

## AFSTUDEERSCHRIPTIE

### *SIoux EXPERIENCE CENTER APP*



Auteur	Joris Reijrink
Datum	09-01-2013
Versie	1.0
Status	Definitief
Project	Sioux Experience Center App

## TITELPAGINA

**Gegevens student**

Naam + voorletters student:

Studentnummer:

Opleiding:

Stageperiode datum:

Reijrink J.

2127713

ICT &amp; Technology Voltijd

10 september 2012 t/m 1 februari 2013

**Gegevens bedrijf**

Naam bedrijf/instelling:

Plaats:

Naam + voorletters bedrijfsbegeleider:

Functie bedrijfsbegeleider:

Sioux Embedded Systems

Eindhoven

Huijbers R.

Senior Software Engineer

**Gegevens docentbegeleider**

Naam + voorletters docentbegeleider:

Schreur B.

---

Getekend voor gezien door de bedrijfsbegeleider Rico Huijbers:



Datum: 9 januari 2013

## REVISIE HISTORIE

Versie	Datum	Auteur	Omschrijving
0.1	04-01-2013	Joris Reijrink	Initiële versie
0.2	07-01-2013	Joris Reijrink	Feedback schoolbegeleider verwerkt
0.3	09-01-2013	Joris Reijrink	Feedback bedrijfsbegeleider verwerkt
1.0	09-01-2013	Joris Reijrink	Definitieve versie

## VOORWOORD

Deze scriptie is gemaakt tijdens mijn afstudeerstage voor de opleiding ICT & Technology aan de Fontys Hogescholen te Eindhoven. In de periode van september 2012 tot februari 2013 heb ik gewerkt aan deze afstudeeropdracht bij Sioux Embedded Systems te Eindhoven. De afstudeeropdracht bestond uit het ontwikkelen van een systeem dat objecten kan herkennen.

In dit verslag wordt er toelichting geven over de aanleiding van de opdracht, werkwijze en gemaakte beslissingen. Ook het onderzoek en het gehele proces van ontwerpen tot en met implementeren worden in dit verslag besproken.

Ik wil graag mijn bedrijfsbegeleider Rico Huijbers bedanken voor zijn goede begeleiding tijdens mijn afstudeerperiode, technische ondersteuning en kritische doch rechtvaardige feedback op documenten. Ook wil ik mijn schoolbegeleider Ben Scheur bedanken voor het bewaken van het afstudeerproces en zijn feedback op documenten. Tot slot wil ik Sioux bedanken voor deze mooie leerzame afstudeerstage.

# INHOUDSOPGAVE

<b>SAMENVATTING</b>	<b>1</b>
<b>SUMMARY</b>	<b>2</b>
<b>DEFINITIES EN AFKORTINGEN</b>	<b>3</b>
Definities	3
Afkortingen	4
<b>1 INLEIDING</b>	<b>5</b>
1.1 Bedrijf	5
<b>2 OPDRACHT</b>	<b>6</b>
2.1 Eisen	6
2.2 Onderzoeksvragen	6
<b>3 WERKWIJZE</b>	<b>7</b>
<b>4 VOORONDERZOEK HERKENNINGSTECHNIEKEN</b>	<b>8</b>
4.1 Evaluatiecriteria	8
4.2 Computer Vision	9
4.3 RFID/NFC	10
4.4 QR Code	11
4.5 IPS + Kompas + Gyroscop	12
4.6 Kinect	13
4.7 Coded Light	15
4.8 Voor- en nadelen	16
4.9 Conclusie	17
<b>5 NFC</b>	<b>18</b>
5.1 Onderzoek	18
5.2 Benodigdheden	18
5.3 Stappenplan	19
5.4 Resultaten	19
5.5 Conclusie	20
<b>6 COMPUTER VISION</b>	<b>21</b>
6.1 Werking algoritmes	21
6.2 Overzicht algoritmes	22
6.3 Onderzoek	23
6.4 Parameters	24
6.5 Ontwerp testapplicatie	25
6.6 Resultaten	28
6.7 Conclusie	34

<b>7 HERKENNINGSAPPLICATIE</b>	<b>35</b>
7.1 Geleerd bij onderzoek	35
7.2 Ontwerp	35
7.3 Ontwikkeling	44
7.4 Testen	46
<b>8 CONCLUSIE</b>	<b>47</b>
8.1 Aanbevelingen	47
<b>EVALUATIE</b>	<b>48</b>
<b>LITERATUURLIJST</b>	<b>49</b>
<b>BIJLAGEN</b>	<b>51</b>



## SAMENVATTING

Sioux is een innovatieve technologiepartner met 300 ingenieurs op het gebied van technische software, remote solutions, elektronica en industriële wiskunde. Het Sioux Experience Center (SEC) is de plek waar de managers van Sioux klanten een rondleiding geeft door haar portfolio. De manager vertelt daarbij een verhaal over de producten in het SEC. Omdat deze managers soms het verhaal bij een product niet precies weten is deze nog wel eens verschillend. Om het verhaal consistent te maken wordt er een Android applicatie ontwikkeld die met behulp van productherkenning de manager meer informatie geeft over de producten. Op deze manier heeft de manager altijd informatie bij de hand over de producten in het SEC tijdens rondleidingen met klanten.

Tijdens de uitvoering van het project is er gebruikt gemaakt van een mengeling tussen scrum en het V-model. Er is bij de opstart een globale planning gemaakt van alle opleveringen. Hierbij is voor elke oplevering een deadline vastgesteld. Dit lijkt erg op het V-model. Vervolgens zijn er continu sprints geweest met een sprintlengte van twee weken.

Voor het herkennen van objecten zijn er verschillende technieken mogelijk. Deze verschillende technieken zijn onderzocht op bruikbaarheid voor de applicatie. Hieruit is gebleken dat NFC en Computer Vision volgens informatie uit externe bronnen een oplossing zouden kunnen bieden.

Na verder onderzoek bleek dat het bereik van NFC in de praktijk te klein is. Als een tag onzichtbaar is weggewerkt is het soms niet eens mogelijk om hem te scannen. Uit het Computer Vision onderzoek is gebleken dat SIFT3 het algoritme is dat het beste heeft gepresteerd ten aanzien van de eisen van de klant. Met dit algoritme voldoet Computer Vision aan bijna alle eisen van de klant. De eis waar niet aan wordt voldaan is betrouwbaarheid. Dit is opgelost doordat het algoritme hetzelfde product meerdere keren moet herkennen voor het naar de gebruiker als herkend aangeduid wordt. Hierdoor duurt herkenning wel langer. Uit de testresultaten blijkt dat dit geen probleem is indien de netwerkontvangst goed is. Het uitvoeren van het algoritme gebeurt op een externe server. Hier is voor gekozen omdat dit op de Android tablet zelf 12 keer langer duurt.

Met het opgeleverde systeem is het mogelijk om met de tablet producten uit het SEC te herkennen. De manager kan met behulp van de tablet extra informatie verkrijgen van een product. Op deze manier wordt de manager ondersteund tijdens het rondleiden van een klant door het SEC.

## SUMMARY

Sioux is an innovative technology partner with 300 engineers in the field of technical software, remote solutions, electronics and industrial mathematics. The Sioux Experience Center (SEC) is the place where the managers of Sioux guide their customers through its portfolio. The manager tells a story about the products in the SEC. The story of a product is sometimes different because the manager does not always exactly know the story. To make the story more consistent an Android application will be developed. This application uses product recognition to give the manager more information about the products. In this way, the manager always has correct information about the products in the SEC during tours with clients.

During the implementation of the project a mix between scrum and the V-model has been used. An overall planning of all deliveries is made at the startup of the project. Each delivery has a deadline. This is very similar to the V-model. Next there has been continuous sprints with a length of two weeks.

There are several techniques available for object recognition. These different techniques are analyzed for usability in the application. It turned out that NFC and Computer Vision could provide a solution.

Further investigation showed that the range of NFC is too small in practice and tags that are concealed cannot always be scanned. The Computer Vision research has shown that the SIFT3 algorithm has performed best with regard to the requirements of the customer. With this algorithm, Computer Vision meets almost all requirements of the customer. The requirement that is not been met is reliability. This is solved by making the algorithm recognize the same product multiple times before it is indicated as recognized to the user. This way the recognition takes longer. The results show that this is not a problem if the network connection is good. The execution of the algorithm is done on a remote server. This is because the Android tablet takes 12 times longer to execute the algorithm.

With the delivered system, it is possible to use the tablet to recognize products in the SEC. The manager can use the tablet to obtain additional information of a product. In this way, the manager is supported while guiding a customer in the SEC.



## DEFINITIES EN AFKORTINGEN

### Definities

Activity	: Een applicatiecomponent in Android die een scherm voorziet waarmee gebruikers kunnen communiceren
API	: Verzameling definities op basis waarvan een computerprogramma kan communiceren met een ander programma of onderdeel
Client	: Een applicatie of een computersysteem met toegang tot een ander systeem, de server, via een netwerk.
Content	: Media bestand
CPU	: Processor van computer
CUDA	: Technologie die het de programmeur mogelijk maakt om algoritmes uit te voeren op de GPU.
Database	: Digitaal opgeslagen archief
False positive	: Foutieve match overeenkomst
GPU	: Videokaart van computer
Keypoint	: Resultaat van foto-extractie door vision algoritme
Match	: Overeenkomst tussen twee afbeeldingen
Phenom	: Een elektronenmicroscop ontwikkeld door Phenom-World
Referentiedatabase	: Database met keypoints van alle productfoto's
Referentiefoto	: Foto uit de referentiedatabase
Request	: Aanvraag van dienst door client
Server	: Computer of programma dat diensten verleent aan clients
Wifi	: Draadloos datanetwerk
XML	: Standaard voor de syntaxis van formele opmaaktalen waarmee men gestructureerde gegevens kan weergeven in de vorm van platte tekst.

## Afkorting

API	: Application Programming Interface
CPU	: Central Processing Unit
CUDA	: Compute Unified Device Architecture
GPU	: Graphics Processing Unit
HTTP	: Hypertext Transfer Protocol
IPS	: Indoor Positioning System
NFC	: Near field communication
PID	: Product Initiation Document
QR Code	: Quick Response Code
R&D	: Research & Development
RFID	: Radio-Frequency Identification
SEC	: Sioux Experience Center
URS	: User Requirements Specification
UML	: Unified Modeling Language
XML	: Extensible Markup Language

# 1 INLEIDING

Het Sioux Experience Center (SEC) is de plek waar de managers van Sioux klanten een rondleiding geeft door haar portfolio. De manager vertelt daarbij een verhaal over de producten in het SEC. Omdat deze managers soms het verhaal bij een product niet precies weten is deze nog wel eens verschillend.

Het doel van de afstudeeropdracht is een oplossing ontwikkelen die de manager kan ondersteunen tijdens deze rondleiding door middel van objectherkenning. Op deze manier is het verhaal over de producten in het SEC consistent.

Er zijn een hele hoop verschillende herkenningstechnieken mogelijk voor dit probleem. Daarom wordt er eerste onderzocht welke het beste gebruikt kan worden door middel van een onderzoek.

In hoofdstuk 2 zijn alle details van de opdracht omschreven. Hoofdstuk 3 bespreekt de werkwijze die is gebruikt tijdens de afstudeerstage. In hoofdstuk 4 worden verschillende herkenningstechnieken onderzocht op bruikbaarheid aan de hand van de eisen. In hoofdstuk 5 & 6 worden de twee overgebleven technieken uit het vooronderzoek verder onderzocht. Hoofdstuk 7 bespreekt de uiteindelijk implementatie met het bijbehorende ontwerp. Tot slot geeft hoofdstuk 8 een afrondende eindconclusie.

## 1.1 Bedrijf

Sioux is een innovatieve technologiepartner met 300 ingenieurs op het gebied van technische software, remote solutions, elektronica en industriële wiskunde. Sioux heeft vestigingen in Nederland, België en back offices in Rusland en Vietnam.



Sioux helpt vooraanstaande bedrijven met nieuwe manieren van innovatie en bij het verbeteren van hun producten of productieproces. Zij ondersteunen daarbij de R&D afdeling of fungeren als de R&D afdeling, zelfs tot productie aan toe. In beide gevallen helpen ze bij het verkorten van de ontwikkeltijd en het ontwikkelen van een duurzaam concurrentievoordeel.

## 2 OPDRACHT

In het SEC geeft de manager rondleiding aan klanten. Omdat het verhaal bij een product wel eens verschilt is het idee naar voren gekomen om een tablet applicatie te maken die meer informatie geeft over de producten in het SEC.

Met behulp van productherkenning door de tablet applicatie kan de manager meer informatie verkrijgen over de producten in het portfolio. Als een product herkend is kan de tablet verschillende bijbehorende content laten zien zoals een presentatie, video of document. Op deze manier heeft de manager altijd informatie bij de hand over de producten in het SEC tijdens rondleidingen met klanten. Omdat er veel technieken zijn voor productherkenning wordt er onderzocht welke techniek het beste bruikbaar is voor het systeem.

Ook moeten de producten eenvoudig beheerd kunnen worden om zo nieuwe producten toe te kunnen voegen.

### 2.1 Eisen

Er zijn een paar belangrijke eisen die de klant heeft gesteld over het systeem. Het SEC mag niet zichtbaar veranderd worden, voor de gebruiker mag er niets zichtbaar zijn van het systeem. Dit heeft te maken omdat het systeem innovatief en professioneel moet overkomen. De user interface is daarom een belangrijk aspect van de applicatie omdat de gebruiker overweldigd moet worden door het systeem. Hierdoor zal de applicatie een innovatieve, uitdagende en professionele user interface moeten hebben. Ook is het belangrijk dat de applicatie makkelijk te bedienen is en intuïtief werkt voor de gebruiker. Verder moet het systeem betrouwbaar zijn. Het systeem moet de correcte producten herkennen en dit mag ook niet te lang duren. Meer details over alle eisen kunnen gevonden worden in het URS in de bijlage.

### 2.2 Onderzoeksvragen

Voor het onderzoek naar verschillende herkenningstechnieken zijn onderzoeksvragen opgesteld. Deze onderzoeksvragen worden als leidraad gebruikt in het onderzoek en worden indirect beantwoord in het onderzoek.

#### Hoofdvraag

- Wat is in dit geval de beste techniek voor objectherkenning?

#### Deelvragen

- Welke technieken zijn er voor objectherkenning?
- Wat zijn de voor- en nadelen van elke techniek?
- Welke overgebleven techniek sluit het beste aan bij de requirements?
- Welke implementatie mogelijkheden zijn er met deze technologieën?
- Mogelijkheid tot Off the shelf software?

### 3 WERKWIJZE

Tijdens de uitvoering van het project is er gebruikt gemaakt van een mengeling tussen scrum en het V-model. Er is bij de opstart van het project een globale planning gemaakt van alle opleveringen. Hierbij is voor elke oplevering een deadline vastgesteld. Dit lijkt erg op het V-model. Vervolgens zijn er continu sprints geweest met een sprintlengte van twee weken. Bij iedere sprint is er geëvalueerd en werd er de nieuwe sprint gepland. Bij elke sprintevaluatie werd ook gekeken of de globale planning nog op koers lag.

De volgende producten zijn opgeleverd:

- Project Initiatie Document (PID)
- Requirements
- Testplan
- Onderzoek beste herkenningmethode
- Systeemontwerp
- Tussenpresentatie
- Software
- Testresultaten
- Eindpresentatie
- Eindverslag
- Sprint rapporten

Onderdelen waaraan werd gewerkt werden direct gedocumenteerd. Als een document klaar was werd deze opgestuurd naar de stagebegeleider voor feedback. Op deze manier was de documentatie altijd actueel. De complete aanpak van het project kan gevonden worden in het PID in de bijlage.



## 4 VOORONDERZOEK HERKENNINGSTECHNIEKEN

Voor het herkennen van objecten zijn er verschillende technieken. In dit hoofdstuk worden verschillende herkenningstechnieken beschreven die een oplossing zouden kunnen bieden voor het gewenste systeem. Er wordt per techniek het volgende beschreven:

- Omschrijving van de werking
- Toepassing in Sioux Experience Center
- Voor- en nadelen
- Onzekerheden

Als een techniek verder onderzocht gaat worden voor toepassing in het Sioux Experience Center zullen de onzekerheden van de techniek nog nader onderzocht moeten worden.

De volgende technieken worden behandeld in dit hoofdstuk:

- Computer Vision
- RFID
- NFC
- QR Code
- IPS + Kompas + Gyroscop
- Kinect
- Coded Light

In sectie 4.8 worden de voor- en nadelen van elke techniek vergeleken met de eisen van de klant. Met deze uitkomsten word er voor elke techniek bepaald of deze genoeg potentie en overlapping heeft met de eisen van de klant om nader onderzocht te worden.

In het vooronderzoek is de diepgang per techniek beperkt. Dit is omdat er globaal wordt gekeken of een techniek bruikbaar is. De bruikbaarheid van een techniek wordt bepaald door informatie uit externe bronnen. De technieken die volgens de externe bronnen bruikbaar zouden kunnen zijn voor het gewenste systeem zullen vervolgens verder in de diepte worden onderzocht.

### 4.1 Evaluatiecriteria

Om te bepalen welke techniek het beste aansluit op de eisen van de klant zijn er criteria opgesteld waar een techniek aan moet voldoen.

- Er mag niks zichtbaar veranderen aan het SEC
- Robuustheid/Betrouwbaarheid van 90%
- Mogelijkheid om producten die vlak bij elkaar liggen te onderscheiden (5 centimeter)
- Herkenningsafstand tussen 5 centimeter en 1 meter
- Het aanpassen van de te herkennen set objecten moet met zo min mogelijk moeite gebeuren



## 4.2 Computer Vision

Computer Vision is een techniek die methoden bevat voor het verkrijgen, verwerken, analyseren en begrijpen van digitale beelden. Er zijn een tal van toepassingen mogelijk met Computer Vision zoals een robotica systeem die zijn omgeving kan waarnemen, een bewakingssysteem die mensen kan herkennen of producten die automatisch geïnspecteerd worden op de lopende band. Het doel van Computer Vision is de wereld te modelleren of om objecten te herkennen via digitale beelden. (1) (2)

### 4.2.1 Toepassing

Deze techniek zou in het Sioux Experience Center gebruikt kunnen worden met behulp van de camera op het tablet device, die door middel van de binnengekomen digitale beelden producten kan herkennen.

### 4.2.2 Voor- en nadelen

Voordelen

- Geen aanpassingen nodig in het Sioux Experience Center
- Geen extra hardware nodig

Nadelen

- Veel processorkracht nodig om beelden te verwerken
- Gevoelig voor vervormingen

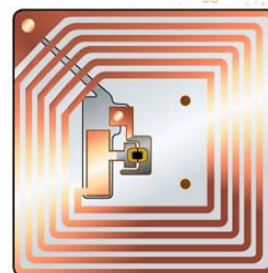
### 4.2.3 Onzekerheden

De betrouwbaarheid van deze techniek is nog onzeker en moet verder onderzocht worden. Het is vooral belangrijk dat onderzocht wordt hoe deze techniek reageert op verschillende hoeveelheden licht en op verschillende aanzichten van het product. Ook moet onderzocht worden of de hoeveelheid processorkracht op de tablet genoeg is om er vision op toe te passen.

## 4.3 RFID/NFC

Radio Frequency Identification (RFID) en Near Field Communication (NFC) zijn contactloze technologieën die radiofrequente elektromagnetische velden gebruiken om gegevens over te sturen van een tag naar een reader. Deze technologie kan gebruikt worden voor doeleinden zoals automatische identificatie of tracking.

NFC heeft een bereik rond de 5 centimeter, bij RFID ligt het bereik bij de 10 centimeter. NFC is in staat om kleine hoeveelheid data tussen een NFC-tag en een apparaat met NFC ondersteuning te sturen, of tussen twee NFC apparaten. (3) NFC wordt ook wel gezien als een extensie van RFID. Een NFC chip kan in drie verschillende standen werken: tag emulatie (ook wel passive mode), waar de NFC-chips zich gedraagt als een RFID tag, active mode waar de NFC zich gedraagt als een reader en peer-to-peer mode waarbij het mogelijk is om te communiceren tussen twee NFC devices (bijvoorbeeld bestandstransmissie). (4)



**Figuur 4.1. Een voedingsloze tag met antenne**

Sommige tags vereisen geen batterij en worden aangedreven door de elektromagnetische velden die worden gebruikt door de reader om te lezen. Anderen gebruiken een lokale voedingsbron en zenden radiogolven uit. De tag bevat elektronisch opgeslagen informatie die kan worden gelezen vanaf een afstand. In tegenstelling tot een barcode hoeft een tag niet in het gezichtsveld aanwezig te zijn, deze kan zelfs verwerkt zijn in het product. (5) (6)

### 4.3.1 Toepassing

Deze techniek zou in het Sioux Experience Center gebruikt kunnen worden met behulp van een tablet met RFID of NFC ondersteuning, of een tablet met een externe RFID reader. Op elk product komt een tag die vervolgens door de tablet gelezen kan worden. Dit is mogelijk door de tablet of reader dicht bij het product te positioneren.

