visiesysteem hogere bandsnelheden verwerken. Bij een hogere bandsnelheid dan 1 m/s overschrijdt de robot namelijk zijn werkgebied.

De maximum bandsnelheid die het systeem kan verwerken is bepaald door drie oorzaken:

- (1) De maximum snelheid van de transportband;
- (2) De maximum snelheid van de robot;
- (3) De maximum benodigde verwerkingstijd van het visiesysteem voor het detecteren van voorwerpen.

Binnen de eerste reeks metingen ligt er steeds één voorwerp onder een vaste hoek. Binnen de tweede reeks ligt er steeds één voorwerp verdraaid onder een variabele hoek. De metingen zijn uitgevoerd voor variabel hoekbereik ( $-\theta; \theta$ ). Om er zeker van te zijn dat de camera het object altijd vindt is een marge van  $5^\circ$  op de maximumhoekverdraaiing van het voorwerp ingesteld. Bij de derde reeks metingen is er getest op twee verschillende voorwerpen die beide altijd onder een vaste hoek op de transportband liggen. Het visiesysteem zoekt hier steeds naar twee voorwerpen terwijl het slechts de locatie van één voorwerp naar de robot verzendt. De robot zal binnen deze meting ook slechts voor één voorwerp lijnen tekenen. De verwerkingstijd bestaat echter uit de tijd voor het zoeken naar twee voorwerpen. De vierde reeks van metingen bestaat uit *teach-in* via JPG-formaat om voorwerp te detecteren. Deze voorwerpen liggen onder een vaste hoek op de transportband. De vijfde en laatste reeks bestaat uit *teach-in* van het voorwerp via DXF-formaat. De voorwerpen liggen binnen deze laatste reeks ook onder een vaste hoek op de transportband.

De meetresultaten voor alle reeksen zijn gelijkaardig aan Figuur 47. De opgemeten parameters zijn de spreiding  $s$ , de afwijking boven de referentielijn  $\Delta a_b$  en de afwijking onder de referentielijn  $\Delta a_o$ . De maximale nauwkeurigheid bij het opmeten van de resultaten bedraagt 0,5 mm. De resultaten van beide opstellingen zijn onderling per type meting vergeleken.

### 6.1.1 Meetresultaten voorwerpen onder een vaste hoek

Bij dit type meting liggen alle voorwerpen onder een vaste hoek van  $0^\circ$  ( $\pm 5^\circ$  marge). De eerste stap bij deze meting is een berekening van de grootst haalbare bandsnelheid voor beide systemen zoals beschreven in paragraaf 6.2.1. Het resultaat voor de afwijking boven  $\Delta a_b$ , de afwijking onder  $\Delta a_o$  en de spreiding  $s$  staan weergegeven in tabel 9.

Tabel 9: Resultaten voor niet verdraaide voorwerpen

$v_{band}$ [m/s]	Cognex7010			IDS uEye-2230RE-C		
	$s$ [mm]	$\Delta a_b$ [mm]	$\Delta a_o$ [mm]	$s$ [mm]	$\Delta a_b$ [mm]	$\Delta a_o$ [mm]
0,1	1,5	1,5	0	1	0	-1
0,2	1,5	1	-0,5	0,5	0,5	0
0,3	1	0,5	-0,5	1	0,5	-0,5
0,4	1	0,5	-0,5	0,5	1	0,5
0,5	1,5	0,5	-1	1	1	0
0,6	1	0	1	1	0,5	-0,5
0,7	2	1	-1	1	1,5	0,5
0,8	2	1	-1	1,5	1,5	0
0,9	1,5	0	-1,5	1,5	2	0,5
1	2	1	-1	1	2	1

Beide visiesystemen kunnen alle voorwerpen detecteren tegen de maximum bandsnelheid van 1 m/s. In de opstelling met het visiesysteem van Cognex zijn volgende

conclusies te trekken. De spreiding varieert t.g.v. de bandsnelheid. Bij lage bandsnelheden (0,1 – 0,5 m/s) bedraagt de spreiding 1 tot 1,5 mm. Bij hoge bandsnelheden (0,5 – 1,0 m/s) bedraagt de spreiding 1,5 tot 2 mm. De maximale afwijking t.o.v. de referentielijn varieert van 0 tot 1,5 mm en dit zowel boven als onder de referentielijn. De afwijking hangt niet af van de bandsnelheid. In de tweede opstelling met het visiesysteem van IDS varieert de spreiding tussen 0,5 en 1,5 mm. De maximale afwijking varieert tussen 0 en 2 mm. Zowel de spreiding als de afwijking zijn afhankelijk van de bandsnelheid. Beide parameters vergroten naargelang de bandsnelheid verhoogt. Daarnaast ligt het lijnenpaar bij hoge bandsnelheden steeds boven de referentielijn.

Qua spreiding scoort de eerste opstelling minder goed als de tweede. Bij nagenoeg elke bandsnelheid is de spreiding binnen de opstelling met het IDS-systeem lager. Qua maximale afwijking scoort de eerste opstelling net iets beter. Een belangrijke vaststelling is dat de afwijking niet afhankelijk is van de bandsnelheid. In de tweede opstelling is dit echter wel zo. Beide systemen blijken echter geschikt voor het herkennen van voorwerpen onder een vaste hoek voor bandsnelheden tot 1 m/s.

### 6.1.2 Meetresultaten voorwerpen onder een willekeurige hoek

Bij dit type meting liggen voorwerpen onder een willekeurige hoek  $\theta$ , waarbij deze hoek ligt tussen twee grenswaarden  $-\theta$  en  $\theta$ . De metingen zijn uitgevoerd voor de minimum en maximum haalbare bandsnelheid voor de hoeken  $-\theta$  en  $\theta$  uit tabel 18. De geteste hoeken  $\theta$  zijn afhankelijk van het visiesysteem. Voor de opstelling met het Cognex-visiesysteem bedragen de hoeken  $10^\circ$ ,  $20^\circ$ ,  $40^\circ$  en  $60^\circ$ . Voor de opstelling met het IDS systeem zijn de testen voor hoeken van  $10^\circ$ ,  $40^\circ$  en  $180^\circ$  uitgevoerd. De minimum bandsnelheid is steeds gelijk aan 0,1 m/s. De maximumbandsnelheid is de berekende maximumbandsnelheid naar onder afgerond tot op een tiende voor het Cognex-systeem en steeds 1 m/s voor het IDS-systeem, zie paragraaf 6.2.2. Het resultaat voor de afwijking onder  $\Delta a_o$ , de afwijking boven  $\Delta a_b$ , de procentuele fout  $a_{fout}$  en de spreiding  $s$  staan weergegeven in tabel 10 voor de eerste opstelling en tabel 11 voor de tweede opstelling.

Tabel 10: Resultaten voor voorwerpen onder een variabele hoek voor de opstelling met het Cognex-systeem

$\theta$ [°]	$v_{band}$ [m/s]	Cognex7010		
		$s$ [mm]	$\Delta a_b$ [mm]	$\Delta a_o$ [mm]
10	0,1	1	1	0
	0,8	1	0	-1
20	0,1	1,5	1,5	0
	0,5	3,5	1,5	-2
40	0,1	2	1	-1
	0,3	2	1	-1
60	0,1	2	2	0
	0,2	2,5	2,5	0

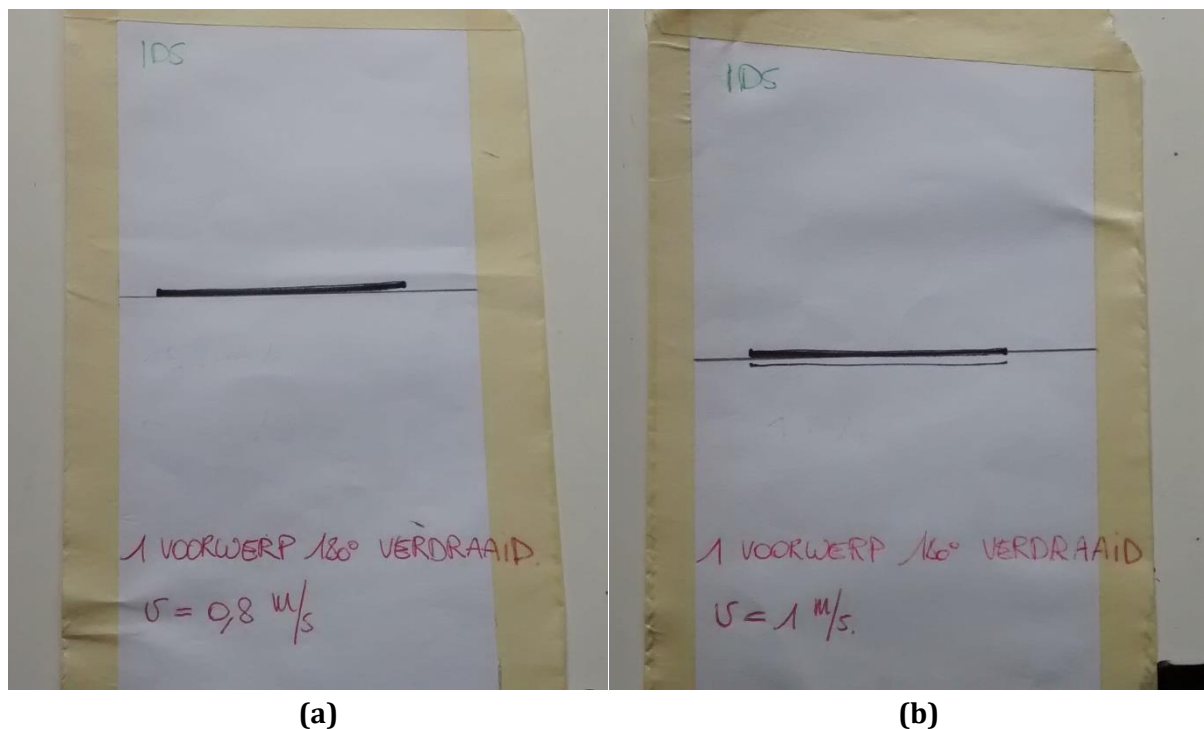
De opgemeten spreiding voor het Cognex-visiesysteem varieert tussen 1mm en 3,5 mm. De maximale afwijking bedraagt 2,5 mm. De spreiding stijgt bij toenemende bandsnelheden en de afwijking verandert niet in functie van de bandsnelheid. Dit komt overeen met de resultaten uit paragraaf 6.1.1. Het visiesysteem legt door zijn hoge verwerkingstijden echter een grote

bepanking op de maximale bandsnelheid  $v_{\text{band}}$  en het hoekbereik  $\theta$ . Hierdoor is het systeem enkel geschikt voor een relatief klein hoekbereik tegen lage bandsnelheden.

**Tabel 11: Resultaten voor voorwerpen onder een variabele hoek voor de opstelling met het IDS-systeem**

$\theta$ [°]	$v_{\text{band}}$ [m/s]	IDS uEye RE2230-C		
		s [mm]	$\Delta a_b$ [mm]	$\Delta a_o$ [mm]
10	0,1	1	0,5	-0,5
	1	1,5	1,5	1
40	0,1	0,5	0,5	0
	1	2	1	2
180	0,1	1	0,5	-0,5
	0,2	0,5	0	-0,5
	0,4	3,5	0,5	-3
	0,6	0,5	0	0,5
	0,8	2	2	0
	1	4	2	-2

Voor de opstelling met het IDS-systeem zijn de resultaten als volgt: de spreiding bedraagt hier minimaal 0,5 mm en maximaal 3 mm. De afwijking varieert tussen 0,5 mm en 4 mm. Voor twee metingen zijn er enkele lijnen een stuk verwijderd van het lijnenpaar getekend. De desbetreffende metingen zijn  $[\theta = 180^\circ; v_{\text{band}} = 0,4 \text{ m/s}]$  en  $[\theta = 180^\circ; v_{\text{band}} = 1,0 \text{ m/s}]$ . Twee meetresultaten zijn weergegeven in Figuur 48. Hierin is figuur 48(a) meting  $[\theta = 180^\circ; v_{\text{band}} = 1,0 \text{ m/s}]$ , zonder dit verschijnsel en figuur 48(b) meting  $[\theta = 180^\circ; v_{\text{band}} = 0,8 \text{ m/s}]$ , met dit verschijnsel.

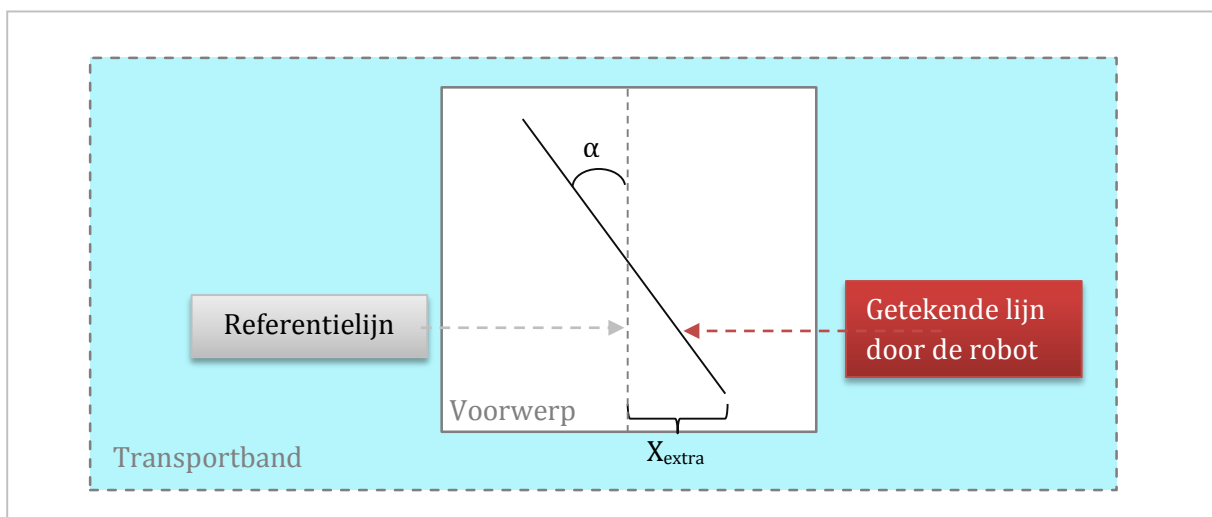


**Figuur 48: Meetresultaat (a) normaal (b) met apart getekende lijn**

Aangezien de robot dit verschijnsel vertoont binnen meerdere metingen is dit geen uitschieter. De robot tekent deze lijn altijd op een bepaalde afstand achter het lijnenpaar. De spatie tussen deze lijn en het lijnenpaar is nooit gevuld met andere lijnen. Dergelijke 'foutieve' lijnen verlagen de nauwkeurigheid van het systeem aanzienlijk. Het systeem tekent deze lijnen bij elke bandsnelheid. De spatie tussen deze lijn en het lijnenpaar is nooit gevuld. Uit de metingen blijkt dat de spatie tussen de afgezonderde lijn en het lijnenpaar is gerelateerd aan de bandsnelheid. De afstand vergroot immers bij toenemende snelheid. Indien er voor de metingen meer dan twintig lijnen getekend zouden zijn, zullen er naar verwachting één of meerdere afwijkende lijnen bijzitten.