### 4.3.2 Voor- en nadelen

Voordelen

- Betrouwbaar
- Makkelijk uitbreidbaar
- Tag kan worden weggewerkt in product
- Tablet met NFC ondersteuning beschikbaar

Nadelen

- Scan afstand van NFC is klein

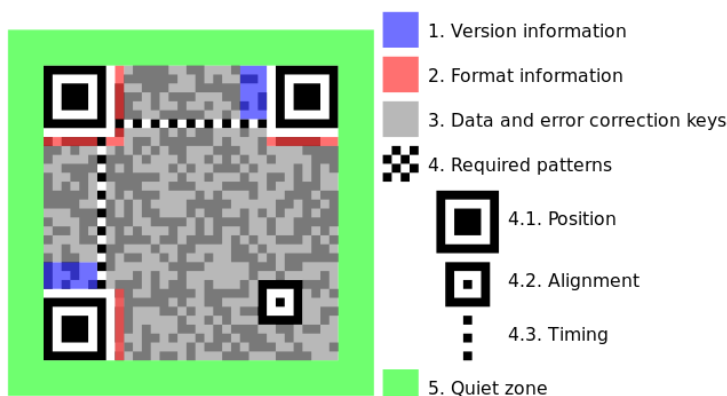
### 4.3.3 Onzekerheden

Het is nog onduidelijk wat de werkelijke afstand tot het scannen in de praktijk is. Het is belangrijk dat een tag niet in beeld is in het Sioux Experience Center, daarom moet nog onderzocht worden of de tag op een onzichtbare plaats nog steeds dergelijk gescand kan worden. Ook bestaat er de kans op overlapping met verschillende tags omdat de scanafstand juist te groot is. Tot slot is het niet duidelijk of het mogelijk is om een RFID reader op een tablet device aan te sluiten.

## 4.4 QR Code

Een QR code, dat staat voor “Quick Response”, is het handelsmerk van een type van matrix barcode die in eerste instantie ontworpen is voor de automotive industrie. Sinds kort is het systeem meer populair geworden buiten de industrie vanwege zijn snelle leesbaarheid en grote opslagcapaciteit in vergelijking met standaard barcodes. (7)

Een QR code bestaat uit zwarte modules (vierkante stipjes) in een vierkant patroon op een witte achtergrond en kan met elke telefoon gelezen worden zolang deze maar een camera heeft en



**Figuur 4.2. Opbouw van een QR code**

software om de QR code te decoderen. Dit kunnen bijna alle hedendaagse telefoons. De content die een QR code bevat kan variëren van een URL, telefoonnummer, SMS bericht, V-card data of gewoon een tekstbericht.

De technische specificatie voor een QR code zijn vastgelegd in de ISO-18004 standaard, zodat ze over de hele wereld hetzelfde zijn. De enige significante verschillen tussen

QR codes (afgezien van de gegevens ervan) is het aantal modules die nodig zijn om de gegevens op te slaan. Een versie 1 QR code bevat een reeks van 21x21 data-elementen, waarbij de reeks steeds met 4 modules groter wordt bij elke toename van het versienummer. De grootste standaard QR code is versie 40 die een reeks van 177x177 data-elementen bevat en maximaal 1852 bytes kan bevatten.

De resolutie van de camera van de gemiddelde hedendaagse telefoon is zodanig dat de grootte van de data modules op een QR code van versie 5 of hoger (37x37) een risico van onjuiste decoding kan opleveren. Om te zorgen dat dit meer robuust werkt bevat een QR code foutcorrectie, interne oriëntatie en uitlijning tekens. Op deze wijze maakt het niet uit of de QR code ondersteboven of gewikkeld rond een gebogen vlak gehouden wordt. (8)

### 4.4.1 Toepassing

Deze techniek zou in het Sioux Experience Center gebruikt kunnen worden met behulp van de camera op de tablet. Op elk product komt een zichtbare QR code die vervolgens door de tablet gescand kan worden.

### 4.4.2 Voor- en nadelen

Voordelen

- Betrouwbaar
- Makkelijk uitbreidbaar

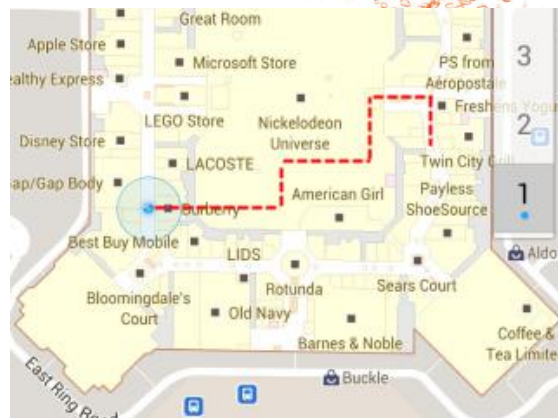
Nadelen

- Zichtbare QR code bij elk product

## 4.5 IPS + Kompas + Gyroscop

Als je binnen in een gebouw bent, is een data- en satellietverbinding vaak niet mogelijk, waardoor het haast onmogelijk is om nauwkeurig je positie binnen een gebouw te bepalen. Door gebruik te maken van radionetwerken, zoals wifi of bluetooth, is het mogelijk om je locatie binnen in een gebouw te bepalen.

De techniek die voor binnen navigatie gebruikt wordt is de zogenoemde 'Indoor Positioning System' (IPS) en bepaalt de locatie op basis van signaalsterkte die wordt ontvangen van nodes met een vaste plaats. Bijvoorbeeld een zender van een draadloos netwerk. Het is met deze techniek mogelijk om de 3D locatie vast te stellen, dus lengte, breedte en hoogte. Op deze manier kan er met een nauwkeurigheid van 30 centimeter de locatie worden bepaald, wat genoeg is om een informatiebalie, wc of schap in een winkel te vinden. (9) (10)



**Figuur 4.3. Navigatie in een winkelcentrum**

Alleen informatie over de positie is niet voldoende om een object te kunnen herkennen, hiervoor is ook de oriëntatie nodig. Om deze te bepalen zijn een kompas en gyroscop nodig. Met een kompas kan de richting naar het noorden bepaald worden, om zo de horizontale hoek van de gebruiker vast te kunnen stellen. Een gyroscop kan dankzij de draaiing van de aarde de verticale hoek bepalen.

Behalve gebruik te maken van radionetwerken voor IPS kan er ook gebruik worden gemaakt van het magnetische veld van de aarde. Bijna elke smartphone heeft een ingebouwd kompas om de horizontale richting te bepalen, deze sensor is gevoelig genoeg om de sterkte van het magnetische veld van de aarde te meten. Hierdoor kan met een nauwkeurigheid van minimaal 10 centimeter de positie worden bepaald. (11)

### 4.5.1 Toepassing

Deze techniek zou kunnen worden toegepast in het Sioux Experience Center door van elke product de driedimensionale locatie te bepalen. Vervolgens kan door de positie en oriëntatie van de gebruiker worden vastgesteld welke producten onder de kijkregio van de gebruiker vallen, hierdoor kunnen producten worden herkend.

### 4.5.2 Voor- en nadelen

Voordelen

- Geen extra hardware nodig

Nadelen

- De maximale nauwkeurigheid is 10 centimeter, dit is te weinig om producten te onderscheiden
- Bij herpositionering van een product moet deze opnieuw worden ingesteld



## 4.6 Kinect

Kinect is een accessoire bedoeld voor de Xbox 360 van Microsoft. Het is een zogenaamd "motion-control" systeem waarmee een speler, met behulp van het lichaam, de bestuurder wordt van de Xbox 360 console. (12)

De Kinect bevat een infrarode lichtbron die de omgeving belicht en waarbij de weerkaatsing wordt opgevangen door een sensor. Uit de vergelijking tussen het verzonden gecodeerde licht en de opgevangen weerkaatsing wordt een dieptebeeld van de omgeving gemaakt. Doordat een eigen infrarode lichtbron wordt gebruikt, kan het systeem ook in slecht verlichte ruimtes



**Figuur 4.4. Opbouw Kinect**

worden gebruikt. Verder is het systeem uitbereid met een tweede camera die niets bijdraagt aan het waarnemen van diepte, maar die simpelweg een kleurenopname maakt van wat zich voor het apparaat afspeelt. De vertaalde data, bestaande uit een beeld met diepte-informatie, kleurenbeeld, audio en zelfs een tijdcodering, wordt vervolgens via USB 2.0 uitgestuurd. (13)

Alle ingebouwde technologie van de Kinect samen maakt het mogelijk om zowel gebaren, 'spraak' als gezichten te herkennen.

### 4.6.1 Toepassing

Deze techniek zou in het Sioux Experience Center gebruikt kunnen door de Kinect te verbinden met het tablet device, die vervolgens door middel van het diepte- en kleurenbeeld meerdere producten kan herkennen.

Een andere toepassing zou het herkennen van de gebruiker zijn. Hiervoor zouden er meerder Kinects in de hoeken van de demo ruimte worden opgehangen. Om zo de positie van de gebruiker te bepalen. Deze kan dan naar een product wijzen en hier informatie over krijgen op de tablet.

### 4.6.2 Voor- en nadelen

#### Voordelen

- Geen aanpassingen nodig in het Sioux Experience Center
- Licht ongevoelig, mits gebruik gemaakt wordt van 3D identificatie

#### Nadelen

- Kinect als externe hardware nodig
- Veel processor kracht nodig om beelden te verwerken
- Indien herkenning van gebruikerspositie, bij herpositionering van een product moet deze opnieuw worden ingesteld

#### 4.6.3 Onzekerheden

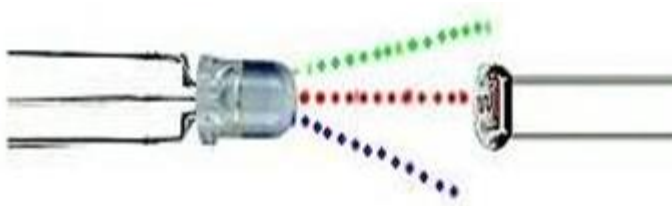
Als de Kinect gebruik wordt om producten te herkennen zijn er een aantal onzekerheden. De Kinect is ontworpen om mensen te herkennen, hoe goed is deze in het herkennen van een product? Kan deze onderscheid maken tussen bijna identieke vormen van producten zoals printplaten? Ook moet onderzocht worden hoe deze techniek reageert op verschillende aanzichten van een product. Tot slot of het wel mogelijk is om een Kinect aan te sluiten op een tablet device.

Indien er meerdere Kinects gebruikt worden om de positie van de gebruiker te bepalen is het onduidelijk of de nauwkeurig genoeg is.



## 4.7 Coded Light

Coded Light is een techniek waarbij gebruik gemaakt wordt van een infrarood zender en ontvanger. De zender stuurt eigen pulsjes uit op een vaste frequentie die ontvangen kunnen worden door de ontvanger, dit kan bijvoorbeeld een camera zijn. Omdat elke zender een eigen puls heeft kan de ontvanger verschillende zenders identificeren.



*Figuur 4.5. Communicatie zender en ontvanger*

### 4.7.1 Toepassing

Een denkbare toepassing voor het Sioux Experience Center met deze techniek is bij elk product een zender met een eigen puls. De tablet kan met behulp van de camera infrarood detecteren en onderscheiden, om zo verschillende producten te identificeren.

### 4.7.2 Voor- en nadelen

Voordelen

- Makkelijk uitbreidbaar

Nadelen

- Externe zichtbare hardware bij elk product

### 4.7.3 Onzekerheden

Een onzekerheid is de frequentie waarmee pulsjes gestuurd gaan worden door de zender. Deze moet snel genoeg zijn om binnen een redelijk tijd een product te kunnen identificeren, maar als dit te snel gaat voor de camera kan deze de pulsjes niet juist onderscheiden.

## 4.8 Voor- en nadelen

Bij het onderzoeken naar de verschillende mogelijke technieken voor objectherkenning zijn al voor- en nadelen opgesomd. In het overzicht hieronder worden alle voor- en nadelen vergeleken met de eisen van de klant. Hierin is duidelijk te zien dat niet alle technieken voldoen aan deze eisen.

	Zichtbaarheid	Betrouwbaar	Nauwkeurigheid	Afstand	Uitbreidbaar
Computer Vision	V	?	?	V	V
RFID/NFC	V	V	V	V	V
QR Code	X	V	V	V	V
IPS + Kompas + Gyroscop	V	?	X	V	X
Kinect	V	?	X/V	V	X/V
Coded Light	X	V	V	V	V

## 4.9 Conclusie

Een belangrijke eis van de product owner is dat het Sioux Experience Center niet aangepast mag worden, tenminste dit mag niet zichtbaar zijn. Hierdoor vallen QR Code en Coded Light af, omdat deze beide zichtbare hardware bij producten nodig hebben voor identificatie die niet weggewerkt kan worden.

Ook moet het systeem producten die 5 centimeter van elkaar liggen kunnen onderscheiden. Hierdoor is het gebruik van IPS met het kompas en de gyroscoop niet mogelijk omdat deze niet de nauwkeurigheid bevat die nodig is.

NFC is een afgeleide techniek van RFID, het voordeel van NFC is dat er tablets zijn die een reader standaard hebben ingebouwd. Hierdoor is er geen externe hardware nodig zoals bij RFID, die uiteindelijk dezelfde functionaliteit oplevert als NFC. Het nadeel van NFC is dat de scanafstand klein is, deze zit tegen het minimum aan van de eisen. Toch is er gekozen om NFC verder te onderzoeken omdat deze techniek aan alle eisen voldoet. In het vervolgonderzoek zal worden onderzocht of de scanafstand in de praktijk ook aan de eisen voldoet.

Tot slot zijn er nog de Kinect en Computer Vision, deze zijn allebei afgeleid van beeldherkenning technieken. Het verschil tussen deze technieken is dat de Kinect behalve een 2D aanzicht ook een 3D aanzicht tot zijn beschikking heeft. Het grote nadeel van de Kinect is dat deze nog aangesloten moet worden op de tablet. Daar tegenover kan voor Computer Vision de camera op de tablet gebruikt worden. Omdat het een eis is van de klant dat het systeem innovatief, uitdagend en futuristisch over moet komen is gekozen om Computer Vision verder te onderzoeken en de Kinect niet, omdat een grote Kinect achterop een tablet niet innovatief overkomt maar eerder onpraktisch.

Achteraf was er het idee om meerdere Kinects in de demo ruimte te hangen. Om vervolgens de gebruiker te herkennen die naar een product aan het wijzen is. Helaas kwam het idee voor deze oplossing te laat en is er geen tijd meer om deze te onderzoeken.

De overgebleven technieken die nader worden onderzocht zijn NFC en Computer Vision.

## 5 NFC

NFC is uit het vooronderzoek gekomen als een herkenningstechniek die verder onderzocht gaat worden. In dit hoofdstuk wordt deze techniek onderzocht in de praktijk. Op deze manier kan worden bepaald of de techniek voldoet aan de eisen van de klant.

### 5.1 Onderzoek

Met behulp van een smartphone of tablet met NFC ondersteuning worden tags uitgelezen. Dit gebeurt op verschillende manieren om zo te kunnen meten of aan de verschillende eisen wordt voldaan. De volgende vragen worden onderzocht:

- Hoelang duurt het herkennen van een tag?
- Wat is de minimale en maximale afstand om een tag te herkennen?
- Wat gebeurt er met twee tags naast elkaar?
  - Wat is de minimale afstand tussen tags om ze te kunnen onderscheiden?
- Hoe snel kan er met de reader langs een tag bewogen worden?
- Hoe vaak wordt een tag niet herkend?

Bovenstaande vragen worden eerst onderzocht met losse tags. Als deze uitkomsten acceptabel zijn worden dezelfde stappen herhaald met de tags bevestigd aan een object op een onzichtbare plek. Verder wordt er nog gekeken naar de hoeveelheid werk om een tag in te stellen.

### 5.2 Benodigheden

- Smartphone of tablet met NFC ondersteuning
- Twee tags
- Software om NFC uit te lezen
- Objecten om te herkennen

De objecten om te herkennen zijn representatief voor de producten in het Sioux Experience Center. Er is gekozen voor de volgende objecten:

- **Raspberry Pi**, deze representeert een printplaat. De tag wordt hierbij aan de onderkant geplaatst.
- **Desktop**, deze representeert de Phenom. De tag wordt hierbij aan de binnenkant van de behuizing geplaatst.
- **Plastic telefoon protector**, deze representeert een blikjesautomaat. De tag wordt aan de binnenkant van de protector geplaatst. Bij de blikjesautomaat zal dit in de binnenkant van de deur zijn, waardoor de tag en de reader alleen gescheiden zullen zijn door een plastic laag.

## 5.3 Stappenplan

1. Bestaande NFC applicatie installeren op device
2. Tags programmeren met uniek herkenbaar nummer, meet de tijd en het aantal stappen die hiervoor nodig zijn.
3. Herhaal elke onderstaande stap 10 keer en pak het gemiddelde
  - Houd het device tegen een tag, meet de tijd tot herkenning van de tag.
  - Beweeg het device richting de tag totdat deze herkent word, meet de afstand.
  - Zet 2 tags naast elkaar en houd het device tegen de tags, schrijf op wat er gebeurt.
  - Zet 2 tags naast elkaar en houd het device bij de linker tag. Maak de afstand groter totdat de rechter tag niet meer herkend word. Meet de afstand van de tags.
  - Beweeg in verschillende snelheden het device tegen een tag, schrijf op welke met maximale snelheid de tag herkent kan worden.
  - Beweeg het device langs de tag, doe dit binnen de maximale herkenningafstand en -snelheid. Noteer hoe vaak de tag herkent word.
4. Indien uitkomsten van stap 3 acceptabel, stap 3 herhalen maar dan met tags bevestigd aan de objecten op een onzichtbare plek

## 5.4 Resultaten

Bij het uitvoeren van dit onderzoek is er gebruik gemaakt van een HTC One X smartphone. Hierop is een NFC applicatie geïnstalleerd genaamd "NFC Task Launcher". De tags zijn eenvoudig geprogrammeerd met behulp van de applicatie. De tags hebben een werkfrequentie van 13,56 MHz en een capaciteit van 1024 bytes. De identificatie tekst van de eerste tag was "tag1" en bij de tweede was dit "tag2". Voor het programmeren van een tag behoefde alleen de gewenste tekst opgegeven te worden en het NFC device tegen de tag gehouden te worden voor ongeveer één seconde, daarna is de tag klaar voor gebruik.

Vervolgens zijn de tags op verschillende manieren gescand, de uitkomsten hiervan zijn zichtbaar in onderstaand overzicht.

Uitkomsten	Losse tag	Printplaat	Desktop	Plastic
<b>Minimale Tijd (sec)</b>	0,32	-	-	0,4
<b>Minimale Afstand (cm)</b>	0	-	-	0
<b>Maximale Afstand (cm)</b>	3,8	-	-	2,95
<b>Maximale snelheid (cm/s)</b>	12,5	-	-	13
<b>Minimale onderscheiding afstand (cm)</b>	4	-	-	3
<b>Betrouwbaarheid (%)</b>	90	-	-	70

In het overzicht is duidelijk zichtbaar dat de tag op geen mogelijkheid herkend wordt als deze bevestigd is aan de printplaat of aan de desktop. Ook is er duidelijke achteruitgang zichtbaar wanneer een tag aan een plastic object vast zit, de scanafstand neemt met 23% af en het duurt ook 25% langer tot herkenning. Als meerder tags dicht bij elkaar liggen worden deze beide herkend door de NFC reader.



## 5.5 Conclusie

Buiten het feit dat de tag bij een deel van de objecten niet meer herkend wordt is deze techniek nog steeds niet voldoende voor het Sioux Experience Center. De uitkomsten vergeleken met de eisen van de klant:

Eis	Losse tag	Printplaat	Desktop	Plastic
<b>Robuustheid/Betrouwbaarheid</b>	v	x	x	x
<b>Snelheid</b>	v	x	x	v
<b>Nauwkeurigheid</b>	v	x	x	v
<b>Afstand</b>	x	x	x	x
<b>Set objecten aanpassen</b>	v	v	v	v

De herkenafstand van deze techniek is te klein om gebruik te worden voor het Sioux Experience Center. Ook moeten veel verschillende soorten objecten herken kunnen worden, dat is met deze techniek niet mogelijk.



## 6 COMPUTER VISION

Er zijn veel verschillende algoritmes die gebruikt kunnen worden voor objectherkenning. Enkele van deze algoritmes worden met elkaar vergeleken om zo het algoritme te vinden dat het beste aansluit bij de gewenste toepassing.

### 6.1 Werking algoritmes

Een Computer Vision algoritme bestaat uit:

- Feature detectie
  - Detection
  - Description
- Matching

Feature detectie is het proces waarbij objecten in een afbeelding onderzocht worden op unieke kenmerken, zodanig dat de objecten op basis van deze kenmerken te detecteren zijn op andere afbeeldingen. Dit proces bestaat uit twee onderdelen, detection en description.

*Detection* is een operatie die 2D-locaties zoekt in een afbeelding die geometrisch stabiel zijn onder verschillende transformaties. Deze 2D-locaties worden in verschillende omstandigheden, zoals lichtintensiteit of aanzicht, vaak gevonden. De methode die gebruikt wordt voor het zoeken is per algoritme verschillend. De resultaten worden over het algemeen 'interest points' genoemd.

*Description* is het analyseren van een afbeelding op de posities van de interest points. Het resultaat is een 2D-vector van pixelinformatie. Deze informatie kan worden gebruikt om de interest points te classificeren op schaal- en rotatieafhankelijkheid. Het resultaat zijn alle interest points die als voldoende geclassificeerd zijn en worden keypoints genoemd.