Algemeen valt te besluiten dat het visiesysteem van Cognex niet bruikbaar is voor het volgen van delen onder een variabele hoek. Het systeem kan enkel werken bij een zeer laag hoekbereik en dit tegen zeer lage bandsnelheden. Het systeem van IDS daarentegen verleent zich wel tot deze toepassing. De reden hiervoor ligt weer in de opbouw van de hardware in beide systemen. Bij hoge snelheden zal binnen de tweede opstelling de nauwkeurigheid wel lager liggen door de afwijkende lijnen.

Naast de nauwkeurigheid op de x-coördinaat is ook de nauwkeurigheid op de hoekverdraaiing opgemeten. De meetprocedure is hiervoor gewijzigd. Naast een x-coördinaat stuurt het visiesysteem ook de hoek  $\alpha$  door waaronder het voorwerp op de transportband ligt. In plaats van horizontale lijnen tekent de robot de lijnen nu onder deze hoek  $\alpha$ . Het beginpunt van de te tekenen lijn ligt op de afstand (+) $X_{extra}$  van de referentielijn. Deze afstand komt voort uit de tangens van de hoek  $\alpha$ . Het eindpunt is analoog berekend aan het beginpunt. De zin van de hoek  $\alpha$  is positief tegen de richting van de klok in. Een schematische weergave van deze meetprocedure is afgebeeld in figuur 49.



**Figuur 49: Schematische weergave hoeknauwkeurigheid**

Voor het Cognex-visiesysteem is deze meting niet uitgevoerd aangezien dit visiesysteem niet geschikt is voor het herkennen van voorwerpen onder een variabele hoek zoals net aangetoond. Voor het visiesysteem met de uEye-camera en de Halcon software is deze hoekverdraaiing wel opgemeten voor de bandsnelheden 0,1 m/s en 1 m/s. De opgemeten hoeknauwkeurigheid bedraagt voor beide bandsnelheden  $1^\circ$ .

### 6.1.3 Meetresultaten meerdere voorwerpen

Bij dit type meting liggen er twee verschillende voorwerpen op de transportband. Het visiesysteem *teached* beide voorwerpen in. De visiesystemen zijn dus voortdurend op zoek naar twee voorwerpen. Beide voorwerpen zijn geplaatst onder een vaste hoek om de complexiteit te verlagen. De eerste stap is het bepalen van de maximum bandsnelheid voor beide systemen, zoals beschreven in paragraaf 6.2.3.

Voor het Cognex-systeem is de rekentijd zodanig groot dat de maximum bandsnelheid beperkt blijft tot 0,6 m/s. Deze resultaten zijn terug te vinden in tabel 12. De tabel bevat de afwijking boven  $\Delta a_b$ , de afwijking onder  $\Delta a_o$ , de procentuele fout  $a_{fout}$  en de spreiding  $s$  voor beide opstellingen.

**Tabel 12: Resultaten voor twee voorwerpen onder een vaste hoek**

Cognex7010			
$v_{band}$ [m/s]	$s$ [mm]	$\Delta a_b$ [mm]	$\Delta a_o$ [mm]
0,1	1	0,5	0,5
0,6	2	1	1

Het systeem van Cognex heeft een spreiding van 1 mm en 2 mm voor de respectievelijke bandsnelheden van 0,1 m/s en 0,6 m/s. De maximale afwijking is nagenoeg gelijk voor elke meting. Zoals eerder opgemeten stijgt de spreiding hier ook in functie van de bandsnelheid terwijl de afwijking nagenoeg gelijk blijft.

Het IDS-systeem kan vlot werken op de maximum bandsnelheid van 1 m/s. De metingen voor dit systeem zijn uitgevoerd voor de volgende bandsnelheden: 0,1 m/s, 0,5 m/s, 0,8 m/s en 1,0 m/s. De resultaten voor deze metingen zijn weergegeven in tabel 13.

**Tabel 13: resultaten voor twee voorwerpen onder een vaste hoek voor het IDS-systeem**

IDS uEye RE2230-C			
$v_{band}$ [m/s]	$s$ [mm]	$\Delta a_b$ [mm]	$\Delta a_o$ [mm]
0,1	1	1	0
0,5	2,5	1	-1,5
0,8	4	2	-2
1	3	3	0

Bij de meetresultaten voor de opstelling met het systeem van IDS is de spreiding minimaal 1 mm en maximaal 4 mm. De hoge waarden voor de spreiding van 3 en 4 mm zijn wederom veroorzaakt door het verschijnsel uit figuur 48. De robot heeft telkens een enkele lijn op een bepaalde afstand onder het lijnenpaar getekend. De maximale positieve afwijking boven de referentielijn bedraagt 3 mm. De maximale onderste afwijking bedraagt (-)2 mm. Dit komt door de afwijkende lijn. De nauwkeurigheid van de opstelling neemt af door deze afwijkingen.

Aangezien de voorwerpen niet verdraaid zijn is het mogelijk voor de eerste opstelling met het Cognex-visiesysteem om naar twee voorwerpen te zoeken. De Cognex-camera is bruikbaar tot bandsnelheden van 0,6 m/s. Het systeem van IDS is bruikbaar voor alle bandsnelheden. De nauwkeurigheid is bij hoge bandsnelheden wel lager door de optredende afwijkingen. Deze testen

zijn uit te breiden naar het zoeken van twee voorwerpen waarvan beide voorwerpen onder een variabele hoek liggen. Maar omdat de rekentijden al sterk vergroten bij het Cognex-systeem bij één enkel verdraaid voorwerp gaat deze toepassing te zwaar zijn voor dit systeem. Om deze reden zijn meerdere testen met twee voorwerpen niet uitgevoerd.

#### 6.1.4 Meetresultaten JPG-formaat als model

Het herkennen van voorwerpen op basis van een JPG-formaat is door beide visiesystemen ondersteund. Het model in JPG-formaat is een foto van het voorwerp gemaakt door een extern fototoestel. Het is echter wel belangrijk dat tijdens het nemen van de foto, de camera zo loodrecht mogelijk op het oppervlak van het voorwerp gericht is<sup>8</sup>. Het voorwerp op de transportband ligt steeds onder dezelfde vaste hoek tijdens de meting. Voor het Cognex-visiesysteem bedraagt de maximale bandsnelheid 1 m/s. Voor de opstelling met de Cognex-camera zijn de metingen uitgevoerd voor de laagste en hoogste bandsnelheid. De resultaten voor de spreiding  $s$ , de afwijking boven  $\Delta a_b$  en de afwijking onder  $\Delta a_o$  voor deze opstelling zijn weergegeven in tabel 14.