Nadat de keypoints zijn berekend kan met behulp van *matching* overeenkomsten worden gezocht met een andere set van keypoints. Dit gebeurt door overeenkomsten te zoeken in de verhoudingsposities van keypoints. Het resultaat van matching zijn het aantal overeenkomsten tussen de twee sets van keypoints. Aan de hand van de hoeveelheid overeenkomsten kan worden bepaald of de afbeeldingen een bepaald voorwerp bevat. (14) (15) (16)

## 6.2 Overzicht algoritmes

Er zijn een veel verschillende algoritmes voor objectherkenning. In figuur 6.1 zijn er een aantal te zien met de tussenliggende relaties. Er zijn voor deze algoritmes gekozen omdat het de meest gebruikte algoritmes zijn. In het relatieoverzicht is te zien dat het SIFT algoritme de basis is voor veel algoritmes.

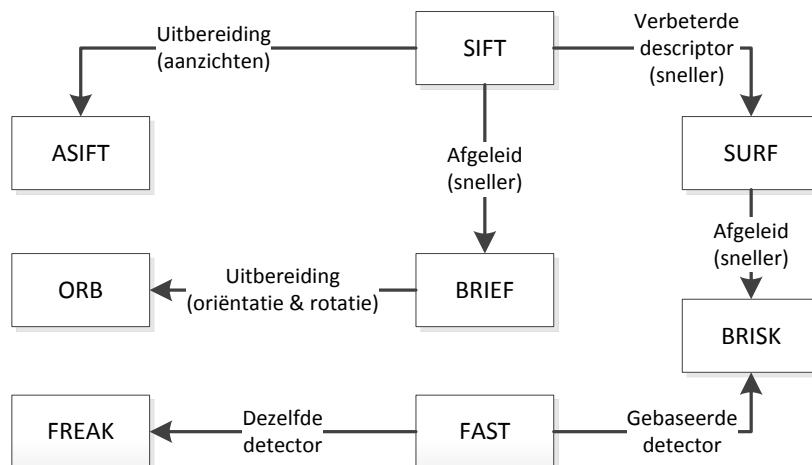
Niet alle algoritmes worden getest in dit onderzoek omdat dit teveel tijd kost. De algoritmes die onderzocht gaan worden zijn:

- SIFT (17) (18) (19) (20)
- SURF (21) (22)
- ORB (23)
- FAST (24) (25)
- FREAK (26)

Algoritmes die niet verder onderzocht worden:

- BRIEF (27)
- ASIFT (28) (29)
- BRISK (30)

Deze algoritmes zijn gekozen omdat ze uiteenlopend zijn op het relatie overzicht. Op deze manier worden verschillende soorten detectors en descriptors onderzocht. Ook zijn deze algoritmes direct beschikbaar in OpenCV, dit is handig bij de implementie.



**Figuur 6.1. Relaties tussen algoritmes**

## 6.3 Onderzoek

Met dit onderzoek wordt de bruikbaarheid van een algoritme bepaald. Door middel van productafbeeldingen (te herkennen objecten) en een variërend inputbeeld worden verschillende onderzoeken uitgevoerd. Het inputbeeld wordt vergeleken met verschillende productafbeeldingen om zo de beste overeenkomst te vinden. Op deze manier kan worden bepaald welk product zicht bevindt op het inputbeeld.

De volgende aspecten worden onderzocht per algoritme:

- Snelheid
- Gevoeligheid voor False positives
- Tolerantie tegen vervorming
  - Kijkhoek
  - Schaal
  - Rotatie
  - Lichtintensiteit
- Afbeelding kwaliteit

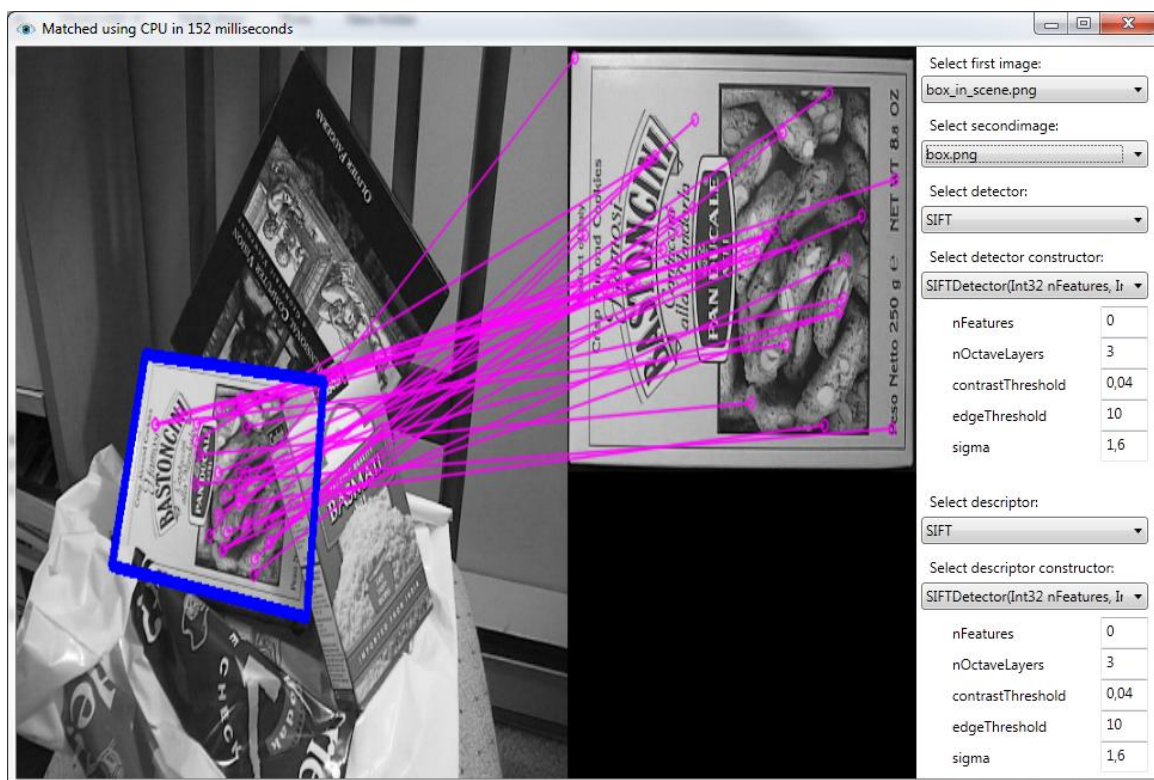
Als er een verkeerd product worden gematched bij een foto dan heet dit een *'false positive'*. Het is belangrijk dat een algoritme geen verkeerde producten vind bij foto's omdat de betrouwbaarheid van de applicatie een belangrijke eis is van de klant.

Het algoritme wordt op tolerantie tegen vervorming onderzocht. Voor elke case zijn een aantal input beelden vastgelegd, voor de kijkhoek zijn dit bijvoorbeeld een aantal afbeeldingen met verschillende kijkhoeken van het te herkennen object. Vervolgens worden de prestaties van het algoritme per case gemeten, hierdoor kunnen de algoritmen met elkaar vergeleken worden om zo het best aansluitende algoritme voor het Sioux Experience Center te bepalen.

Ook wordt er onderzocht welke fotokwaliteit en resolutie het beste gebruik kan worden voor het systeem.

## 6.4 Parameters

Elk algoritme is in te stellen met een eigen set parameters. Er zijn veel parameter combinaties mogelijk per algoritme. In verband met de tijd kunnen niet alle mogelijke parametersets van een algoritme gebruikt worden in de testapplicatie. Daarom is er met de hand de beste parameterset gezocht voor de algoritmes. Hiervoor is een parameter testapplicatie geschreven. Met deze applicatie kan met de hand parameters voor een algoritme ingesteld worden, om zo voor elk algoritme de beste parameterset te vinden. Deze parametersets worden gebruikt in de uiteindelijke testapplicatie.



Figuur 6.2. Printscreen van de parameter testapplicatie

Figuur 6.2 laat user interface van de parameter testapplicatie zien. Er zijn verschillende foto's die geselecteerd kunnen worden om de parameterinstellingen van een algoritme te testen. Op deze manier wordt een algoritme niet getest op één specifieke foto maar op meerder foto's in verschillende situaties. Ook is er keuze uit de verschillende algoritmes die beschreven staan in paragraaf 2. Elk algoritme heeft een eigen set aan parameters die aangepast kunnen worden. Als een parameter is aangepast worden de foto's opnieuw gematched door het algoritme met de nieuwe parameters. Op deze manier kan live worden gekeken of de verandering van de parameter het algoritme heeft verbeterd.



## 6.5 Ontwerp testapplicatie

Voor het vinden van het best passende algoritme voor de gewenste toepassing is een testapplicatie ontwikkeld. Deze applicatie vergelijkt met behulp van een algoritme verschillende testafbeeldingen met referentieafbeeldingen. Dit wordt voor elk algoritme gedaan. Op deze manier kunnen de algoritmes met elkaar vergeleken worden om zo het best presenterende algoritme te vinden.

Voor de testapplicatie worden er vijf verschillende testsets van producten gebruikt. Elke testset wordt één keer gemaakt met de camera van de tablet en één keer met een hogere resolutie camera. Een testset van een product bevat de volgende afbeeldingen:

- 3 verschillende **kijkhoeken** van het product
- 3 verschillende **afstanden** tot het product
- 3 verschillende **rotaties** van het product
- 3 verschillende **lichtintensiteiten** van het product

De opbouw van de 3 afbeeldingen is als volgt:

- **Normaal**, hierbij wordt de foto genomen vanuit normale omstandigheden.
- **Variabel**, hierbij wordt de foto afwijkend van de normale omstandigheden genomen, dit kan in de praktijk vaker voorkomen.
- **Extreem**, hierbij wordt de foto vanuit extreme omstandigheden genomen, in de praktijk komt dit bijna nooit voor.

Met de testsets kan onderzocht worden hoe goed een algoritme presteert in verschillende omstandigheden. Bij de keuze van de producten wordt rekening gehouden met producten die op elkaar lijken en bijzondere producten. Op deze manier wordt de robuustheid tegen false positives van een algoritme onderzocht.

Er wordt ook een productenreferentieset gemaakt. Hier worden de testafbeeldingen mee vergeleken op overeenkomsten om zo het product te vinden die volgens het algoritme het beste overeenkomt met de testafbeelding. Deze productenreferentieset bestaat uit alle producten uit het Sioux Experience Center. Er wordt één set gemaakt met de camera van de tablet en één set met een hogere resolutie camera.

Deze testapplicatie voert onderstaand stappenplan uit:

- Voor elk vision algoritme
  - Voor beide resolutie referentiesets
    - Voor elke product testset
      - Database creëren van referentieafbeelding(en)
      - Voor elke case (kijkhoek, schaal, etc.)
        - Voor elke afbeelding in de case
          - Match de afbeelding tegen de referentiedatabase en kijk of de juiste afbeelding herkend wordt.

In onderstaande flowchart is de doorloop van de testapplicatie weergegeven. Hierbij worden verschillende sets van afbeeldingen gebruikt. Deze zijn te onderscheiden in camera kwaliteit en resolutie. Ook worden er meerdere algoritmes gebruikt met verschillende parameters. De keuze hiervan is vastgelegd door de handmatig de beste parameters te zoeken met behulp van de parameter test applicatie.

### Referenties

Deze bevat keypoints van een referentieset die zijn berekend door het huidige algoritme.

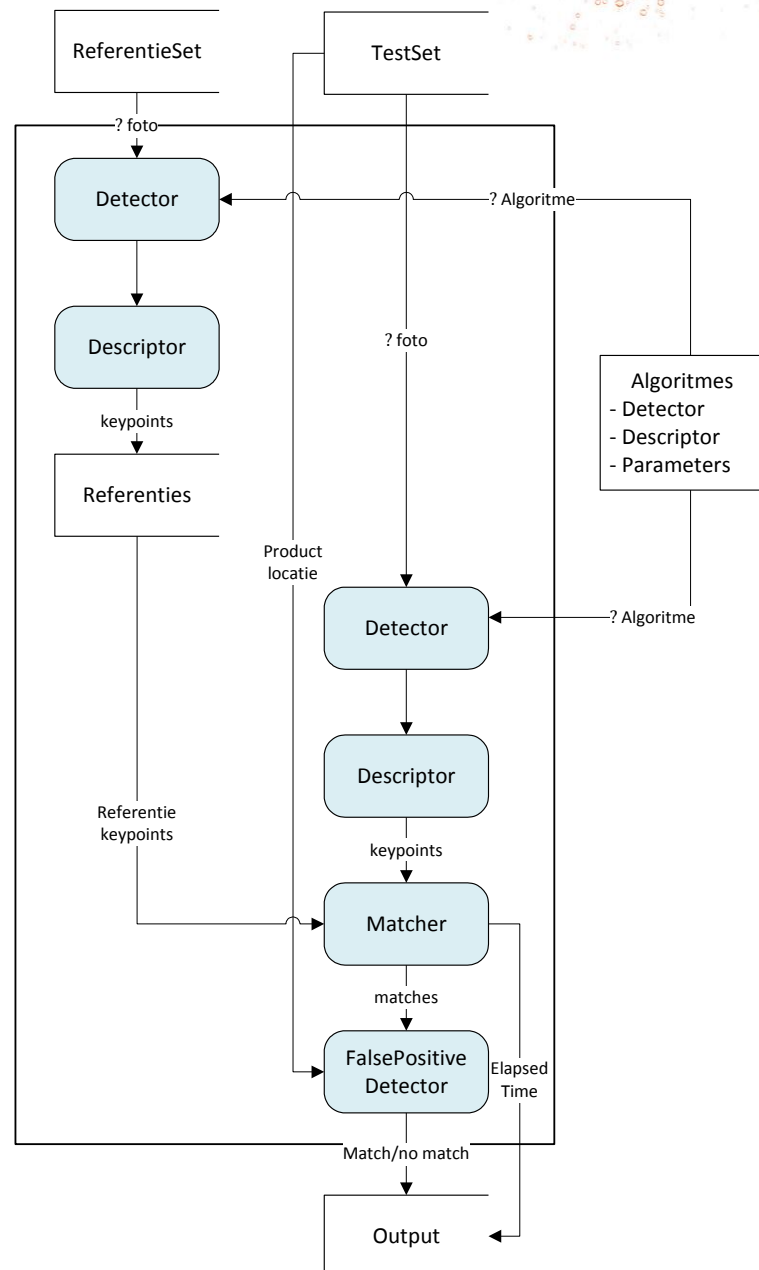
### Matcher

De matcher vergelijkt de berekende keypoints van de testafbeelding met de referenties. Hier komt een lijst met mogelijk matches uit.

### FalsePositive Detector

Deze controleert of het te herkennen product gevonden is door het algoritme. Voor elke afbeelding in de testset is het bekend wat de locatie van het product in de afbeelding is. Hierdoor kan eenvoudig gecontroleerd worden of de uitkomsten van het algoritme correct zijn. De FalsePositive Detector kan drie verschillende uitkomsten opleveren:

- **Succede**  
*Het product is gevonden op de juiste locatie en heeft de meeste matches op die locatie.*
- **FalsePositive**  
*Het product is niet (op de juist locatie) gevonden maar er is wel een ander product op deze locatie gevonden.*
- **NoMatch**  
*Er zijn geen matches gevonden op de aangegeven locatie.*

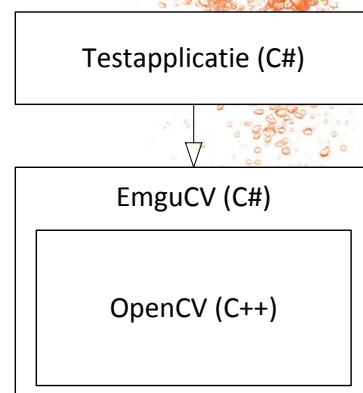


Figuur 6.3. Flowdiagram Testapplicatie



De testapplicatie maakt gebruik van de EmguCV bibliotheek. Dit is de OpenCV bibliotheek met een C# wrapper. Hierdoor is het mogelijk om OpenCV te gebruiken in een C# omgeving. OpenCV is een bibliotheek die programmeerfuncties bevat die vooral gericht zijn op Computer Vision, deze bevat bijvoorbeeld implementaties van de geselecteerde vision algoritmes. (31) (32)

De nieuwste versie van EmguCV bevat alleen GPU CUDA ondersteuning en geen CPU ondersteuning. De testcomputer waar de testapplicatie op wordt uitgevoerd bevat geen Nvidia video kaart, hierdoor werkt EmguCV er niet op. Om dit probleem te verhelpen is uitgezocht hoe EmguCV in elkaar zit en is er een nieuwe build gemaakt die wel op de CPU kan worden uitgevoerd.



**Figuur 6.4. Overview opbouw EmguCV**

## 6.6 Resultaten

In dit hoofdstuk worden de bevindingen uit de onderzoeksresultaten besproken. Hierbij zijn alleen de meest interessante grafieken zichtbaar, een overzicht van alle grafieken is te vinden in de bijlage. Met deze bevindingen kan er een conclusie worden getrokken welk algoritme het beste gebruikt kan worden.

In de test zijn sommige algoritmes meerdere keren gebruikt met een verschillende parametersets. Dit komt omdat er bij sommige algoritmes meerdere goede parametersets zijn gevonden in de parameter testapplicatie. Om deze te onderscheiden heeft elk algoritme een nummer. De parameterinstellingen van elk gebruikt algoritme is te vinden in de bijlage.

### 6.6.1 Match resultaten

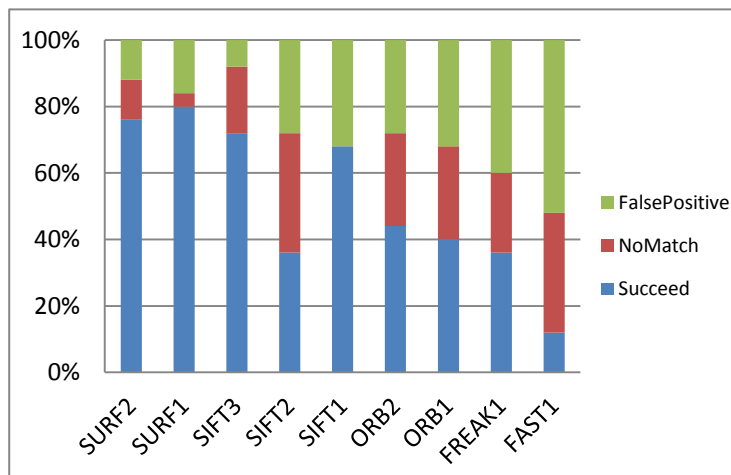
De FalsePositive Detector kan drie verschillende resultaten opleveren zoals staat beschreven in paragraaf 5. Voor elk algoritme is het percentage van elk resultaat berekend. In figuur 6.5 staan de percentages van elk algoritme. Uit deze match resultaten komen vier algoritmes die beduidend beter presteren:

- SURF2
- SURF1
- SIFT3
- SIFT1

Deze grafiek is berekend zonder extreme foto's, met extremen liggen de slagingspercentages lager. In figuur 6.5 heeft SIFT3 de minste false positives (8%) en heeft SURF1 het hoogste slagingspercentage met 80%.

De overige algoritmes hebben minder goed gepresteerd. Dit kan verschillende oorzaken hebben:

- Het kan zijn dat het algoritme niet ontworpen is voor dit doeleinde
- Het kan een slecht algoritme zijn.
- Het is ook mogelijk dat de parameters niet goed gekozen zijn voor de huidige toepassing.
  - Een oorzaak hiervan zou kunnen zijn dat de testfoto's van de parameter testapplicatie anders waren dan die van de testapplicatie. Helaas is er niet genoeg tijd om dit te onderzoeken.