*Tabel 14: Resultaten voor het zoeken op basis van JPG-formaat met het visiesysteem van Cognex*

$v_{band}$ [m/s]	Cognex7010		
	$s$ [mm]	$\Delta a_b$ [mm]	$\Delta a_o$ [mm]
0,1	1,5	0	-1
1	1,5	-1,5	-3

De spreiding blijft voor beide bandsnelheden gelijk en bedraagt 1,5 mm. Deze meetresultaten spreken voorgaande meetresultaten tegen waar de spreiding steeg bij verhoogde bandsnelheid. De maximale afwijking voor zowel boven als onder stijgt bij toenemende bandsnelheid. Dit spreekt ook voorgaande resultaten tegen, waar deze nagenoeg constant bleven. Een mogelijke reden hiervoor kan zijn dat het model niet meer dezelfde hoogte tot breedte verhouding heeft als het naamkaartje. Indien het visiesysteem het model dan herkent in een afbeelding en dit model plaatst op het herkende naamkaartje, ligt het middelpunt van het model niet op dezelfde hoogte als het middelpunt van de afbeelding. De berekende coördinaten liggen dan iets voor of achter het echte middelpunt van het naamkaartje.

Voor het IDS-visiesysteem is de hoogst haalbare bandsnelheid ook gelijk aan de hoogst mogelijke bandsnelheid van 1 m/s. De metingen zijn uitgevoerd voor bandsnelheden van 0,1 m/s, 0,5 m/s en 1 m/s. De resultaten voor de spreiding  $s$ , de afwijking boven  $\Delta a_b$  en de afwijking onder  $\Delta a_o$  zijn weergegeven in tabel 15.

<sup>8</sup> Dit is een moeilijke opgave omdat een persoon zijn fototoestel nooit perfect loodrecht houdt.

Tabel 15: Resultaten voor het zoeken op basis van JPG-formaat met het visiesysteem van IDS

$v_{\text{band}}$ [m/s]	IDS uEye RE2230-C		
	$s$ [mm]	$\Delta a_b$ [mm]	$\Delta a_o$ [mm]
0,1	1	1,5	0,5
0,5	2	1	-1
1	1	2	1

Binnen de opstelling met het IDS-systeem bedraagt de minimale spreiding 1 mm voor lage bandsnelheid en 2 mm voor hoge bandsnelheid. De hoogste waarde voor de spreiding komt door een afwijkende lijn binnen de meting [ $v_{\text{band}} = 0,5$  m/s]. De maximale afwijking bedraagt 2 mm. Beide visiesystemen ondersteunen de mogelijkheid tot het aanleren met JPG-formaten.

### 6.1.5 Meetresultaten DXF-formaat als model

Het Cognex-visiesysteem ondersteunt patroonherkenning d.m.v. DXF-*files* niet. De beeld verwerkende Halcon-software in het IDS-visiesysteem ondersteunt deze functie wel. Het voorwerp is onder een vaste hoek op de transportband geplaatst. De resultaten van de opstelling met het IDS-systeem voor de spreiding  $s$ , de afwijking boven  $\Delta a_b$  en de afwijking onder  $\Delta a_o$  zijn weergegeven in tabel 16.

Tabel 16: Resultaten voor het zoeken van voorwerpen op basis van DXF-formaat met het visiesysteem van IDS

$v_{\text{band}}$ [m/s]	IDS uEye-2230RE-C		
	$s$ [mm]	$\Delta a_b$ [mm]	$\Delta a_o$ [mm]
0,1	0,5	0,5	0
0,9	7	2	-5
1	4	4	0

De maximale spreiding bedraagt maar liefst 7 mm. Deze hoge waarde voor de spreiding is wederom veroorzaakt door het verschijnsel uit figuur 48. De waarde van deze afwijking is echter groot in vergelijking met de afwijkingen uit vorige metingen. De maximale afwijking bedraagt 5 mm en de oorzaak is ook hier weer deze extra lijn uit figuur 48. Het IDS-visiesysteem is geschikt voor het inlezen van DXF-formaat maar de nauwkeurigheid is bij hoge bandsnelheden relatief laag. Voor lage bandsnelheden is de nauwkeurigheid echter hoog.

## 6.2 Verwerkingssnelheid

De parameter verwerkingssnelheid bepaalt hoe snel voorwerpen achter elkaar op de band kunnen liggen. In paragraaf 1.3 is reeds uitgelegd dat de maximum toegelaten bandsnelheid beperkt is door de benodigde verwerkingstijd van het visiesysteem. Deze bandsnelheid is berekend aan de hand van formule (2).

### 6.2.1 Meetresultaten voorwerpen onder een vaste hoek

Bij dit type meting liggen alle voorwerpen onder een vaste hoek van  $0^\circ$  ( $\pm 5^\circ$  marge). De verwerkingstijden voor voorwerpen onder een vaste hoek zijn terug te vinden in tabel 17. Hierin



is  $M$  de hoogte van het camerabeeld,  $\Phi$  de hoogte van het voorwerp,  $t_{ds}$  de verwerkingstijd van het visiesysteem en  $v$  de maximaal haalbare bandsnelheid zonder een voorwerp te missen.

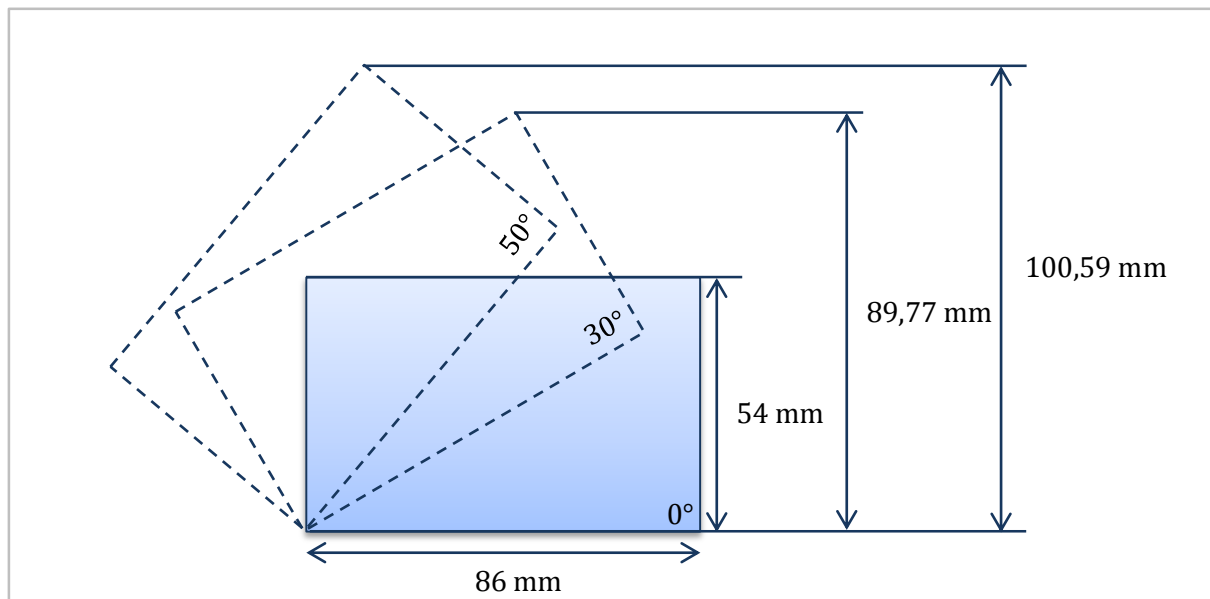
**Tabel 17: Berekening maximale bandsnelheid voor niet verdraaide voorwerpen**

	$M$ [mm]	$\Phi$ [mm]	$t_{ds}$ [ms]	$v$ [m/s]
<b>Cognex</b>	209	54	110	1,41
<b>IDS</b>	435	54	132	2,89

Beide visiesystemen kunnen werken met de maximale bandsnelheid van 1 m/s indien de voorwerpen onder een vaste hoek liggen. Het visiesysteem met de uEye camera en de beeldverwerking op een externe PC heeft een hogere haalbare bandsnelheid ondanks de hogere verwerkingstijd. Dit valt te verklaren doordat het beeld van de uEye camera groter is dan het beeld van de Cognex camera en hierdoor meer bandoppervlakte tegelijk is gescand. Deze redenering blijkt ook uit formule (2), zie paragraaf 1.3.