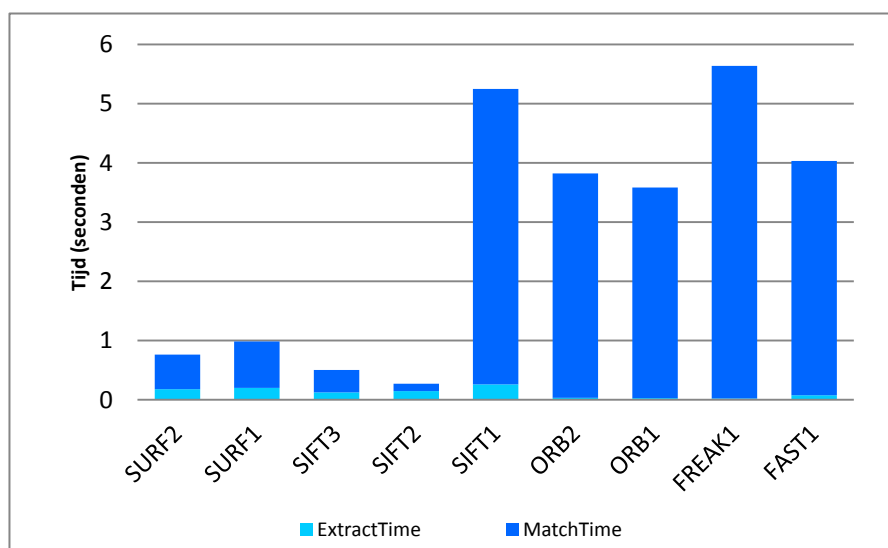


**Figuur 6.5. Resultaten per algoritme in %, zonder extremen**

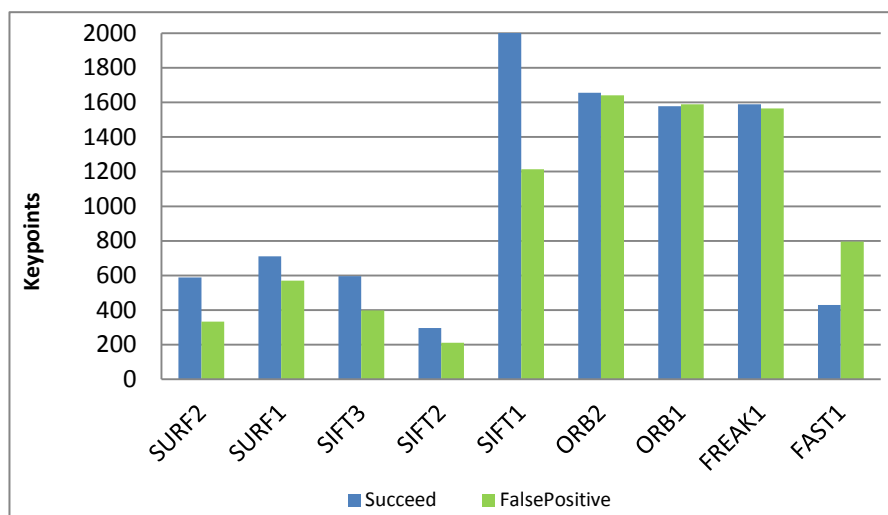
## 6.6.2 Tijd

Het verschil in tijd tussen de algoritmes is erg groot. Hier is een logisch verklaring voor: In figuur 6.6 is te zien dat de algoritmes die lang doen over het matchen erg snel zijn in het extraheren. Bij het extraheren blijven er een hoop keypoints over (figuur 6.7). Dit komt omdat er bij het wegfilteren een lage grens is om keypoints vast te stellen, dit levert snel veel keypoints op. Helaas zijn dit niet allemaal goede bruikbare keypoints, zoals uit de match resultaten blijkt. Niet alleen de testfoto heeft hierdoor veel keypoints, ook de referentiedatabase waarmee vergeleken wordt heeft voor elke referentiefoto veel keypoints. Omdat de vele keypoints van de testfoto vergeleken worden met een database die veel keypoints bevat kost dit veel tijd. Het is dus belangrijk dat een algoritme alleen de bruikbare keypoints uit een foto overhoudt na het extraheren. Dit zijn keypoints die in verschillende situaties (Licht, Schaal, Hoek, enz.) altijd gevonden worden. Omdat deze dan op elke foto van een product dezelfde keypoints opleveren en zo bij het matchen correcte overeenkomsten opleverd.

SURF en SIFT zijn de algoritmes die goed preseteren. De oorzaak dat de andere algoritmes lang doen over het verwerken van een foto zou wederom kunnen komen door een slechte parameter keuze.



**Figuur 6.6. Gemiddelde tijd per algoritme**



**Figuur 6.7. Gemiddelde keypoints per algoritme**

### 6.6.3 Cases

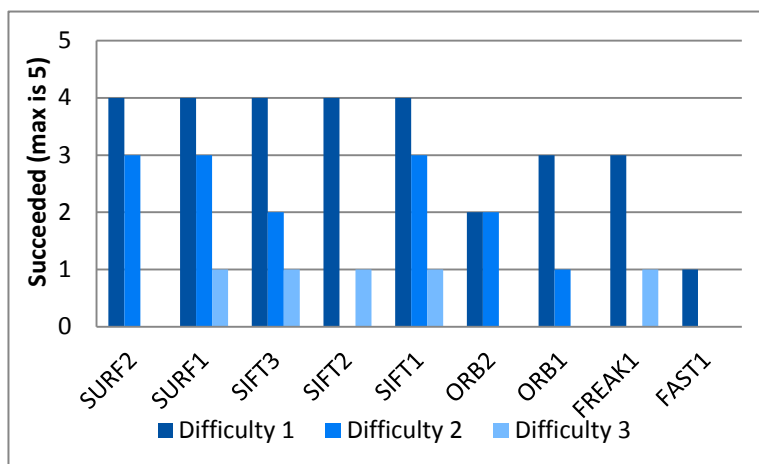
De testapplicatie heeft elk algoritme op vier verschillende cases onderzocht. Een case is een vervorming waar algoritmes in het algemeen moeite mee hebben. De vier onderzocht cases zijn:

- Hoek
- Schaal
- Lichtintensiteit
- Rotatie

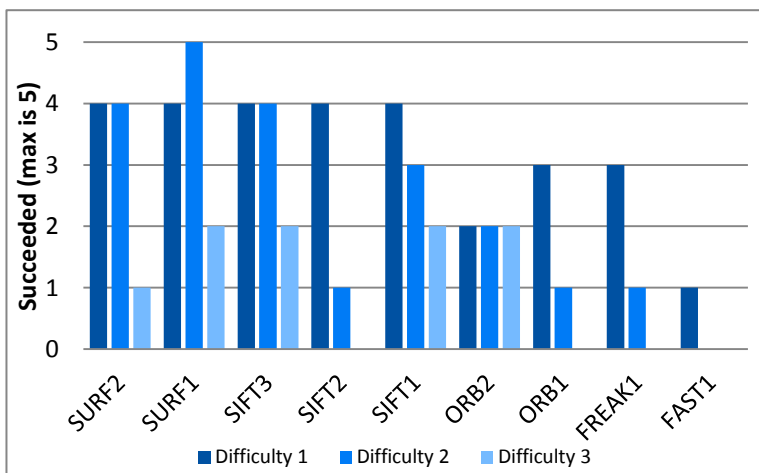
Uit de testapplicatie blijkt dat de algoritmes het meeste moeite hebben met hoeken. De resultaten van deze case zijn te vinden in figuur 6.8. Het gemiddeld slagingspercentage is bij deze case 50%.

De belangrijke case is schaal omdat deze vaak voorkomt in de praktijk. De gebruiker zal vaak van een afstand het product proberen te herkennen met de tablet. Zoals is te zien in figuur 6.9 heeft SURF1 bij deze case het beste resultaat met 11 van de 15 geslaagd (73%), waarbij dit zonder de extremen 9 van de 10 is (90%).

De algoritmes SIFT en SURF hebben goed gepresteerd bij de cases licht en rotatie. Zoals bij alle cases hebben ze vooral moeite met extremen. De resultaten hiervan zijn te vinden in de bijlage.



Figuur 6.9. Resultaten hoek case



Figuur 6.8. Resultaten schaal case

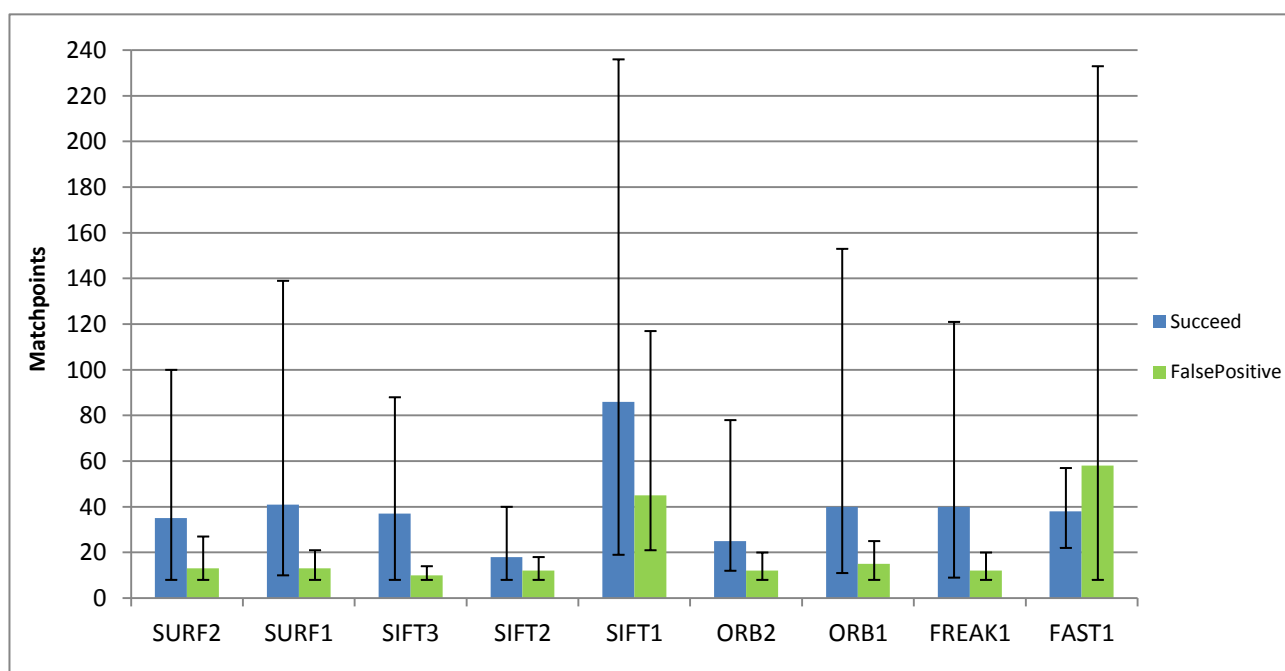
#### 6.6.4 Minimale grens

Matchpoints zijn het aantal matches dat is gevonden tussen één foto uit referentiedatabase en de testfoto. Door een minimale grens van matchpoints te trekken kan het aantal false positives worden verminderd. Dit is belangrijk omdat een algoritme betrouwbaar moet zijn en alleen correcte producten moet herkennen. Het aantal false positives wordt door deze grens verminderd omdat er een minimaal aantal matches is waaraan voldaan moet worden voor een match goedgekeurd is. In figuur 6.10 is te zien dat bij de meeste algoritmes een duidelijke grens is te trekken in het aantal matchpoints tussen geslaagde en false positive resultaten.

Bij SIFT3 is deze grens het beste te trekken. Bij de false positive resultaten is het maximaal gevonden matchpoints 14. Het gemiddelde matchpoints voor de geslaagde resultaten is 37. Dit is een groot verschil. We kiezen ervoor om de matchpoint grens te stellen op 20 omdat er dan een hoop minder false positives zijn. De grens ligt niet direct op de maximale matchpoint waarde van false positives omdat er een stuk speling moet zijn. Op deze manier kan de matchpoint waarde in de praktijk nog iets hoger zijn. Helaas wordt er ook 27% van de geslaagde resultaten niet meer gevonden omdat deze ook onder de grens zitten. 43% hiervan zijn extreme testfoto's.

Bij SURF1 moet de grens hoger gezet worden omdat de maximale waarde van de false positives 21 is. Wel ligt het gemiddelde matchpoints van geslaagde hier hoger, dit is 41. Ook is het maximaal aantal matchpoints van de geslaagde hoger dan bij SIFT3. Als de matchpoints grens hier op 30 wordt gezet word 46% van de geslaagde resultaten niet meer gevonden. 38% hiervan zijn extreme testfoto's.

Het percentage dat niet meer gevonden wordt is bij SURF1 hoger dan bij SIFT3. Ook is er bij SIFT3 beter een grens te trekken. SIFT3 is dus het beste algoritme om false positives bij te verminderen.



**Figuur 6.10. Gemiddeld aantal matches per resultaat, Met min en max aanduiding**



### 6.6.5 Resolutie

We hebben twee verschillende resoluties onderzocht, een lage resolutie van 0,3 megapixels en een hoge resolutie van 6 megapixels. Verrassend genoeg is het verschil in de resultaten niet groot, de lage resolutie heeft iets minder false positives dan de hoge. Wel valt het op dat de lage resolutie aanzienlijk beter presteert dan de hoge resolutie wanneer de extreme testfoto's worden weggelaten, deze heeft dan 44% minder false positives en 29% meer geslaagd. Het verwerken van de hoge resolutie testfoto's duurt ongeveer 34 keer langer dan de lage resolutie testfoto's. Er zal gebruik gemaakt gaan worden van een lage resolutie omdat deze sneller te verwerken is en hetzelfde resultaat oplevert.

### 6.6.6 Camera kwaliteit

Om te onderzoeken of verschil in camerakwaliteit andere resultaten oplevert hebben we twee camerakwaliteiten gebruikt, de huidige tablet camera (Asus Transformer Pad TF101) en de camera van een nieuwere telefoon (HTC One S). Beide fotosets zijn verkleint naar een resolutie van 0,3 megapixels. Dit is gedaan omdat de camera kwaliteit wordt onderzocht en niet resolutie verschil.

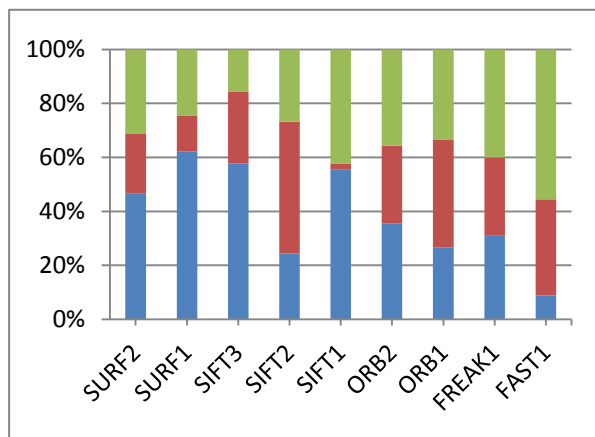
De resultaten verschillen niet veel. De camera van de telefoon presteert overal net iets beter. 10% meer geslaagd, 10% minder false positives en 10% minder geen matches. De camera van de huidige tablet levert iets minder goede resultaten op dan de camera van een nieuwer device. Maar het is niet direct nodig om een nieuw device aan te schaffen, de prestaties zijn goed genoeg.

### 6.6.7 Uitsnijden

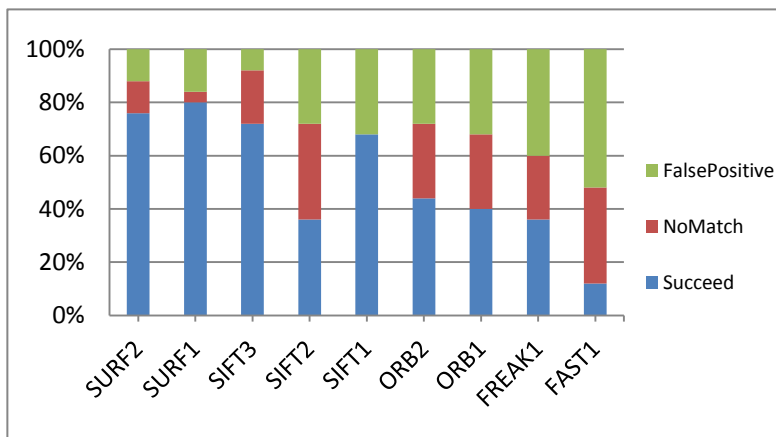
We hebben onderzocht of het een voordeel is om de referentie foto's uit te snijden, waarbij de omgeving van het product verwijderd wordt. Uit de resultaten blijkt dat dit averechts effect heeft, er zijn minder resultaten geslaagd. Een mogelijke verklaring hiervoor is dat de omgeving van een product ook gebruikt wordt voor herkenning.

### 6.6.8 Extreem

In figuur 6.11 en 6.12 is te zien dat alle algoritmes de meeste moeite hebben met de extreme testfoto's. Dit is ook logisch, want in deze testfoto's is het object in een extreme situatie weergegeven waardoor het veel verschilt met de referentiefoto. Gelukkig zullen deze situaties in de praktijk niet vaak voorkomen.



Figuur 6.11. Met extremen



Figuur 6.12. Zonder extremen

### 6.6.9 Producten

Niet alle producten zijn even makkelijk te herkennen door de algoritmes. Dit heeft vooral te maken met het aantal bruikbare keypoints dat een algoritme kan vinden bij een product. Hieronder staan twee afbeeldingen, een makkelijk te herkennen product (figuur 6.14) en een moeilijk te herkennen product (figuur 6.13).

Het makkelijk te herkennen product (cola-automaat) heeft veel contrastverschillen. Hierdoor kan een algoritme veel bruikbare keypoints vinden. Het moeilijk te herkennen product (Alice router) heeft weinig contrastverschillen en levert niet veel (bruikbare) keypoints op. Ook verschillen de locaties van keypoints vaak bij dit product, waardoor het ook minder vaak herkend wordt. Het moeilijk te herkennen product levert gemiddeld rond de 35 keypoints op. Bij makkelijk te herkennen product is dit gemiddeld rond de 600. Zie de grafieken in de bijlage voor alle productresultaten.

Een product met veel details is niet direct een makkelijk te herkennen product voor een algoritme. Een printplaat met veel detail levert rond de 800 keypoints op, hierin zitten veel kleine details. Deze kleine details zijn op een foto niet goed zichtbaar wanneer bijvoorbeeld de schaal of lichtintensiteit verandert. Er kan dus een minimale en maximale grens gesteld worden om te bepalen of een product foto goed herkenbaar is. Heeft een product minder dan 100 keypoints, of meer dan 750, dan is deze slecht herkenbaar.



**Figuur 6.13. Cola automaat, veel contrast**  
**Makkelijk te herkennen**



**Figuur 6.14. Router, weining contrast**  
**Moeilijk te herkennen**

## 6.7 Conclusie

Een lage resolutie presteert beter dan een hoge resolutie. Dit scheelt ook veel in de rekentijd per foto. Een nieuwe camera levert iets betere resultaten op dan die van de huidige tablet. Het uitsnijden van de referentie foto's heeft een averechts effect.

SURF1 en SIFT3 zijn de beste algoritmes om te gebruiken voor de gewenste applicatie. Beide algoritmes hebben eigen sterke punten, bij SURF1 is dit het hoge slagingspercentage en bij SIFT3 is dit het lage aantal false positives. Van deze twee algoritmes is SIFT3 het best bruikbare algoritme voor de gewenste applicatie omdat betrouwbaarheid een belangrijke requirement is. Dit algoritme levert minder geslaagde resultaten op dan SURF1. Maar dit is minder belangrijk dan geen foutieve resultaten.

SIFT3 voldoet niet aan alle eisen van de klant. De betrouwbaarheid van 90% is niet gehaald. Zelfs zonder de extreme waarden is het slagingspercentage slechts 72%. Het slagingspercentage kan niet eenvoudig hoger gemaakt worden, het aantal false positives kan wel flink naar beneden worden gebracht. Dit kan door het minimaal aantal matches te verhogen naar 20 voordat een match goedgekeurd wordt. Hierdoor is het algoritme een stuk betrouwbaarder qua foutieve matches.

Aan de rest van de eisen voldoet het algoritme wel. Het herkent een product binnen een seconde, in de praktijk wordt dit een paar keer herhaald in verband met betrouwbaarheid. Het algoritme kan producten herkennen tussen de vereiste afstand. Voor het toevoegen van een nieuw product om te herkennen is slecht één foto nodig.

Eis	Vision
Robuustheid/Betrouwbaarheid	x
Snelheid	v
Nauwkeurigheid	v
Afstand	v
Set objecten aanpassen	v

Er is een verschil in herkenbaarheid per product. Als het algoritme minder dan 100 keypoints heeft gevonden bij een referentie foto kan deze gemarkeerd worden als moeilijk herkenbaar.

## 7 HERKENNINGSAPPLICATIE

In dit hoofdstuk wordt in gegaan op de ontwikkeling van het uiteindelijke systeem. Als eerste wordt het complete ontwerp van het systeem beschreven. Vervolgens wordt er ingegaan op de ontwikkeling van het systeem.

### 7.1 Geleerd bij onderzoek

Van alle herkenningstechnieken die onderzocht zijn blijkt dat NFC en Computer Vision volgens informatie uit externe bronnen een oplossing zouden kunnen bieden voor de gewenste toepassing. Voor beide technieken is een onderzoek opgesteld om te onderzoeken of deze in de praktijk ook een oplossing biedt. Uit de resultaten van deze onderzoeken is gebleken:

- Het bereik van is NFC te klein
- NFC tags die onzichtbaar weggewerkt zijn kunnen niet altijd gescand worden
- Computer Vision biedt een uitkomst voor de gewenste toepassing.

SIFT3 is het vision algoritme dat het beste presteert in het vision onderzoek ten aanzien van de eisen van de klant. De camerakwaliteit van de huidige tablet is goed genoeg om te gebruiken voor vision. De resolutie van de foto's wordt verkleind naar 0,3 megapixels (640 x 480) om ervoor te zorgen dat het verwerken van foto's snel genoeg gaat. Ook zorgt dit voor een betere betrouwbaarheid in niet extreme situaties.

Om te zorgen dat het detectieresultaat betrouwbaar is worden de laatste 5 resultaten bewaard. Een match telt pas als in 3 van de 5 meest recente resultaten hetzelfde product herkend wordt. Een resultaat wordt maximaal 20 seconden bewaard.

Er is een Android applicatie geschreven met het SIFT algoritme geïmplementeerd. Het extracten van een foto duurt 1,5 seconden. Voor de totale match tijd komt dit neer op 6 seconden. Dit is 12 keer langzamer dan een desktop PC en is onacceptabel langzaam. Daarom is er gekozen om gebruik te maken van een externe server die het vision algoritme uitvoert.