### 6.2.2 Meetresultaten voorwerpen onder een willekeurige hoek

Indien een voorwerp verdraaid ligt, varieert de hoogte van het voorwerp naar gelang de hoek  $\theta_i$ . Dit is geïllustreerd in figuur 50.



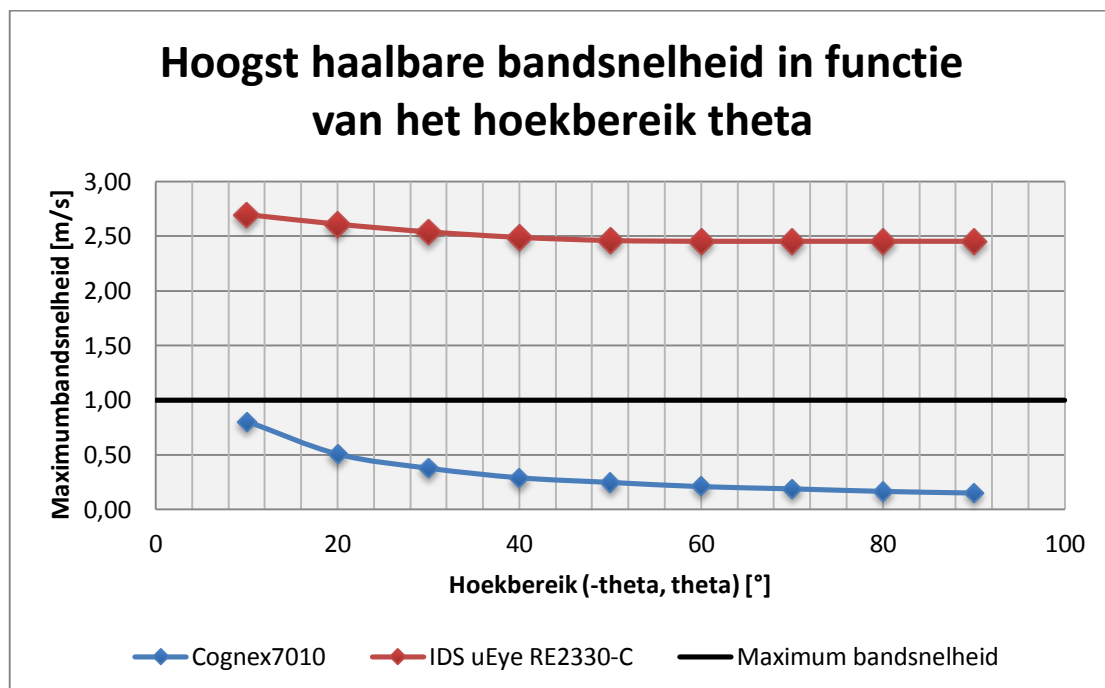
**Figuur 50: Hoogte van het naamkaartje voor verschillende hoeken**

De rekentijd is afhankelijk van de hoogte van het voorwerp en dus ook van de hoek  $\theta_i$  van het voorwerp. In het slechtste geval ligt het voorwerp onder een van de hoeken  $-\theta$  of  $\theta$ . De berekening voor de hoogst haalbare bandsnelheid gebeurt aan de hand van formule (1) uit paragraaf 1.3. De resultaten van deze berekening zijn weergegeven in tabel 18.

Tabel 18: Berekening maximale bandsnelheid voor willekeurig verdraaide voorwerpen

$\theta$ [°]	$\Phi$ [mm]	Cognex7010			IDS uEye-2230RE-C		
		M [mm]	$t_{ds}$ [ms]	$v_{band}$ [m/s]	M [mm]	$t_{ds}$ [ms]	$v_{band}$ [m/s]
10	68,11	209	175	0,81	435	136	2,70
20	80,16	209	255	0,51	435	136	2,61
30	89,77	209	317	0,38	435	136	2,54
40	96,65	209	389	0,29	435	136	2,49
50	100,59	209	440	0,25	435	136	2,46
60	101,55	209	517	0,21	435	136	2,45
70	101,55	209	573	0,19	435	136	2,45
80	101,55	209	655	0,16	435	136	2,45
90	101,55	209	720	0,15	435	136	2,45
180	101,55	209	1323	0,08	435	136	2,45

Het Cognex-visiesysteem heeft bij toenemende hoek  $\theta$  een sterk toenemende rekentijd. Hierdoor verlaagt de maximale bandsnelheid aanzienlijk. De rekentijd neemt bij het IDS-systeem veel minder sterk toe, waardoor de maximum bandsnelheid van 1 m/s steeds binnen het bereik van de opstelling blijft. Figuur 51 geeft voor beide systemen de hoogst haalbare bandsnelheid in functie van het hoekbereik  $-\theta, \theta$  weer.



Figuur 51: Hoogst haalbare bandsnelheid in functie van het hoekbereik voor beide systemen<sup>9</sup>

In beide systemen is er een invers dalend verloop te zien. Het Cognex-systeem kan in geen enkel geval werken onder de maximale bandsnelheid van 1 m/s. Het IDS-visiesysteem daarentegen kan bij elk hoekbereik steeds de maximale bandsnelheid aan. Dit verschil is te wijten aan de opbouw van beide visiesystemen. Het Cognex-systeem is zoals eerder vermeld een smart-

<sup>9</sup> Dit geldt enkel theoretisch zo, de formule houdt geen rekening met eventuele storingen. In de praktijk zal de maximale bandsnelheid net iets lager liggen.

camera. De interne processor is relatief zwak. Het IDS-systeem maakt daarentegen gebruik van een externe krachtige processor. Hierdoor kan het visiesysteem de berekeningen veel sneller uitvoeren.

### 6.2.3 Meetresultaten twee voorwerpen

Bij dit type meting liggen er twee verschillende voorwerpen op de transportband. Het visiesysteem *teached* beide voorwerpen in. De visiesystemen zijn dus voortdurend op zoek naar twee voorwerpen. Hierdoor vergroot de vereiste rekentijd in beide systemen. Beide voorwerpen zijn geplaatst onder een vaste hoek om de complexiteit te verlagen. In tabel 19 zijn de hoogte van het camerabeeld  $M$ , de hoogte van het voorwerp  $\Phi$ , de verwerkingstijd  $t_{ds}$  en de maximaal haalbare bandsnelheid  $v$  weergegeven.

Tabel 19: Berekening maximale bandsnelheid voor twee voorwerpen

	$M$ [mm]	$\Phi$ [mm]	$t_{ds}$ [ms]	$v$ [m/s]
<b>Cognex7010</b>	209	54	246	0,63
<b>IDS uEye-2230RE-C</b>	435	54	142	2,68

Het Cognex-visiesysteem kan de maximale bandsnelheid van 1 m/s niet meer behalen. Het IDS-visiesysteem kan werken met een bandsnelheid van 2,68 m/s. Het verschil in haalbare bandsnelheid valt door twee redenen te verklaren. De eerste reden is het verschil in verwerkingstijd zoals eerder aangegeven. Hierdoor heeft het visiesysteem met de Cognex-camera meer tijd nodig om de contouren van de twee naamkaartjes te zoeken in een genomen afbeelding. De tweede reden is het groter beeld van de uEye-camera.

### 6.2.4 Meetresultaten JPG-formaat als model

Indien het visiesysteem het model aanleert m.b.v. een JPG-formaat zijn de afmetingen van het voorwerp op de band bijna nooit gelijk aan de afmetingen van het model. Het visiesysteem moet dus zoeken op geschaalde voorwerpen. Geschaald zoeken vergroot in beide systemen de verwerkingstijd. Zoals weergegeven in tabel 20 kunnen beide visiesystemen echter met de maximale bandsnelheid van 1 m/s werken.

Tabel 20: Berekening maximale bandsnelheid indien model aangeleerd met JPG-formaat

	$M$ [mm]	$\Phi$ [mm]	$t_{ds}$ [ms]	$v$ [m/s]
<b>Cognex7010</b>	209	54	100	1,55
<b>IDS uEye-2230RE-C</b>	435	54	172	2,22

Het verschil in haalbare bandsnelheid is hier weer te verklaren door de snellere CPU en het groter beeld van de camera bij het IDS-visiesysteem.

### 6.2.5 Meetresultaten DXF-formaat als model

Aangezien het Cognex-visiesysteem het aanleren van een model m.b.v. DXF-formaat niet ondersteunt, is in tabel 21 enkel de maximum haalbare bandsnelheid voor het IDS-visiesysteem weergegeven. Indien het model is aangeleerd m.b.v. DXF-formaat moet het visiesysteem zoeken

naar geschaalde voorwerpen. Het is namelijk bijna nooit zo dat het voorwerp op de band dezelfde afmetingen heeft als de contouren van het aangeleerde model.

**Tabel 21: Berekening maximale bandsnelheid in dien model aangeleerd met DXF-formaat**

	<b>M</b> [mm]	<b><math>\Phi</math></b> [mm]	<b><math>t_{ds}</math></b> [ms]	<b>v</b> [m/s]
<b>IDS uEye-2230RE-C</b>	435	54	136	2,80

De maximum haalbare bandsnelheid valt ruim boven de 1 m/s. Het visiesysteem is dus in staat om het voorwerp altijd te detecteren bij een bandsnelheid van 1 m/s. Daarnaast is het duidelijk dat er nog veel extra verwerkingstijd is toegelaten om bv. naar gedraaide voorwerpen te zoeken. Deze metingen zijn niet uitgevoerd binnen de thesis.

### 6.3 Overige parameters

De overige te beoordelen parameters zijn robuustheid, flexibiliteit, gebruiksgemak en kostprijs. Deze parameters zijn subjectief beoordeeld. De laatste parameter kostprijs is vergeleken in de vergelijkingsmatrix in de volgende paragraaf.

Het Cognex-visiesysteem heeft een interne CPU voor de beeldverwerking. Het visiesysteem met de uEye camera en Halcon heeft een externe PC nodig voor de beeldverwerking. Hierdoor is er een extra onderdeel in de installatie aanwezig dat kan falen waardoor de robuustheid van het Cognex-visiesysteem hoger is.

De In-Sight software, voor het Cognex-systeem, werkt enkel met het Windows-besturingssysteem. Door de eenvoudige *user interface* van de In-Sight software is er een beperking op de flexibiliteit van het systeem. De code is niet vrij programmeerbaar, maar kan enkel de opdrachten uitvoeren die ingebouwd zijn in de In-Sight software. Dit maakt het systeem daarentegen wel eenvoudig in gebruik. De mogelijkheid om modellen aan te leren beperkt zich tot het rechtstreeks aanleren met behulp van een afbeelding die de camera maakt en het aanleren met behulp van JPG-bestanden. De ingang van de camera kan dienen om beelden te nemen via een externe trigger.

Halcon werkt op de besturingssystemen Windows, Linux en Mac. Het visiesysteem dat bestaat uit de IDS camera samen met de Halcon software is vrij programmeerbaar. Dit vergt meer werk voor de programmeur dan bij het instellen van de In-Sight software. Er bestaan wel *assistant*-functies die het mogelijk maken om basis operaties grafisch uit te voeren. Door deze functie te gebruiken maakt Halcon zelf de code aan die hoort bij de grafische instellingen. Vaak volstaan deze *assistant* functies echter niet en dient de programmeur extra code toe te voegen om een volledige oplossing te verkrijgen. Eenmaal het programma geprogrammeerd is in Halcon is het mogelijk de code te exporteren naar: C, C++, C#, visual basic en tekst *file*. Hierdoor bestaat de mogelijkheid om een *user interface* te ontwerpen naar alle behoefte van de gebruiker. Daarnaast kan Halcon modellen op drie manieren aanleren. Zo kan het aanleren zowel rechtstreeks via de camera als met behulp van JPG- en DXF-bestanden gebeuren. De uEye-camera bezit net zoals het Cognex-visiesysteem een ingang die dienst kan doen als externe trigger.

Ten slotte is de *teach-in* methode via JPG-formaat vrij omslachtig. De gebruiksvriendelijkheid en flexibiliteit binnen deze methode liggen eerder laag. Dit is zo voor beide opstellingen en ligt niet aan één van de visiesystemen. Hiervoor zijn verschillende redenen:

- het nemen van de foto moet nauwkeurig gebeuren door de gebruiker;
- het bepalen van de parameters *accept tolerance*, *max deformation* en schalingsfactoren zijn niet eenvoudig;
- de parameters *accept tolerance*, *max deformation* en schalingsfactoren zijn afhankelijk van de opstelling en omgevingsinvloeden.

De JPG-bestanden zijn foto's van het echte voorwerp. Het zou nuttig zijn om via een softwarepakket het voorwerp na te tekenen en zo in te laden. Zo is de vorm van het voorwerp steeds correct en zijn de parameters voor de instelling van beide visiesystemen gemakkelijker te bepalen.