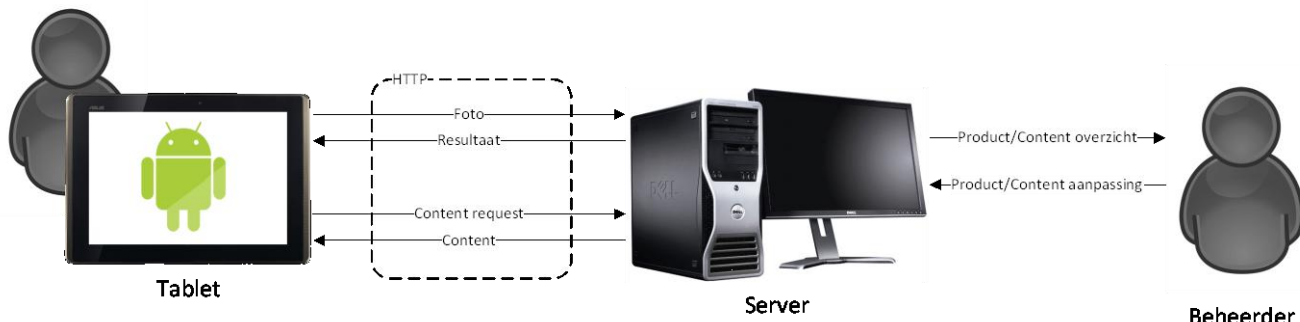
### 7.2 Ontwerp

Voor het implementeren is een ontwerp van het uiteindelijke systeem gemaakt. In deze sectie wordt als het architectuur overzicht van het complete systeem beschreven. Vervolgens wordt er dieper ingegaan op elk individueel component.



### 7.2.1 Architectuur

In de architectuur worden de verschillende systemen die gebruikt gaan worden beschreven. Er wordt per systeem bepaald welke verantwoordelijkheden deze heeft en hoe deze met elkaar communiceren. In figuur 7.1 staat de architectuur van het systeem met de verschillende apparaten die gebruikt gaat worden.



**Figuur 7.1: Architectuuroverzicht**

Omdat het verwerken en matchen van een foto op de tablet te traag is wordt er gebruik gemaakt van een externe server voor de beeldherkenning. Bovendien is het systeem zo makkelijk uitbreidbaar, omdat een andere tablet eenvoudig verbonden kan worden met de server en gebruik kan maken van dezelfde referentiedatabase.

De server heeft een passieve functie en is toestandloos. Deze krijgt aanvragen van de tablet en reageert hier op. De tablet houdt zelf zijn eigen toestand bij. Op deze manier wordt de serverapplicatie niet gecompliceerd als er met meerdere devices wordt gecommuniceerd.

Het protocol waarmee gecommuniceerd wordt tussen de server en de tablet is HTTP omdat de server op deze manier eenvoudig requests van verschillende devices tegelijk aan kan. Het initiatief komt van de tablet waarbij de server reageert.

Op de server staat een beheerapplicatie waarmee de beheerder via de user interface producten met bijbehorende content kan toevoegen of aanpassen. De computer die de functionaliteit als server gaat invullen is de computer in de demo ruimte. Door de beheerapplicatie op de server te zetten kan er direct in de demo ruimte producten worden aangepast en getest. De referentie productfoto's en content worden op de server bijgehouden. Zo is er een centraal punt waar alle data wordt bijgehouden.

De tablet stuurt een live foto op, de server geeft als respons het product dat de beste overeenkomst heeft. Zodra de tablet genoeg zekerheid heeft over het object dat herkend is kan deze bijbehorende content opvragen bij de server. Bijvoorbeeld als hetzelfde object meerder malen achter elkaar herkend is.

We maken het mogelijk om met de tablet nieuwe product foto's toe te voegen aan de server. De tablet applicatie heeft dus twee functies, product herkennen en nieuwe product foto's toevoegen. De functionaliteit voor het toevoegen van nieuwe product foto's is voor het beheer van de herkenningssystemen. Deze functionaliteit ligt bij de tablet omdat de beheerder dan eenvoudig nieuwe producten kan toevoegen en niet onnodig foto's hoeft over te zetten tussen verschillende systemen.

Als er een productfoto met de hand is genomen op de tablet wordt deze verstuurd naar de server. Deze komt in een queue die alle nieuwe productfoto's bevat. Vervolgens kan de beheerder een foto uit de queue toewijzen aan een nieuw product of de foto van een bestaande product vernieuwen.



## 7.2.2 Protocol

De server kan benaderd worden met een http request. Een http request naar de server heeft de volgende vorm:

*http://<IP>:<Poort>/<Functie>?<Parameter>=<Waarde>&<Parameter>=<Waarde>*

De server ondersteund de volgende POST functies:

	Detecteren op een foto	Foto toevoegen aan de queue
<b>Functie</b>	detectPhoto	addPhoto
<b>Type</b>	POST	POST
<b>Parameters</b>	-	-
<b>Body</b>	Binaire JPEG	Binaire JPEG
<b>Antwoord</b>	String met match product, leeg als geen overeenkomst	String met "SUCCES" als geslaagd, anders leeg

Met deze POST functies kan de client een foto laten matchen door de server of een foto toevoegen aan de queue voor het productbeheer. De body wordt op server direct omgezet naar een JPEG image en gebruikt. Om het netwerkverkeer en de matchtijd laag te houden moet de client er zelf voor zorgen dat de foto is gedownscaled.

De server ondersteund de volgende GET functies:

	Overzicht product content aanvragen	Content van een product aanvragen
<b>Functie</b>	getProduct	getContent
<b>Type</b>	GET	GET
<b>Parameters</b>	Product (Productnaam)	product (Productnaam)& content (Bestandsnaam)
<b>Antwoord</b>	product XML	Binaire content file

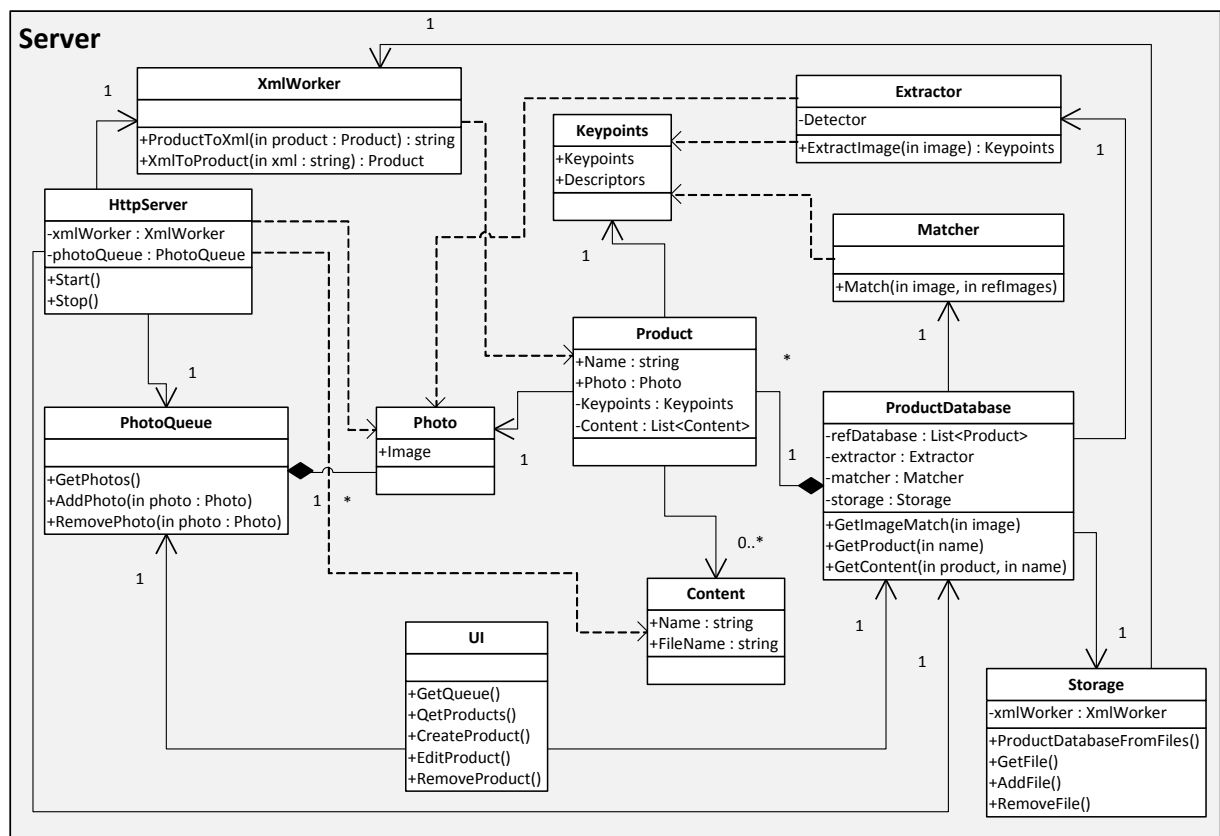
Met deze GET functies kan de client informatie krijgen over producten en bijbehorende content downloaden. De opbouw van het product XML antwoord heeft de volgende structuur:

```
<Product name="productNaam">
  <Content>
    <Name>Product foto</Name>
    <FileName>content1.jpg</FileName>
  </Content>
  <Content>
    <Name>Product specs </Name>
    <FileName>content2.pdf</FileName>
  </Content>
</Product>
```

## 7.2.3 Server

De belangrijkste functionaliteiten van de server applicatie zijn het matchen en productbeheer. Ook moet deze kunnen communiceren met client applicaties. De hoofdfunctionaliteiten zijn zoveel mogelijk uit elkaar getrokken in eigen klassen. Ook zijn sequentie diagrammen gemaakt om het gedrag van de applicatie vast te leggen.

### 7.2.3.1 Decompositie



Figuur 7.2: UML diagram server

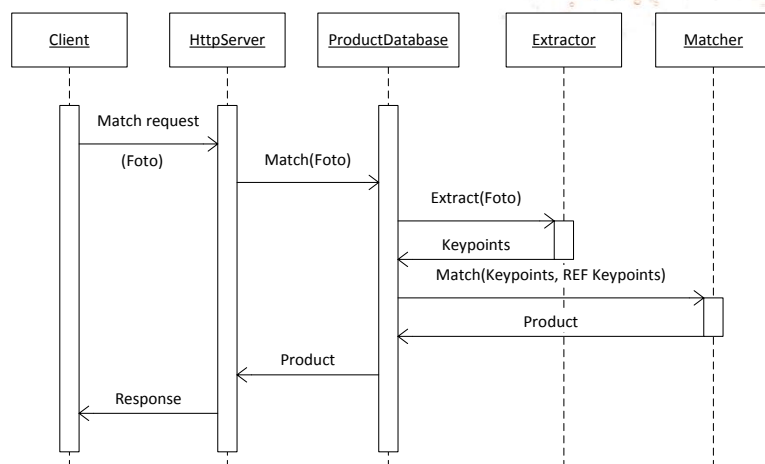
In figuur 7.2 staat het UML ontwerp van de server. Hierin zijn twee externe interfaces, de HttpServer en de UI. Bij de HttpServer zijn dit requests van de client. De HttpServer zorgt ervoor dat deze requests worden uitgevoerd en geeft de resultaten hiervan terug. De UI handelt acties van de beheerder af zoals het toevoegen van nieuwe producten.

De ProductDatabase klasse heeft twee functionaliteiten, het beheren van de producten met bijbehorende content en het matchen van een foto met de referentiedatabase. De HttpServer maakt gebruik van de ProductDatabase voor het matchen van een foto. Om te zorgen dat de server niet overbelast raakt met teveel parallele matches handelt deze maar een paar tegelijk uit. Dit aantal is afhankelijk van het aantal CPU's dat beschikbaar is. Als er een match request binnenkomt terwijl de ProductDatabase aan het maximale aantal parallele matches zit dan zal deze geblokkeerd worden tot er weer plek is om te matchen. Voor het matchen maakt de ProductDatabase gebruik van de Extractor en Matcher klassen. Bij het beheren van de producten wordt de product foto en bijbehorende content op de harde schijf opgeslagen. Voor het opslaan en ophalen van files wordt er gebruik gemaakt van de Storage klasse. Dit zorgt ervoor dat het aanleveren en opslaan van files makkelijk uitbreidbaar is naar een andere locatie.

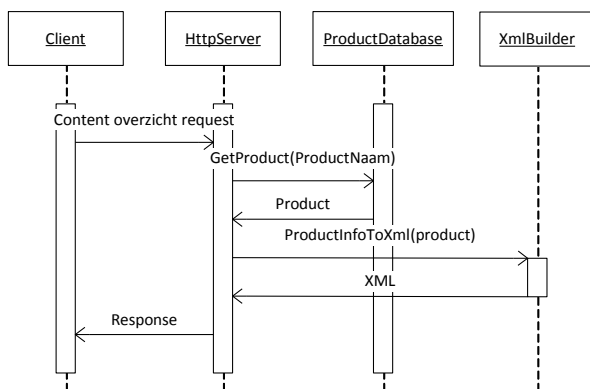
Als de client een aanvraag doet voor een overzicht van alle content bij een product wordt hierop geantwoord in het XML formaat. Hier is voor gekozen omdat dit leesbaar is als tekst en het platformonafhankelijk is. De HttpServer maakt hiervoor gebruik van de XmlWorker klasse.

### 7.2.3.2 Applicatie flow bij matching

In figuur 7.3 is het sequentie diagram voor het matchen van een foto te zien. Voor foto matching stuurt de client een live foto op naar de server. De HttpServer converteert de binnengekomen data naar een bitmap. Deze bitmap wordt naar de ProductDatabase gestuurd voor matching. Deze zet de foto om naar keypoints en laat deze matchen met de referentiedatabase. Het resultaat hiervan wordt teruggestuurd naar de client.



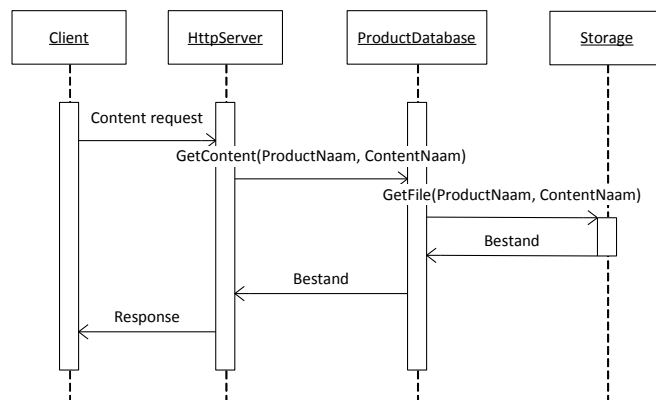
**Figuur 7.3: Sequentie diagram matching**



**Figuur 7.4: Sequentie diagram content overzicht**

De client kan een content overzicht opvragen van een product. Het sequentie diagram hiervan is te zien in figuur 7.4. Het product wordt opgevraagd bij de ProductDatabase. Dit product wordt vervolgens geconverteerd naar xml. De resulterende xml wordt teruggestuurd naar de client.

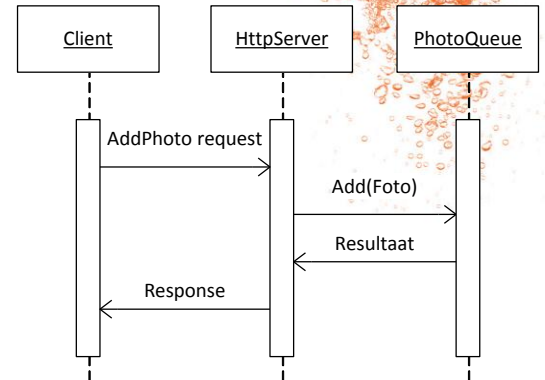
Nadat de client een overzicht heeft van de mogelijke content bij een product kan deze specifieke content aanvragen. In figuur 7.5 is het sequentie diagram voor het opvragen van een content file te zien. Hiervoor stuurt de client de naam van het product en de content filenaam op. De ProductDatabase haalt de file op van de harde schijf. Deze file wordt vervolgens opgestuurd naar de client.



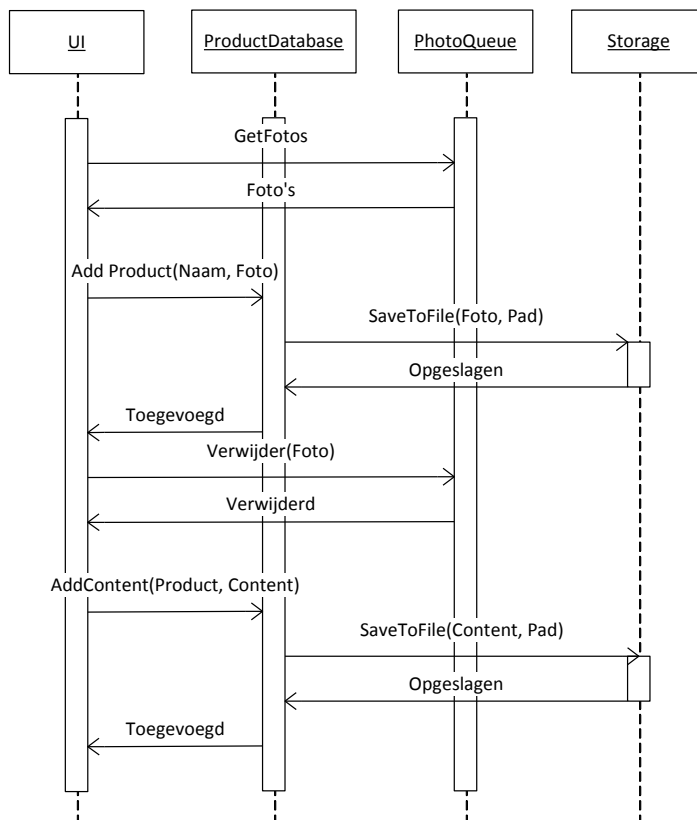
**Figuur 7.5: Sequentie diagram content**

### 7.2.3.3 Applicatie flow bij beheer

Om een nieuw product toe te voegen aan het systeem is content en een productfoto nodig. In figuur 7.6 is het sequentie diagram voor het toevoegen van een foto aan de queue te zien. De productfoto wordt genomen vanaf de client. Deze stuurt de foto naar de server. De HttpServer zorgt ervoor dat de foto wordt toegevoegd aan de PhotoQueue.



Figuur 7.6: Sequentie diagram foto toevoegen



Figuur 7.7: Sequentie diagram product toevoegen

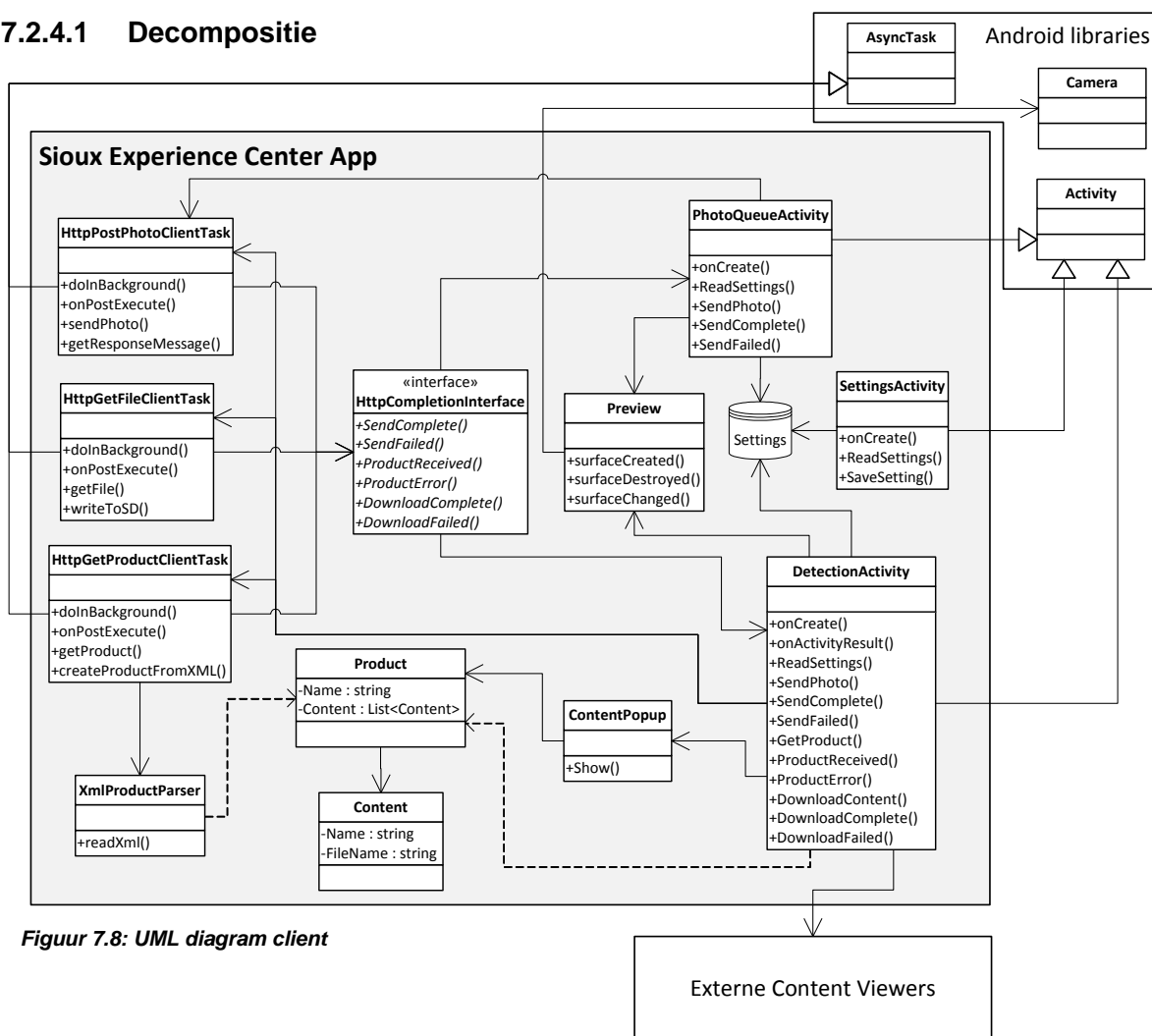
Vervolgens kan de beheerder via de UI de foto uit de PhotoQueue halen en aan een nieuw product toewijzen. Dit is te zien in figuur 7.7. Als het product is aangemaakt en de foto op de juiste plek op de harde schijf is opgeslagen wordt deze uit de PhotoQueue verwijderd.