## 6.4 Vergelijkingsmatrix

Tabel 22: Vergelijkingsmatrix tussen beide opstellingen

Te testen parameter	Opstelling		IDS uEye-2230RE-C	
	Cognex7010 Beoordeling	Opmerking	Beoordeling	Opmerking
Nauwkeurigheid - 1 voorwerp vaste hoek - 1 voorwerp variabel (theta 10°) - 1 voorwerp variabel (theta 20°) - 1 voorwerp variabel (theta 30°) - 1 voorwerp variabel (theta 40° - 60°) - 1 voorwerp variabel (tussen 70° en 180°) - Meerdere voorwerpen vaste hoek	Hoog 0,1 m/s - 1 m/s 0,1 m/s - 0,8 m/s 0,1 m/s - 0,5 m/s 0,1 m/s - 0,3 m/s 0,1 m/s - 0,2 m/s 0,1 m/s - 0,1 m/s	Niet geschikt voor elke bandsnelheid	Hoog 0,1 m/s - 1 m/s 0,1 m/s - 1 m/s 0,1 m/s - 1 m/s 0,1 m/s - 1 m/s 0,1 m/s - 1 m/s 0,1 m/s - 1 m/s 0,1 m/s - 1 m/s	
Teach-in via JPG-formaat	Matig	Methode is onpraktisch	Matig	Methode is onpraktisch
Teach-in via DXF-formaat	Geen	Niet mogelijk	Hoog	
Verwerkingssnelheid	Matig	Smart-camera	Hoog	Extern besturingsstelsel
Robuustheid	Hoog		Hoog	Net iets minder robuust als de andere opstelling
Flexibiliteit	Laag	Software In-Sight heeft beperkingen	Hoog	Vrij programmeerbaar en Halcon library's
Gebruiksgemak	Hoog		Matig	Programmeervaardigheden vereist
Kostprijs	Laag	Smart-camera	Hoog	Halcon licentie

## 7 Besluit

In deze thesis werden twee soorten 2D-visiesystemen geïmplementeerd in een robotopstelling voor het volgen van vlakke voorwerpen. Deze systemen zijn vervolgens getest en vergeleken op verschillende parameters. De vergeleken visiesystemen zijn enerzijds de Cognex 7010 smartcamera en anderzijds de IDS uEye-2230RE-C camera in combinatie met de Halcon beeldverwerkingssoftware op een extern besturingssysteem.

De algemene werking is voor beide visiesystemen gelijkaardig. Het systeem scant een lopende transportband en zoekt naar een voorwerp dat vooraf is aangeleerd. Binnen deze thesis zijn naamkaartjes als voorwerp gebruikt. Indien het visiesysteem het aangeleerde voorwerp herkent geeft het een synchronisatiepuls aan de robot waardoor de robot de bandverplaatsing begint op te meten. Na de synchronisatiepuls verzendt het visiesysteem de coördinaten van het gevonden voorwerp via een TCP/IP-verbinding naar de robot. Doordat de robot is gesynchroniseerd met de band kan de robot op basis van verkregen coördinaten de plaats van het voorwerp op de band berekenen. Vervolgens verplaatst de robot zich naar het voorwerp en volgt dit gedurende enkele centimeters. Hierna keert de robot terug naar de *home* positie en wacht daar tot er een nieuw voorwerp is herkend om dan voorgaande procedure te herhalen.

De geteste en vergeleken parameters zijn nauwkeurigheid, verwerkingssnelheid, robuustheid, flexibiliteit en gebruiksgemak. De nauwkeurigheid is gekwantificeerd door een meetprocedure waarbij de robot lijnen tekent ter hoogte van het te volgen voorwerp. Door het voorwerp meerdere keren voorbij het visiesysteem te laten komen en de robot dus meerdere keren een lijn te laten tekenen is het mogelijk om de spreiding op deze lijnen op te meten. Deze opgemeten spreiding samen met de gemiddelde afstand tussen de getekende lijnen en de referentielijn, die zich ter hoogte van het middelpunt van het voorwerp bevindt, zijn een maat voor de nauwkeurigheid.

Uit de meetresultaten voor de nauwkeurigheid en verwerkingssnelheid blijkt dat het Cognex-visiesysteem bruikbaar is voor niet verdraaide voorwerpen. Indien de voorwerpen verdraaid liggen op de band behaalt het dit visiesysteem nauwkeurigheids met spreidingen tussen 1 mm en 3,5 mm. De CPU van het visiesysteem is niet krachtig genoeg om naar verdraaide voorwerpen met een hoek tot 180° te zoeken. Zelfs bij een toegelaten hoekverdraaiing van 60° is de maximum toegelaten bandsnelheid slechts 0,2 m/s. Ook bij het zoeken naar twee niet verdraaide voorwerpen loopt de rekentijd op waardoor de maximum toegelaten bandsnelheid voor het visiesysteem vastligt op 0,6 m/s. Bij deze snelheid behaalt het systeem een spreiding van 1 mm voor lage bandsnelheid en 2 mm voor hoge bandsnelheid. Als het model aangeleerd is m.b.v. een JPG-bestand is de maximale bandsnelheid van 1 m/s haalbaar. De spreiding op de lijnen wijzigt niet t.o.v. de metingen met het rechtstreekse aangeleerde model maar de afstand tot de referentielijn loopt op waardoor de nauwkeurigheid daalt.

Het IDS-visiesysteem heeft een verwerkingssnelheid die hoog genoeg is om delen op de band te detecteren bij de maximale bandsnelheid van 1 m/s voor alle uitgevoerde metingen. De nauwkeurigheid van het visiesysteem is relatief hoog bij lage bandsnelheden (0,1 – 0,5 m/s) met een spreiding van 1 – 2 mm. Bij hoge bandsnelheden (0,6 – 1 m/s) heeft het systeem een lagere nauwkeurigheid met spreidingen tussen de 1 en 4 mm. De robot tekent met regelmaat een foutieve lijn op een bepaalde afstand achter het lijnenstel. Deze afwijking binnen het systeem komt

regelmatig voor en is dus niet te verwaarlozen. Het beïnvloed de nauwkeurigheid nadelig en heeft meer invloed bij een stijgende bandsnelheid. Tussen de verschillende soorten testen zijn geen verschillen in nauwkeurigheid merkbaar. De nauwkeurigheid is dus enkel functie van de bandsnelheid omwille van de afwijkende lijn die regelmatig voorkomt.

De hogere verwerkingssnelheid van het visiesysteem met externe PC voor de beeldverwerking heeft twee redenen. De eerste reden is het verschil in snelheid van de processor gebruikt voor de beeldverwerking. Het Cognex-systeem is een smart-camera met een relatieve zwakke interne processor. Het IDS-systeem maakt daarentegen gebruik van een externe krachtige processor. Hierdoor kan het IDS-systeem de berekeningen sneller uitvoeren. De tweede reden is het grotere beeld van de uEye-camera, waardoor het systeem meer bandoppervlakte tegelijk kan scannen.

De parameters robuustheid, flexibiliteit en gebruiksgemak zijn subjectief beoordeeld. De In-Sight software, die dient om de Cognex-camera in te stellen, werkt met een eenvoudige grafische user interface waardoor er een beperking ligt op de flexibiliteit van het systeem. De code is niet vrij programmeerbaar, maar kan enkel de opdrachten uitvoeren die ingebouwd zijn in de In-Sight software. Dit maakt het systeem daarentegen wel eenvoudig in gebruik. Het aanleren van modellen kan rechtstreeks of m.b.v. een JPG-bestand gebeuren. De beeldverwerkingssoftware Halcon daarentegen is vrij programmeerbaar. Dit vergt meer werk voor de programmeur dan bij het instellen van de In-Sight software. Er bestaan *assistant* functies die het mogelijk maken om basisoperaties grafisch uit te voeren. Vaak volstaan deze *assistant* functies echter niet en dient de programmeur extra code toe te voegen om een volledige oplossing te verkrijgen. Eenmaal het programma geprogrammeerd is in Halcon kan Halcon de code exporteren naar: C, C++, C#, visual basic en tekst file. Hierdoor is er de mogelijkheid een eigen user interface te ontwerpen die eenvoudig werkt voor de operator. Halcon kan het model rechtstreeks, met behulp van een JPG-bestand of met behulp van een DXF-bestand aanleren.

Verder onderzoek kan erin bestaan het gehele systeem te kalibreren door middel van interactie tussen robot en visiesysteem om zo een hogere nauwkeurigheid te bereiken. Een tweede verbetermogelijkheid bestaat erin de bandsnelheid continu op te meten. In de huidige opstelling moet de robot voor elke bandsnelheid eerst de module snelheidsmeting doorlopen. Bij een wijzigende bandsnelheid komt de werkelijke snelheid anders niet meer overeen met de gemeten snelheid. Indien de robotsturing de bandsnelheid tijdens bedrijf continu opmeet kan de bandsnelheid wijzigen zonder het systeem terug te moeten initialiseren.



# Bibliografie

- [1] TEKSIS LTD., „Teksis,” [Online]. Available:  
<http://www.teksis.co.uk/assets/images/products/thumbnails/4430.jpg>.
- [2] Robot welding , „Robot welding the essential guide,” 2015. [Online]. Available:  
[http://www.robotwelding.co.uk/KRC4\\_GO\\_020.jpg](http://www.robotwelding.co.uk/KRC4_GO_020.jpg).
- [3] Newsmax, „Newsmax,” 2011. [Online]. Available:  
[http://www.newsmax.de/bilder/110530\\_31167.jpg](http://www.newsmax.de/bilder/110530_31167.jpg).
- [4] transportbanden.biz, „transportbanden.biz,” 2015. [Online]. Available:  
<http://www.transportbanden.biz/images/categories/rvs.jpg>.
- [5] J. Baeten, „websites.khlim,” 2010. [Online]. Available: <http://websites.khlim.be/jbaeten/>.
- [6] Vision-doctor, „Lens calculator - Calculating the focal length,” [Online]. Available:  
<http://www.vision-doctor.co.uk/optical-calculations/lens-calculator-calculation-focal-length.html>. [Geopend 2015].
- [7] MvTec Software GMBH, „The Various Modes of Grabbing,” in *Solution Guide II-A Image Aquisition*, Munchen, Germany, MvTec Software GMBH, 2015, pp. 23-36.
- [8] Y. D. W. Z. C. Z. Jiangping Mei, „Fast detection, position and classification of moving objects,” *Optik - International Journal for Light and Electron Optics*, vol. 2009, nr. 121, p. 2176–2178, 2009.
- [9] Refsnes Data, „XML Tree,” Refsnes Data, [Online]. Available:  
[http://www.w3schools.com/xml/xml\\_tree.asp](http://www.w3schools.com/xml/xml_tree.asp). [Geopend 2015].
- [10] KUKA Roboter GmbH, KUKA.Ethernet KRL XML 1.1, Augsburg: KUKA Roboter GmbH, 2007.
- [11] KUKA Roboter GmbH, KUKA.ConveyorTech 3.3 for KUKA System Software (KSS) 5.x, Augsburg: KUKA Roboter GmbH, 2010.
- [12] KUKA Roboter GMBH, „In gebruik nemen van de robot,” in *Training Robotprogrammering 1*, Augsburg, Duitsland, KUKA Roboter GMBH, 2012, pp. 49 - 90.
- [13] Cognex Corporation, In-Sight 7000 manual, Natick, USA: Cognex Corporation, 2011.
- [14] Cognex Corporation, „Cognex products,” 2015 . [Online]. Available:  
<http://www.cognex.com/productstemplate.aspx?id=13608&rdr=true&LangType=2057>.
- [15] Cognex corporation, *In-Sight Explorer 4.9.1 Help*, Natick, USA: Cognex corporation, 2014.

- [16] MVTec Software GmbH, „Halcon applications,” 2015. [Online]. Available: <http://www.halcon.com/halcon/applications/imgs/3d-vision-figure-3d-calibration.png>.
- [17] IDS, „Datasheet IDS uEye 2330RE-C-HQ Rev.3”.
- [18] MVTec Software GmbH, „Non-Real-Time Grabbing Using grab\_image,” in *Solution Guide II-A Image Acquisition*, München, Germany, MVTec Software GmbH, 2012, pp. A-24.
- [19] IDS Imaging Development Systems GmbH, „uEye Camera Manual 4.60,” IDS Imaging Development Systems GmbH, [Online]. Available: [https://en.ids-imaging.com/manuals/uEye\\_SDK/EN/uEye\\_Manual/index.html](https://en.ids-imaging.com/manuals/uEye_SDK/EN/uEye_Manual/index.html). [Geopend 2015].
- [20] MVTec Software GmbH, „Grabbing Without Delay Using Asynchronously Resettable Cameras,” in *Solution Guide II-A Image Acquisition*, München, Germany, MVTec Software GmbH, 2012, pp. A-26.

# Bijlagen

- 1 Aansluitschema Cognex 7010 inbedrijfname
- 2 Aansluitschema Cognex 7010 bedrijfsmodus
- 3 Aansluitschema IDS UI 2230-RE-C
- 4 Router.xml
- 5 XmlApiConfig.xml
- 6 Cognex7010.xml
- 7 Cognex7010+.xml
- 8 sps.sub programma
- 9 snelheidsmeting programma
- 10 meetprogramma nauwkeurigheid 1 kaartje
- 11 meetprogramma nauwkeurigheid 2 kaartjes
- 12 meetprogramma nauwkeurigheid hoekverdraaiing
- 13 DEMO
- 14 In-Sight 7000 datasheet
- 15 Cognex programma 1 naamkaartje
- 16 Cognex programma 2 naamkaartjes
- 17 UI-2230RE-C-HQ datasheet
- 18 Template matching create object 1 kaartje
- 19 Template matching create object 2 kaartjes
- 20 Template matching create object DXF
- 21 Template matching create objec JPG
- 22 Template matching 1 kaartje
- 23 Template matching 2 kaartjes
- 24 Template matching DXF
- 25 Template matching JPG
- 26 Demo filmpje

# Auteursrechtelijke overeenkomst

Ik/wij verlenen het wereldwijde auteursrecht voor de ingediende eindverhandeling:  
**Integratie van robotvolgsysteem met 2D-visie voor vlakke voorwerpen**

Richting: **master in de industriële wetenschappen: energie-automatisering**  
Jaar: **2015**

in alle mogelijke mediaformaten, - bestaande en in de toekomst te ontwikkelen - , aan de Universiteit Hasselt.

Niet tegenstaand deze toekenning van het auteursrecht aan de Universiteit Hasselt behoud ik als auteur het recht om de eindverhandeling, - in zijn geheel of gedeeltelijk -, vrij te reproduceren, (her)publiceren of distribueren zonder de toelating te moeten verkrijgen van de Universiteit Hasselt.

Ik bevestig dat de eindverhandeling mijn origineel werk is, en dat ik het recht heb om de rechten te verlenen die in deze overeenkomst worden beschreven. Ik verklaar tevens dat de eindverhandeling, naar mijn weten, het auteursrecht van anderen niet overtreedt.

Ik verklaar tevens dat ik voor het materiaal in de eindverhandeling dat beschermd wordt door het auteursrecht, de nodige toelatingen heb verkregen zodat ik deze ook aan de Universiteit Hasselt kan overdragen en dat dit duidelijk in de tekst en inhoud van de eindverhandeling werd genotificeerd.

Universiteit Hasselt zal mij als auteur(s) van de eindverhandeling identificeren en zal geen wijzigingen aanbrengen aan de eindverhandeling, uitgezonderd deze toegelaten door deze overeenkomst.

Voor akkoord,

**Dingenen, Bert**

**Mertens, Rob**

Datum: **9/06/2015**