De beheerder kan content toewijzen aan het product. Deze wordt ook opgeslagen op de harde schijf. De content wordt door de beheerder zelf aangeleverd, bijvoorbeeld met een usb-stick. De aangeleverde content kan in de beheerapplicatie geselecteerd worden. Deze zorgt ervoor dat de content gekopieerd wordt naar de database.

## 7.2.4 Tablet

De tablet applicatie kan producten herkennen door foto's naar de server te sturen voor matching. Ook kan de applicatie foto sturen naar de server voor het producten beheer. De grootste functionaliteit is dus het versturen van foto's. Verder kan de applicatie nog content opvragen en aan de gebruiker laten zien, zoals een PowerPoint presentatie.

### 7.2.4.1 Decompositie



Figuur 7.8: UML diagram client

In figuur 7.8 staat het UML ontwerp van de client applicatie. Deze applicatie communiceert via een http verbinding met de server. Elke http request heeft een eigen klasse omdat de objecten die hiermee verstuurd en ontvangen worden verschillen. De enige uitzondering hierop is het versturen van een match foto en het uploaden van een foto voor de queue. Deze sturen beide een foto en krijgen een tekstueel antwoord terug van de server. Daarom wordt voor deze requests dezelfde klasse gebruikt. Elke request klasse is een taak die op de achtergrond uitgevoerd wordt zodat de user interface hier geen last van heeft. Op het moment dat de taak voltooid is geeft de request klasse antwoord terug via de HttpCompletion Interface.



Een Activity is een representatie van de presentatielaag van een Android applicatie, bijvoorbeeld een scherm dat de gebruiker ziet. De client applicatie heeft de volgende activiteiten:

- DetectionActivity
- PhotoQueueActivity
- SettingsActivity

De DetectionActivity en PhotoQueueActivity maken gebruik van de camera. Dit gebeurt met de Preview klasse. Deze klasse is een UI element die het camerabeeld verbindt aan zijn eigen UI. Bij het afsluiten zorgt deze ervoor dat de verbinding met de camera netjes wordt afgesloten.

Voor de instellingen is een eigen Activity. In deze Activity kan bijvoorbeeld het IP adres van de server worden ingesteld. Deze instellingen worden in het gedeelde applicatie geheugen weggeschreven en zijn beschikbaar voor de andere activiteiten.

Voor het laten zien van content wordt gebruik gemaakt van externe viewers. Hiervoor worden een Intent gebruikt. Een Intent is een bericht die het mogelijk maakt voor Android componenten om functionaliteit te vragen van andere componenten in het Android-systeem. De applicatie zorgt er zelf voor dat de content wordt gedownload en opgeslagen. Na het downloaden wordt er met een Intent een oproep gedaan naar applicaties die de content kunnen laten zien. Als er een applicatie beschikbaar is wordt de content hierop gevisualiseerd.

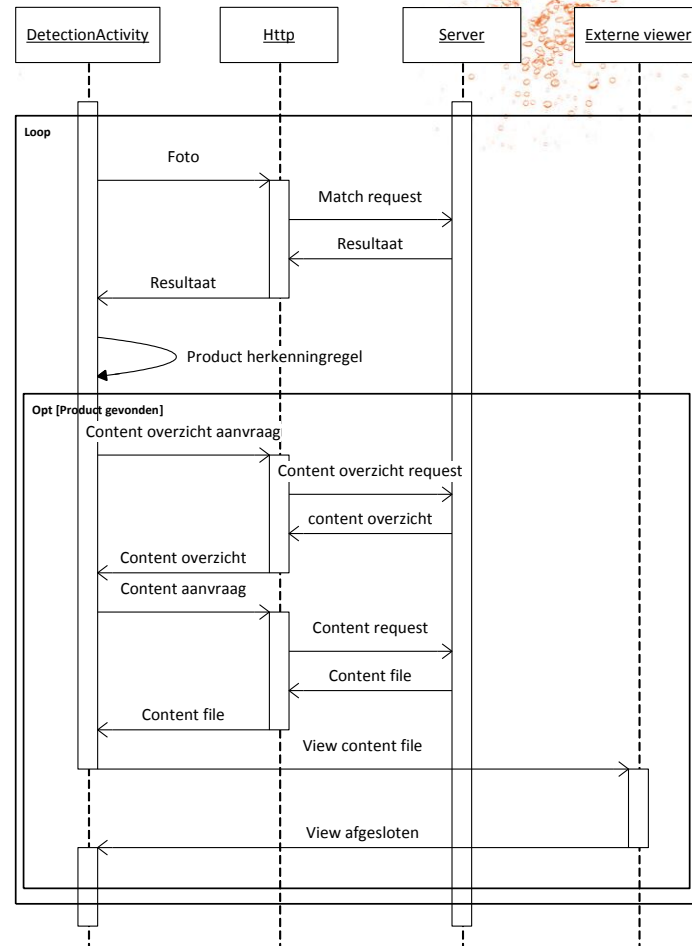
### 7.2.4.2 Applicatie flow bij herkenning

In figuur 7.9 staat het sequentiediagram voor het herkennen van een product. De DetectionActivity maakt een foto en stuurt deze op naar de server voor matching. De laatste 5 resultaten worden bewaart met een maximale bewaartijd van 20 seconden. Als van deze 5 resultaten er 3 hetzelfde zijn dan is dit product herkend. Is er geen product herkend dan worden deze stappen herhaald.

Is er een product herkend dan wordt de bijbehorende content aangevraagd bij de server. Zijn er meerdere content files beschikbaar voor het product dan krijgt de gebruiker de keuze welke hij wil zien. Dit gebeurt met een keuze popup. Is er maar één content file bij het product of heeft de gebruiker gekozen dan wordt deze file automatisch opgevraagd bij de server en gedownload.

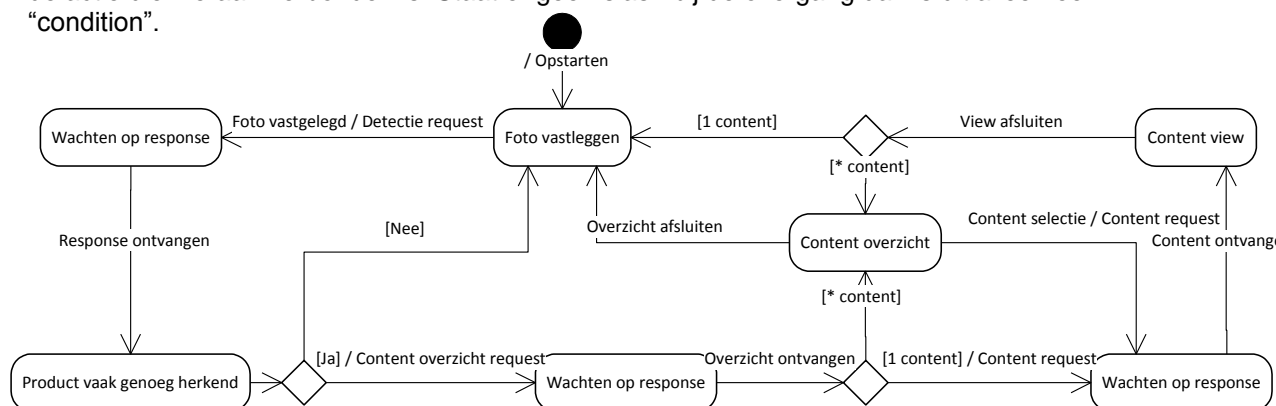
Nadat de file is gedownload en opgeslagen wordt de externe viewer aangeroepen om de file te laten zien.

Met de Back button wordt de externe viewer afgesloten. Na het afsluiten hiervan wordt er teruggekeerd naar de laatste status. Was er maar één content file bij het product dan is dit het foto vastleggen. Als er meerdere content files voor het product beschikbaar waren dan wordt de content keuze popup van het product opnieuw zichtbaar.



Figuur 7.9: Sequentie diagram herkenning

De State Machine over het gedrag van de applicatie is te vinden in figuur 7.10. Hier worden de verschillende states van de tablet applicatie bij het matchen doorlopen. Voor de overgangen wordt gebruik gemaakt van de “condition/effect” notatie. Links van de slash staat de conditie waaraan voldaan moet worden voor er naar de volgende state overgegaan wordt. Rechts staat de actie die hieraan verbonden is. Staat er geen slash bij de overgang dan is dit alleen een “condition”.



Figuur 7.10: State diagram herkenning

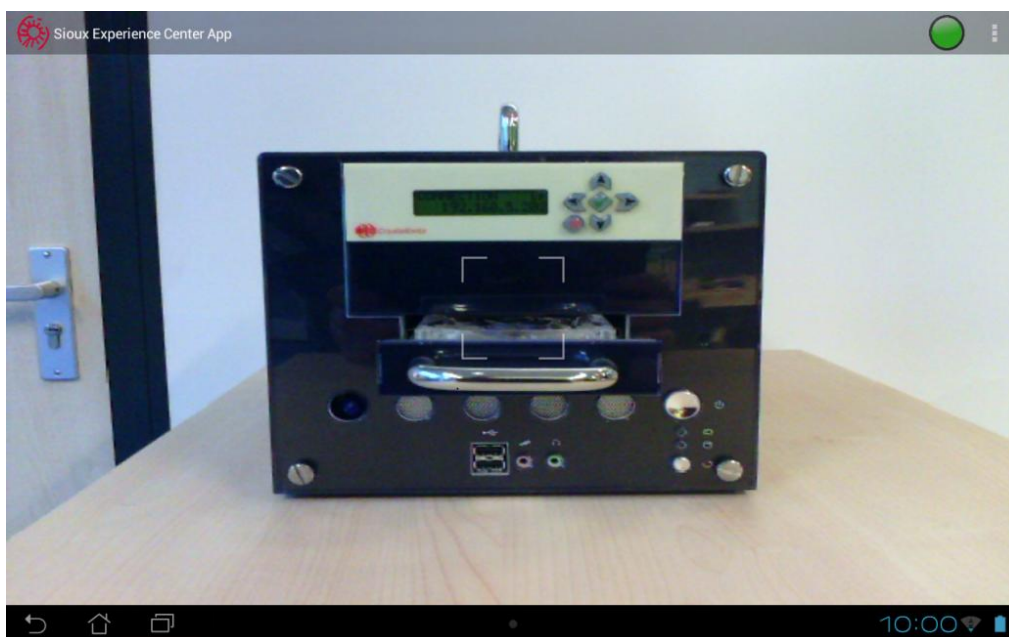
## 7.3 Ontwikkeling

Er is een tablet en een beheer applicatie ontwikkeld zoals beschreven in het ontwerp. Er is begonnen met de communicatie tussen de server en de tablet. Aan het begin waren er veel time-outs. Het probleem was dat de connectie niet goed werd afgesloten door de tablet applicatie.

De gebruiker van de tablet applicatie ziet een live preview camerabeeld. De applicatie moet van dit camerabeeld een foto kunnen opvragen zodat de server deze kan matchen. Om het camerabeeld op te vragen zijn er twee mogelijkheden. Er kan aangevraagd worden dat er een foto moet worden genomen. Door deze aanvraag maakt de camera een foto in JPEG formaat en blijft het live preview beeld hangen. Dit komt omdat het beeld de gemaakte foto laat zien. Een andere mogelijkheid is het live preview camerabeeld op te vragen. Op deze manier ontvangt de applicatie een foto in NV21 formaat (een Android specifiek beeldformaat). Er is gekozen het preview camerabeeld op te vragen omdat het storend is als het live camerabeeld blijft hangen. Hiervoor zal de NV21 foto wel moeten worden omgezet naar JPEG.

De tablet applicatie kan content opvragen bij de server. Om vast te stellen welk type het opgevraagde bestand is wordt gebruik gemaakt van MIME type. De MIME type die ontvangen wordt van de server wordt direct gebruikt voor het opstarten van de externe viewer. Dit is mogelijk omdat Android gebruik maakt van de standaard MIME types die zijn vastgelegd door IANA (33)

De HttpServer van de beheerapplicatie werkt asynchroon. Om ervoor te zorgen dat de server goed blijft presteren bij veel requests wordt er een semafoor gebruikt. De server behandelt maximaal één matchrequest per processor tegelijk. Als er meer matchrequests binnenkomen dan er tegelijk aankunnen dan zullen deze moeten wachten. De tablet applicatie heeft een time-out van 5 seconden. Als deze tijd is verlopen zal de connectie worden afgesloten door de Android applicatie.



**Figuur 7.11: Printscreen applicatie**

### 7.3.1 Performance analyse

Het duurt ongeveer 600 milliseconden om een foto te matchen op de tablet. 200 milliseconden hiervan is het omzetten van het NV21 formaat naar JPEG. Dit gebeurt op de tablet. De gemiddelde grootte van een JPEG foto is 240 kilobytes, voor een NV21 foto is dit 600 kilobytes. Als het overzetten op de server zou gebeuren in plaats van op de tablet dan duurt het oversturen van de foto (met een snelheid van 3 Mb/s) 270 milliseconden in plaats van 150 milliseconden. Dat is 120 milliseconden langer. Om nog sneller te blijven dan de tablet moet de server de foto omzetten binnen 80 milliseconden. Al zou het omzetten binnen deze tijd kunnen dan is dit maar een kleine winst. En bij een slechte netwerkverbinding kan het opsturen van meer data juist tegenwerken. Daarom is er gekozen om het omzetten door de tablet te laten doen.

De overige 400 milliseconden is voor het opsturen van de foto en wachten op antwoord. Hiervan is 150 milliseconden voor het opsturen van de JPEG body. 150 milliseconden is voor het omzetten van de foto naar keypoints en te matchen op de server. De laatste 100 milliseconden is het beantwoorden door de server en het netwerkverkeer. De tijd van het matchen is hier nog laag. Dit komt omdat de referentiedatabase nog niet volledig gevuld is met producten. Uit de resultaten van de vision onderzoek blijkt dat deze tijd rond de 500 milliseconden zal zijn. Daarmee zal de totale match tijd rond de 950 milliseconden uitkomen. De volledige tijdlijn van het matchen op de tablet is te zien in figuur 7.12.



**Figuur 7.12: Tijdlijn match op tablet (volledige referentiedatabase)**



## 7.4 Testen

Het testplan is uitgevoerd en is terug te lezen in de bijlage. Uit het testplan is gebleken dat de applicatie op de meeste punten in geslaagd. De applicatie kan producten toevoegen en herkennen.

In de demo ruimte duurt het erg lang voor een product herkend is. Dit komt omdat de wifiverbinding erg slecht is. Er is met een snelheidstest gecontroleerd hoe snel de netwerkverbinding is. In figuur 7.13 zijn de verschillende printscreens te zien van de test. Links is de test in de kantoorruimte waar goede netwerkverbinding is. De printscreen in het midden en rechts zijn allebei vanuit de demo ruimte. In de eerste printscreen is te zien dat de tablet geen netwerkverbinding heeft, dit komt vaak voor. In de rechter printscreen heeft de tablet wel een netwerkverbinding. De upload is hier wel laag, bijna 20 keer lager dan in de kantoorruimte. Dit is er onhandig in verband met het uploaden van de live foto.



*Figuur 7.13: Printscreens snelheidstest*

De test waarbij de gebruiker kan kiezen uit verschillende gevonden producten is ook niet geslaagd. Dit komt omdat de applicatie zelf het herkende product kies. Dit gebeurt met behulp van de crosshair in het midden van het scherm. Op deze manier kan de gebruiker richten op het product dat herkend moet worden.

Tot slot is het externe scherm niet geïmplementeerd. In verband met de tijd was dit niet meer mogelijk.



## 8 CONCLUSIE

Uit het vooronderzoek naar herkenningstechnieken is gebleken dat NFC en Computer Vision een oplossing zouden kunnen bieden voor het gewenste systeem. Na verder onderzoek bleek dat het bereik van NFC in de praktijk te klein is. Uit het Computer Vision onderzoek is gebleken dat SIFT3 het algoritme is dat het beste heeft gepresteerd ten aanzien van de eisen van de klant. Met dit algoritme voldoet Computer Vision aan bijna alle eisen van de klant. De eis waar niet aan wordt voldaan is betrouwbaarheid. Dit is opgelost doordat het algoritme hetzelfde product meerdere keren moet herkennen voor het naar de gebruiker als herkend aangeduid wordt. Hierdoor duurt herkenning wel langer. Uit de testresultaten blijkt dat dit geen probleem is indien de netwerkontvangst goed is. Het uitvoeren van het algoritme gebeurt op een externe server. Hier is voor gekozen omdat dit op de tablet zelf 12 keer langer duurt.

Met het opgeleverde systeem is het mogelijk om producten uit het SEC te herkennen. De manager kan met behulp van de tablet extra informatie verkrijgen van een product. Op deze manier wordt de manager ondersteund tijdens het rondleiden van een klant door het SEC.

### 8.1 Aanbevelingen

Als er verder gewerkt gaat worden aan dit project dan moet het Wi-Fi bereik in het SEC verbeterd worden. Deze is op dit moment slecht. Omdat er gebruik gemaakt wordt van een externe server in het lokale netwerk is het belangrijk dat het signaal goed is.

Achteraf was er het idee om gebruik te maken van meerder Kinect's in verschillende hoeken van het SEC. Op deze manier kan de positie van de gebruiker worden bepaald. Door middel van wijzen zou het dan mogelijk kunnen zijn om een product te herkennen. Tijdens dit project was er niet meer genoeg tijd om dit idee te onderzoeken.

Tot slot is er de wens om de clientapplicatie beschikbaar te maken voor iOS. Dit is relatief eenvoudig omdat er in de architectuur gebruik wordt gemaakt van een externe server. Hierdoor hoeft de client alleen te voldoen aan het gespecificeerde protocol.

## EVALUATIE

Mijn stage bij Sioux heb ik als erg prettig ervaren. Toen ik mijn stage begon was alles al direct goed geregeld. Ik had mijn eigen werkplek en mijn vragen kon ik bij iedereen kwijt. Ik was erg op zoek naar een afstudeeropdracht dat een stuk Android bevat. Daar heeft deze opdracht goed aan voldaan. Ook zat er een groot stuk theoretische diepgang in door middel van objectherkenning, dat zag ik als de grootste uitdaging van de afstudeeropdracht.

Ik heb tijdens mijn stage het onderzoek onderschat. Dit was meer werk dan oorspronkelijk gepland. Ook had ik moeite met het documenteren. Het schrijfwerk was niet concreet genoeg en vaak in de passieve vorm geschreven. Gelukkig heeft mijn stagebegeleider mij hier veel bij geholpen en zelfs een lijstje met schrijftips gemaakt. Het programmeren ging erg goed en snel, dit heeft de achterstand van het onderzoek een deel kunnen compenseren.

Tijdens de afstudeerstage heb ik veel zelfstandigheid gehad, dit vond ik erg fijn. Er was elke week een sprintevaluatie om te kijken of ik nog op koers lag. Mijn stagebegeleider zat recht tegenover mij. Als ik vragen had dan kon ik die meteen stellen. Hierdoor kon ik vaak direct verder.

## LITERATUURLIJST

1. Computer Vision. *Answers*. [Online] <http://www.answers.com/topic/computer-vision>.
2. Computer Vision. *Wikipedia*. [Online] 15 oktober 2012.  
[http://en.wikipedia.org/wiki/Computer\\_vision](http://en.wikipedia.org/wiki/Computer_vision).
3. Near Field Communication. *Android Developers*. [Online] Google, 22 december 2012.  
<http://developer.android.com/guide/topics/connectivity/nfc/index.html>.
4. Near field communication. *Wikipedia*. [Online] 1 januari 2013.  
[http://nl.wikipedia.org/wiki/Near\\_field\\_communication](http://nl.wikipedia.org/wiki/Near_field_communication).
5. *RFID Wikipedia*. [Online] 21 februari 2012. [http://rfidwikipedia.org/index.php/Main\\_Page](http://rfidwikipedia.org/index.php/Main_Page).
6. Radio-frequency identification. *Wikipedia*. [Online] 3 januari 2013.  
[http://en.wikipedia.org/wiki/Radio-frequency\\_identification](http://en.wikipedia.org/wiki/Radio-frequency_identification).
7. QR Code. *Wikipedia*. [Online] 1 januari 2013. [http://en.wikipedia.org/wiki/QR\\_code](http://en.wikipedia.org/wiki/QR_code).
8. WHAT'S A QR CODE? *QR Stuff*. [Online] [http://www.qrstuff.com/qr\\_codes.html](http://www.qrstuff.com/qr_codes.html).
9. **Coldewey, Devin**. Indoor GPS. *NBC News*. [Online] 26 april 2012.  
<http://www.nbcnews.com/technology/gadgetbox/get-ready-indoor-gps-tracks-your-every-step-735448>.
10. **Beijsterveld, Joost van**. Indoor navigatie. *Ego weblog*. [Online] 8 december 2011.  
<http://www.egoweblog.nl/2011/12/08/indoor-navigatie/>.
11. **Anthony, Sebastian**. Indoor navigation. *ExtremeTech*. [Online] 9 juli 2012.  
<http://www.extremetech.com/computing/132484-indoor-navigation-on-your-smartphone-using-the-earths-magnetic-field-just-like-a-homing-pigeon>.
12. **Caat**. Wat is Kinect van Microsoft. *LeerWiki*. [Online]  
[http://www.leerwiki.nl/Wat\\_is\\_Kinect\\_van\\_Microsoft](http://www.leerwiki.nl/Wat_is_Kinect_van_Microsoft).
13. **Hulsebosch, Paul**. Microsoft Kinect. *Tweakers*. [Online] 13 november 2010.  
<http://tweakers.net/reviews/1886/1/microsoft-kinect-wat-kan-het-en-hoe-werkt-het-aandacht.html>.
14. **Remondino, Fabio**. Detectors and descriptors for photogrammetric applications. *ISPRS*. [Online] 2006. [http://www.isprs.org/proceedings/XXXVI/part3/singlepapers/P\\_05.pdf](http://www.isprs.org/proceedings/XXXVI/part3/singlepapers/P_05.pdf).
15. **Szeliski, Steve Seitz and Rick**. Feature matching. *University of California Irvine*. [Online] [www.ics.uci.edu/~dramanan/teaching/cs116\\_fall08/lec/features.ppt](http://www.ics.uci.edu/~dramanan/teaching/cs116_fall08/lec/features.ppt).
16. **Pedersen, Jacob Toft**. SURF: Feature detection & description. *Aarhus University*. [Online] 2011. <http://cs.au.dk/~jtp/SURF/report.pdf>.
17. **Lindeberg, Tony**. Scale Invariant Feature Transform. *Scholarpedia*. [Online] 8 mei 2012.  
<http://www.scholarpedia.org/article/SIFT>.
18. **Faraj Alhwarin, Chao Wang, Danijela Risti -Durrant, Axel Gräser**. Improved SIFT-Features Matching for . [Online] 2008. [http://www.bcs.org/upload/pdf/ewic\\_vs08\\_s6paper2.pdf](http://www.bcs.org/upload/pdf/ewic_vs08_s6paper2.pdf).
19. **Bakken, Thomas**. An evaluation of the SIFT . [Online] 2007.  
[http://caim.uib.no/publications/Bakken\\_N30-2007.pdf](http://caim.uib.no/publications/Bakken_N30-2007.pdf).
20. **Yuehua Tao, Marjorie Skubic, Tony Han, Youming Xia, and Xiaoxiao Chi**. Performance Evaluation of SIFT-Based. *Iaeng*. [Online] 17 maart 2010.  
[http://www.iaeng.org/publication/IMECS2010/IMECS2010\\_pp1453-1456.pdf](http://www.iaeng.org/publication/IMECS2010/IMECS2010_pp1453-1456.pdf).
21. **Edouard Oyallon, Julien Rabin**. SURF: Speeded-Up Robust Features. *Image Processing On Line*. [Online] <http://www.ipol.im/pub/pre/H2/>.
22. **Herbert Bay, Andreas Ess, Tinne Tuytelaars, Luc Van Gool**. Speeded-Up Robust Features (SURF). *Computer Vision Laboratory*. [Online] 10 september 2008.  
[ftp://ftp.vision.ee.ethz.ch/publications/articles/eth\\_biwi\\_00517.pdf](ftp://ftp.vision.ee.ethz.ch/publications/articles/eth_biwi_00517.pdf).
23. **Ethan Rublee, Vincent Rabaud, Kurt Konolige, Gary Bradski**. ORB: an efficient alternative to SIFT or SURF. *Willowgarage*. [Online]  
[https://willowgarage.com/sites/default/files/orb\\_final.pdf](https://willowgarage.com/sites/default/files/orb_final.pdf).
24. **Rosten, Edward**. FAST Corner Detection . *Edward Rosten*. [Online]  
<http://www.edwardrosten.com/work/fast.html>.
25. **Edward Rosten, Reid Porter, and Tom Drummond**. Faster and better: a machine learning approach to. *Edward Rosten*. [Online]  
[http://www.edwardrosten.com/work/rosten\\_2008\\_faster.pdf](http://www.edwardrosten.com/work/rosten_2008_faster.pdf).

26. **Alexandre Alahi, Raphael Ortiz, Pierre Vandergheynst.** FREAK: Fast Retina Keypoint. [Online] <http://infoscience.epfl.ch/record/175537/files/2069.pdf>.
27. **Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua.** BRIEF: Binary Robust Independent. [Online] <http://www.robots.ox.ac.uk/~vgg/rg/papers/CalonderLSF10.pdf>.
28. **Jean-Michel more, Guoshen Yo.** ASIFT: A NEW FRAMEWORK FOR FULLY AFFINE INVARIANT. [Online] [http://www.ipol.im/pub/art/2011/my-asift/ASIFT\\_SIIMS\\_final.pdf](http://www.ipol.im/pub/art/2011/my-asift/ASIFT_SIIMS_final.pdf).
29. **Guoshen Yu, Jean-Michel Morel.** ASIFT: An Algorithm for Fully Affine Invariant Comparison. *Image Processing On Line*. [Online] 24 februari 2011. <http://www.ipol.im/pub/art/2011/my-asift/>.
30. **Stefan Leutenegger, Margarita Chli and Roland Y. Siegwart.** BRISK: Binary Robust Invariant Scalable Keypoints. [Online] <http://www.robots.ox.ac.uk/~vgg/rg/papers/brisk.pdf>.
31. OpenCV. [Online] <http://opencv.org/>.
32. EmguCV. [Online] [www.emgu.com](http://www.emgu.com).
33. **IANA.** MIME Media Type. *IANA*. [Online] 12 september 2012. <http://www.iana.org/assignments/media-types/index.html>.
34. **Viswanathan, Deepak Geetha.** Features from Accelerated Segment Test (FAST). [Online] [http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL\\_COPIES/AV1011/AV1FeaturefromAcceleratedSegmentTest.pdf](http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/AV1011/AV1FeaturefromAcceleratedSegmentTest.pdf).

## BIJLAGEN

- A - PID
- B - Requirements
- C - Onderzoek parameters
- D - Onderzoek uitkomsten
- E - Testplan



**BIJLAGE A**  
Product Initiation document

**PROJECT: Experience Center App / PID**

**Document**



<b>Projectcode:</b>	Afstuderen
<b>Datum voltooid:</b>	01 oktober 2012
<b>Auteur:</b>	Joris Reijrink
<b>Versie:</b>	3.0
<b>Status:</b>	Release
<b>Document ID:</b>	PID
<b>Bestandsnaam:</b>	PID- Experience Center App.docx

## Documenthistorie

### Revisies

Versie	Status	Datum	Wijzigingen
0.1	Concept	11-09-2012	Geen wijzigingen
0.2	Concept	13-09-2012	Aanpassingen na toetsing stagebegeleider en gesprek met product owner
1.0	Release	14-09-2012	Goedkeuring stage begeleider
2.0	Release	24-09-2012	Feedback schoolbegeleider
3.0	Release	01-10-2012	Feedback schoolbegeleider

### Goedkeuring

Dit document behoeft de volgende goedkeuringen:

Versie	Datum goedkeuring	Naam	Functie	Paraaf
		Rico Huijbers	Stagebegeleider	
		Ben Schreur	Schoolbegeleider	

### Distributie

Dit document is verstuurd aan:

Versie	Datum verzending	Naam	Functie
0.1	11-09-2012	Rico Huijbers	Stagebegeleider
0.2	13-09-2012	Rico Huijbers	Stagebegeleider
1.0	14-09-2012	Ben Schreur	Schoolbegeleider
1.0	14-09-2012	Rico Huijbers	Stagebegeleider
2.0	24-09-2012	Rico Huijbers	Stagebegeleider
3.0	01-10-2012	Ben Schreur	Schoolbegeleider
3.0	01-10-2012	Rico Huijbers	Stagebegeleider

## Inhoudsopgave

<b>1 INLEIDING</b>	<b>4</b>
1.1 Doel van dit document	4
1.2 Opbouw van dit document	4
<b>2 PROJECTDEFINITIE</b>	<b>5</b>
2.1 Projectdoelstellingen	5
2.2 Gekozen oplossing of aanpak	5
2.3 Scope van het project	5
2.4 Producten & eindresultaat	5
2.5 Beperkingen	5
2.6 Randvoorwaarden	6
2.7 Eigendom code	6
2.8 Ontwikkelresources	6
2.9 Risico's	6
<b>3 PROJECTORGANISATIESTRUCTUUR</b>	<b>7</b>
3.1 Eindklant	7
3.2 Bedrijfsbegeleider	7
3.3 Schoolbegeleider	7
<b>4 PROJECTBEHEERSING</b>	<b>8</b>
4.1 Deliverables	8
4.2 Voortgangsbewaking	8
4.3 Issue management	8
4.4 Afwijkings- en escalatieprocedure	8
<b>BIJLAGE A: COMMUNICATIEPLAN</b>	<b>9</b>
<b>BIJLAGE B: CONTACTGEGEVENS</b>	<b>9</b>
<b>BIJLAGE C: PLANNING</b>	<b>9</b>

# 1 INLEIDING

## 1.1 Doel van dit document

Dit document is opgesteld om alle relevante basisinformatie en uitgangspunten van het project vast te leggen om het op de juiste wijze te kunnen besturen. Het heeft tot doel het project te definiëren, als basis te dienen voor het management ervan en de beoordeling van het succes van het project mogelijk te maken.

Dit Projectinitiatiedocument (of PID) behandelt de volgende fundamentele aspecten van het project:

- Wat beoogt men met het project te bereiken?
- Waarom is het belangrijk om deze doelstellingen te bereiken?
- Wie zijn er betrokken bij het managen van het project en wat zijn hun rollen en verantwoordelijkheden?
- Hoe en wanneer zullen de maatregelen die in dit PID besproken worden gerealiseerd worden?

Het document wordt gebruikt:

- Om er zeker van te zijn dat het project een gezonde basis heeft voordat de product owner gevraagd wordt zich aan het project te committeren;
- Om te dienen als basisdocument op grond waarvan de product owner en de stagiair de voortgang en wijzigingen kunnen toetsen en bewaken en vragen omtrent geldigheid van het project tijdens de uitvoering ervan kunnen beoordelen.

## 1.2 Opbouw van dit document

De volgende onderwerpen worden behandeld in dit document:

- Projectdefinitie (Hoofdstuk 2)
- Projectorganisatiestructuur (Hoofdstuk 3)
- Communicatieplan (Bijlage A)
- Contactgegevens (Bijlage B)
- Planning (Bijlage C)



## 2 PROJECTDEFINITIE

### 2.1 Projectdoelstellingen

Het doel van het project is een tablet applicatie te ontwikkelen waarmee Sioux haar zakenrelatie een rondleiding geeft door haar portfolio. Met behulp van product herkenning door de applicatie kan de gebruiker meer informatie verkrijgen over de producten in het portfolio. Uit onderzoek zal blijken wat de meeste geschikte herkenningmethode is. Verder is de wens dat de applicatie indrukwekkend ervaring geeft naar de gebruiker, om zo de portfolio rondleiding een echte “experience” te maken.

### 2.2 Gekozen oplossing of aanpak

Tijdens de uitvoering van het project wordt gebruik gemaakt van de scrum methode met een sprintlengte van twee weken. Aan het begin van iedere sprint wordt met de product owner overlegd welke producten worden opgeleverd en wat de prioriteit hiervan is. Aan het einde van de sprint worden de uitgewerkte producten gepresenteerd en/of opgeleverd, ook wordt de sprint geëvalueerd door de stagiair middels een sprintrapport.

### 2.3 Scope van het project

De scope van het project omvat zowel hardware als software benodigd voor de werking van het systeem. Denk hierbij aan een tablet of desktop server, maar ook de software die erop komt te draaien.

### 2.4 Producten & eindresultaat

De volgende producten worden opgeleverd:

- Project Initiatie Document (PID)
- Requirements
- Testplan
- Onderzoek beste herkenningmethode
- Systeemontwerp
- Tussenpresentatie
- Software
- Testresultaten
- Eindpresentatie
- Eindverslag
- Sprint rapporten

Na oplevering van een product betekent dit niet dat deze statisch wordt, een product kan nog gewijzigd worden gedurende het proces. Dit is vooral van toepassing bij documenten zoals de requirements en het testplan.

### 2.5 Beperkingen

Beschikbare werktijd	Beschikbare resources	Gewenste opleverdata	Beschikbare Doorlooptijd
800 uur in totaal, waarvan 560 uur effectief (gerekend met focusfactor van 70%)	- Asus Transformer Tablet - Begeleiding: 4 uur per week	Eind iedere sprint	20 weken

## 2.6 Randvoorwaarden

Producten die aan de product owner en/of begeleiders ter goedkeuring worden aangeboden dienen binnen één werkweek voorzien van commentaar te worden geretourneerd. Mocht dit niet haalbaar zijn, dan wordt binnen deze periode gecommuniceerd wanneer inhoudelijke feedback wel mogelijk is.

## 2.7 Eigendom code

Alle code en documenten die door de stagiair gemaakt worden zijn eigendom van Sioux en mogen niet zonder goedkeuring van de eigenaar gedistribueerd worden naar externen.

## 2.8 Ontwikkelresources

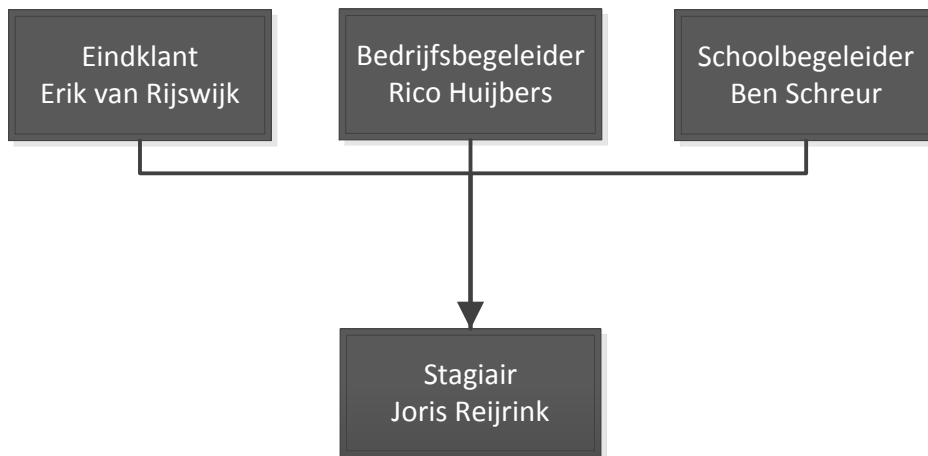
Sioux heeft de volgende resources beschikbaar gesteld voor de stagiair om het systeem mee te ontwikkelen:

- Werkstation
- Software
- Bestaande code en documenten over het project
- Archief

## 2.9 Risico's

Er is nagedacht over de eventuele risico's van dit project en er is de conclusie getrokken dat er geen specifieke risico's zijn die voor dit project benoemt dienen te worden.

### 3 PROJECTORGANISATIESTRUCTUUR



#### 3.1 Eindklant

Erik van Rijswijk is de product owner en heeft opdracht gegeven voor de uitvoering van het project. Hij vormt de verzoekende partij en is verantwoordelijk voor het overbrengen van wensen en eisen aan het op te leveren product. Daarnaast heeft hij het laatste woord bij de beoordeling van het opgeleverde product en voorgestelde productuitbreidingen.

#### 3.2 Bedrijfsbegeleider

Rico Huijbers is bedrijfsbegeleider bij dit project en is het eerste aanspreekpunt voor de stagiair, voor zowel procesmatige als technische vragen. Verder is hij verantwoordelijk voor de benodigde begeleiding van de stagiair.

#### 3.3 Schoolbegeleider

Ben Schreur is schoolbegeleider bij dit project en kan de stagiair adviseren bij vraagstukken op het gebied het proces. Daarnaast fungeert hij als aanspreekpunt voor facilitaire zaken die geregeld moeten worden bij Fontys.

## 4 Projectbeheersing

### 4.1 Deliverables

Rapport:	Partij:	Product owner	Bedrijfsbegeleider	Schoolbegeleider	Stagiair
<b>PID</b>			G	G	O T D
<b>Requirements</b>		O G	A T	I	O T D
<b>Testplan</b>		G	A T	I	O T D
<b>Onderzoeksdocument</b>			G	G	O T D
<b>Ontwerp</b>			A G	I	O T D
<b>Sprintrapport</b>			I	I	O T D
<b>Logboek</b>				I	O T D
<b>Eindproduct</b>		G	A T	A I	O T D
<b>Eindverslag</b>			G	G	O T D
<b>Sprint demo's</b>		T	I		O T

<b>O</b>	Opstellen	<b>A</b>	Adviseren	<b>I</b>	Ontvangen ter informatie
<b>T</b>	Toetsen	<b>D</b>	Distribueren/archiveren	<b>G</b>	Goedkeuren

### 4.2 Voortgangsbewaking

Overleg	Aanwezig	Frequentie	Doel	Onderwerpen
Daily scrum	Stagiair, Bedrijfsbegeleider	Iedere werkdag	Status update taken	Resultaten vorige werkdag, obstakels, taken nieuwe werkdag
Scrum meeting	Stagiair, Bedrijfsbegeleider, Product owner	Eens per twee weken	Klant informeren en nieuwe doelen opstellen	Resultaten presenteren, nieuwe doelen opstellen
Spring evaluatie	Stagiair, Bedrijfsbegeleider	Eens per twee weken	Sprint evalueren	Sprint rapport doornemen

### 4.3 Issue management

In het geval dat zich problemen voor gaan doen met het project, wordt er een vergadering ingepland met de product owner en/of bedrijfsbegeleider om de ernst van de zaak te bepalen. Daarnaast wordt een plan opgesteld om het probleem op te lossen.

### 4.4 Afwijkings- en escalatieprocedure

Bij afwijkingen zal de stagiair met de product owner in gesprek gaan om te kijken of deze afwijkingen acceptabel zijn. Mochten deze afwijkingen problemen geven in de planning dan is het mogelijk dat er in overleg tegemoetkoming zal worden besproken op het gebied van tijd, kosten of functionaliteit.

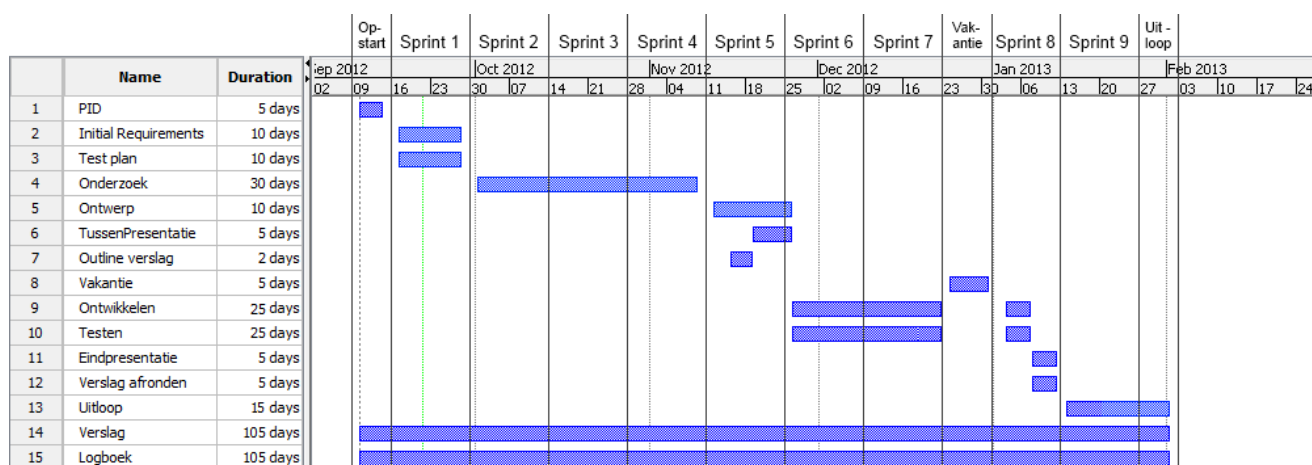
## BIJLAGE A: COMMUNICATIEPLAN

Van	Naar	Informatie	Medium	Frequentie
Stagiair	Schoolbegeleider	Proces	E-mail, persoonlijk	Indien nodig
Schoolbegeleider	Stagiair	Feedback	E-mail, persoonlijk	Indien nodig
Stagiair	Bedrijfsbegeleider	Voortgang	Daily scrum	Iedere werkdag
Bedrijfsbegeleider	Stagiair	Feedback	E-mail, persoonlijk	Indien nodig
Stagiair	Product owner	Expertisevragen, scrum meeting	E-mail, persoonlijk	Indien nodig
Product owner	Stagiair	N.n.t.b.	E-mail, persoonlijk	Indien nodig

## Bijlage B: Contactgegevens

Naam	PCN	StudentNr	E-mailadres	Telefoonnummer
Joris Reijrink	205790	2127713	<a href="mailto:j.reijrink@student.fontys.nl">j.reijrink@student.fontys.nl</a>	+31634625910
Rico Huijbers			<a href="mailto:Rico.huijbers@sioux.eu">Rico.huijbers@sioux.eu</a>	+31639452112
Ben Schreur			<a href="mailto:b.schreur@fontys.nl">b.schreur@fontys.nl</a>	0885086970

## Bijlage C: Planning



### Data oplevering documenten

- PID (14 september 2012)
- Initial Requirements (28 september 2012)
- Testplan (28 september 2012)
- Onderzoek (09 november 2012)
- Systeemontwerp (23 november 2012)
- Testrapport (04 Januari 2012)
- Eindverslag (12 Januari 2012)



## **BIJLAGE B**

### Requirements

# **PROJECT: Experience Center App / INITIATIEFASE**

## **REQUIREMENTS DOCUMENT**



**Projectcode:** Afstuderen

**Datum voltooid:** 25 oktober 2012  
**Auteur:** Joris Reijrink

**Versie:** 1.3  
**Status:** Release

**Document ID:** Requirements  
**Bestandsnaam:** Requirements - Experience Center App.docx

## Documenthistorie

### Revisies

Versie	Status	Datum	Wijzigingen
0.1	Concept	13-09-2012	Geen wijzigingen
0.2	Concept	21-09-2012	Aanpassingen na toetsing stagebegeleider
0.3	Concept	24-09-2012	Aanpassingen na toetsing stagebegeleider
1.0	Release	27-09-2012	Aanpassingen na toetsing stagebegeleider
1.1	Release	16-10-2012	Kleine aanpassing requirements
1.2	Release	24-10-2012	Aanpassingen feedback product owner
1.3	Release	25-10-2012	Requirement toegevoegd

### Goedkeuring

Dit document behoeft de volgende goedkeuringen:

Versie	Datum goedkeuring	Naam	Functie	Paraaf
		Erik van Rijswik	Product owner	

### Distributie

Dit document is verstuurd aan:

Versie	Datum verzending	Naam	Functie
0.1	13-09-2012	Rico Huijbers	Stagebegeleider
0.2	21-09-2012	Rico Huijbers	Stagebegeleider
0.3	24-09-2012	Rico Huijbers	Stagebegeleider
1.0	27-09-2012	Rico Huijbers	Stagebegeleider
1.0	27-09-2012	Erik van Rijswik	Product Owner
1.0	27-09-2012	Ben Schreur	Schoolbegeleider
1.2	24-10-2012	Rico Huijbers	Stagebegeleider
1.3	25-10-2012	Rico Huijbers	Stagebegeleider

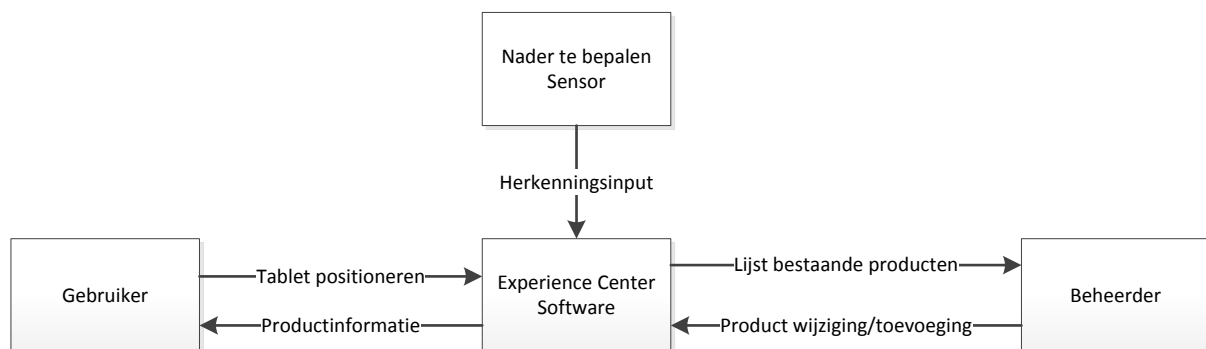
## Inhoudsopgave

<b>1 GEBRUIKER</b>	<b>4</b>
<b>2 CONTEXT</b>	<b>4</b>
Terminators	4
Flows	4
<b>3 POINTS OF INTEREST</b>	<b>5</b>
<b>4 BEPERKINGEN</b>	<b>6</b>
<b>5 USER INTERFACE</b>	<b>6</b>
<b>6 FUNCTIONELE EN NIET FUNCTIONELE REQUIREMENTS</b>	<b>7</b>
Functionele requirements	7
Niet functionele requirements	7
<b>USE CASES</b>	<b>8</b>

# 1 GEBRUIKER

De applicatie zal gebruikt worden voor het informeren van klanten over de werkzaamheden van Sioux door middel van de producten die staan in het Sioux Experience Center. Dit zal gebeuren door de managers die werken bij Sioux. Omdat deze managers soms het verhaal bij een product niet weten is deze nog wel eens verschillend, hier zal de applicatie verandering in gaan brengen. De applicatie zal op een nog nader te bepalen methoden producten kunnen herkennen, die er vervolgens informatie over geeft aan de manager. Op deze manier heeft de manager altijd informatie bij de hand over de producten in het Sioux Experience Center voor tijdens rondleidingen met klanten.

## 2 CONTEXT



### Terminators

**Gebruiker:** zorgt voor input van de sensor door het positioneren van de tablet en krijgt hierdoor product informatie terug van de software.

**Beheerder:** voegt nieuwe producten toe aan de software of past bestaande aan.

**Sensor:** geeft input aan de software zodat de software het bijbehorend product kan identificeren.

### Flows

**Tablet positioneren:** door de tablet juist te positioneren krijgt de sensor specifieke input, hierdoor kan de software een product herkennen.

**Productinformatie:** als de software een product heeft herkend krijgt de gebruiker informatie terug over dat product.

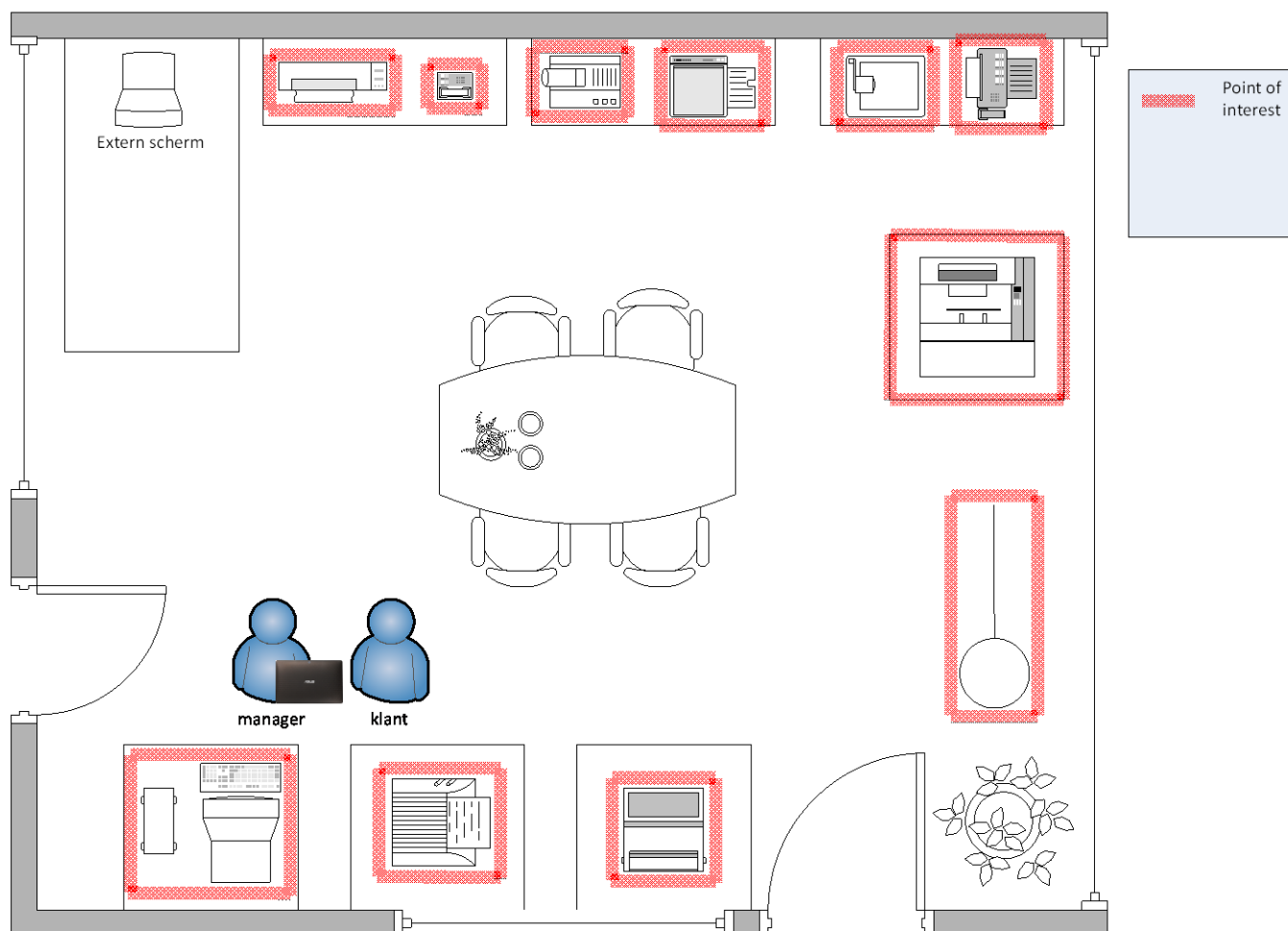
**Herkenninginput:** geeft identificatie input door aan de software als de tablet hiervoor gepositioneerd is, hierdoor kan de software bepalen bij welk product het gepositioneerd is.

**Lijst bestaande producten:** de complete dataset van alle producten zodat deze beheerd kunnen worden. Deze dataset bevat onder andere de herkenningpunten van elke product en de content hiervan.

**Product wijziging/toevoeging:** de beheerder kan een product toevoegen of data van een product aanpassen, dit kan bijvoorbeeld een herkenningpunt of content zijn.



### 3 POINTS OF INTEREST



Het Sioux Experience Center bevat verschillende “points of interest” die allemaal herkend moeten kunnen worden door het systeem. Het is belangrijk te weten dat de producten zich op verschillende hoogten bevinden en dus ook boven elkaar kunnen liggen. Hierdoor is het mogelijk dat sommige product herkenning methoden uitgesloten zijn omdat deze bijvoorbeeld geen drie dimensies aankunnen. Ook moet er rekening gehouden worden met producten die in sommige gevallen bijna tegen elkaar aan liggen, het systeem moet dus accuraat zijn.

## 4 BEPERKINGEN

Het Sioux experience Center mag niet zichtbaar veranderd worden, er mag voor de gebruiker niets zichtbaar zijn van het systeem. Hierdoor zijn niet alle object herkenning methoden mogelijk omdat er geen zichtbare input geplaatst mag worden.

## 5 USER INTERFACE

De user interface is een belangrijk aspect van de applicatie omdat de klant overweldigd moet worden door het systeem. Hierdoor zal de applicatie een innovatieve, uitdagende en professionele user interface moeten hebben. Ook is het belangrijk dat de applicatie makkelijk te bedienen is en intuïtief werkt voor de gebruiker. Verder moet de product informatie duidelijk en overzichtelijk gepresenteerd worden naar de gebruiker.

## 6 FUNCTIONELE EN NIET FUNCTIONELE REQUIREMENTS

Voor het vaststellen van de prioriteiten is gebruik gemaakt van de MoSCoW methode. Dit is een afkorting waarvan de letters staan voor:

- **M** – MUST: deze eis *moet* in het eindresultaat terugkomen, zonder deze eis is het product niet bruikbaar
- **S** – SHOULD: deze eis is zeer gewenst, maar zonder is het product wel bruikbaar.
- **C** – COULD: deze eis mag alleen aan bod komen als er tijd genoeg is.
- **W** – WON'T – deze eis zal in dit project niet aan bod komen maar kan in de toekomst, bij een vervolgproject, interessant zijn.

	Functionele requirements	MoSCoW
1.0	Product herkenning	M
	<i>Het systeem kan een product van het Sioux Experience Center herkennen.</i>	
1.1	PowerPoint presentatie	M
	<i>De applicatie kan een PowerPoint presentatie bij een herkend product presenteren</i>	
1.2	Automatische informatie	M
	<i>De applicatie geeft informatie over een object als deze herkend is zonder dat hier de interactie van de gebruiker voor nodig is.</i>	
1.3	Makkelijk producten beheren	M
	<i>De beheerder kan eenvoudig producten en content toevoegen, aanpassen en verwijderen in het systeem.</i>	
1.4	Extern scherm	C
	<i>Video content van een herkend product kan op een extern scherm gepresenteerd.</i>	
1.5	Direct bruikbaar	M
	<i>Nadat de applicatie is opgestart kan deze direct objecten herkennen zonder hiervoor extra handelingen uit te hoeven voeren, mits de instellingen juist zijn ingevuld.</i>	

	Niet functionele requirements	MoSCoW
1.6	Android	M
	<i>De applicatie draait op een android device.</i>	
1.7	Betrouwbaar	M
	<i>De applicatie herkent minstens 90% van de tijd het product, indien deze bekend is in het systeem.</i>	
1.8	Nauwkeurigheid	M
	<i>Producten met 5 centimeter of meer tussenruimte kunnen nog onderscheiden worden.</i>	
1.9	Afstand	M
	<i>De afstand van het herkennen van objecten ligt tussen 5 centimeter en 1 meter</i>	
1.10	Snelheid	M
	<i>Het mag niet langer dan 10 seconden duren voor een product herkend is</i>	

## USE CASES

<b>ID</b>	<b>1. Product herkennen</b>
<b>Acteurs</b>	Gebruiker
<b>Pre-Condition</b>	<ul style="list-style-type: none"><li>• Het systeem is opgestart</li><li>• Minstens één product bekend in systeem</li></ul>
<b>Acties</b>	<ul style="list-style-type: none"><li>• De gebruiker positioneert de tablet</li><li>• De sensor ontvangt herkenninginput</li><li>• Het systeem herkent product(en)</li><li>• Indien meerdere producten herkend door sensor kiest de gebruiker een product op het scherm</li><li>• De gebruiker krijgt informatie over het product zichtbaar op de tablet</li></ul>
<b>Post-Condition</b>	Productinformatie op het scherm

<b>ID</b>	<b>2. Geen product herkend</b>
<b>Acteurs</b>	Gebruiker
<b>Pre-Condition</b>	<ul style="list-style-type: none"><li>• Het systeem is opgestart</li></ul>
<b>Acties</b>	<ul style="list-style-type: none"><li>• De gebruiker positioneert de tablet</li><li>• De sensor ontvangt herkenninginput</li><li>• Het systeem kent de herkenninginput niet</li><li>• Op de tablet gebeurt niets zichtbaars</li></ul>
<b>Post-Condition</b>	Het scherm is ongewijzigd

<b>ID</b>	<b>3. Product toevoegen</b>
<b>Acteurs</b>	Beheerder
<b>Pre-Condition</b>	<ul style="list-style-type: none"><li>• Het systeem is opgestart</li></ul>
<b>Acties</b>	<ul style="list-style-type: none"><li>• De beheerder verbindt met het systeem</li><li>• Indien nodig wordt de sensor geplaatst/ingesteld</li><li>• Het product wordt aan het systeem toegevoegd met de herkenningsdata en eigen content</li><li>• Het systeem voegt het product toe aan de database</li></ul>
<b>Post-Condition</b>	Het product is toegevoegd aan het systeem

<b>ID</b>	<b>4. Product wijzigen</b>
<b>Acteurs</b>	Beheerder
<b>Pre-Condition</b>	<ul style="list-style-type: none"><li>• Het systeem is opgestart</li><li>• Minstens één product bekend in systeem</li></ul>
<b>Acties</b>	<ul style="list-style-type: none"><li>• De beheerder verbindt met het systeem</li><li>• Het aan te passen product wordt opgezocht</li><li>• Indien nodig wordt de sensor aangepast</li><li>• De herkenningsdata of content van het product wordt aangepast</li><li>• Het systeem past het product aan in de database</li></ul>
<b>Post-Condition</b>	Het product is gewijzigd in het systeem

<b>ID</b>	<b>5. Product verwijderen</b>
<b>Acteurs</b>	Beheerder
<b>Pre-Condition</b>	<ul style="list-style-type: none"><li>• Het systeem is opgestart</li><li>• Minstens één product bekend in systeem</li></ul>
<b>Acties</b>	<ul style="list-style-type: none"><li>• De beheerder verbindt met het systeem</li><li>• Het te verwijderen product wordt opgezocht</li><li>• Het gevonden product wordt verwijderd</li><li>• Het systeem verwijdert het product van de database</li></ul>
<b>Post-Condition</b>	Het product is verwijderd van het systeem



## **BIJLAGE C**

### Onderzoek parameters

### **SIFT1**

Detector:

Descriptor:

- nFeatures 3
- nOcateLayers 10
- contrastThresold 0,04
- edgeThreshold 10
- sigma 1,6

### **SIFT2**

Detector:

Default

Descriptor:

- nFeatures 10
- nOcateLayers 10
- contrastThresold 0,1
- edgeThreshold 10
- sigma 1,6

### **SIFT3**

Detector:

Descriptor:

- Default

### **SURF1**

Detector:

Descriptor:

- hessianThresh 500
- extendedFlag False

### **SURF2**

Detector:

Descriptor:

- hessianThresh 750
- extendedFlag FALSE
- nOctaves 4
- nOctaveLayers 2

### **ORB1**

Detector:

Descriptor:

- NumberOfFreatures 2000

### **ORB2**

Detector:

Descriptor:

- NumberOfFreatures 2000
- scaleFactor 1,5
- nLevels 8
- edgeThreshold 0
- firstLevel 1
- WTKA 2
- scoreType Harris
- patchSize 31

### **FAST1**

FAST Detector

- threshold 22
- nonmaxSupression True

SURF Descriptor

- hessianThresh 500
- extendedFlag False

### **FREAK1**

ORB Detector

- NumberOfFreatures 2000

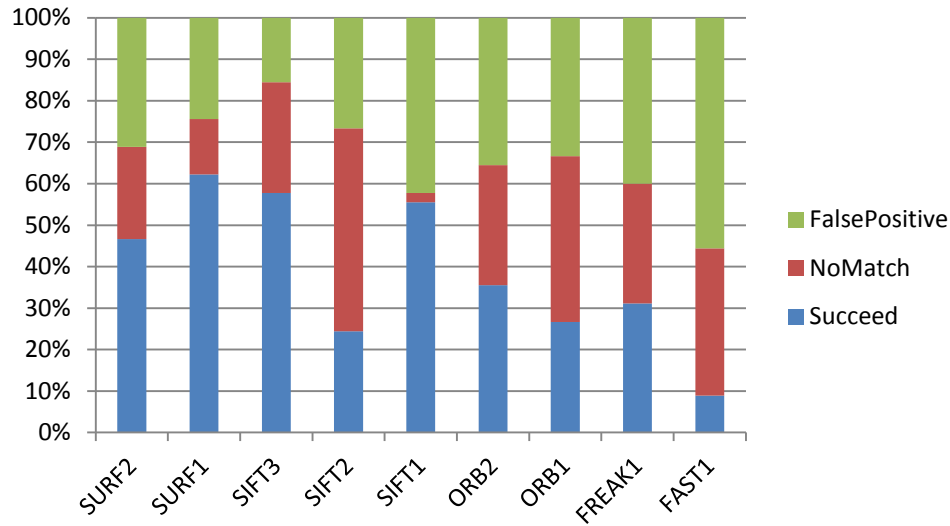
Freak Descriptor

- orientationNormalized True
- scaleNormalized True
- patternScale 5
- nOctaves 12

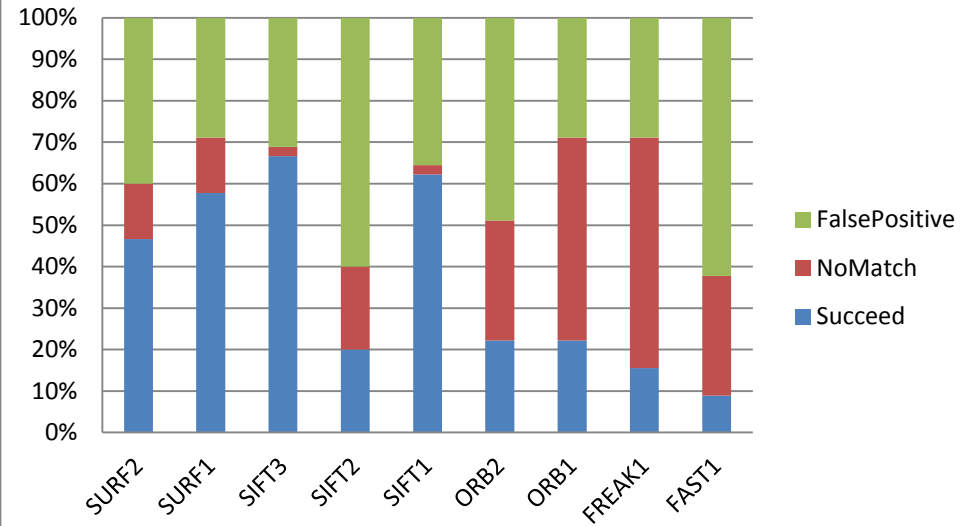
**BIJLAGE D**  
Onderzoek uitkomsten

# Match Results

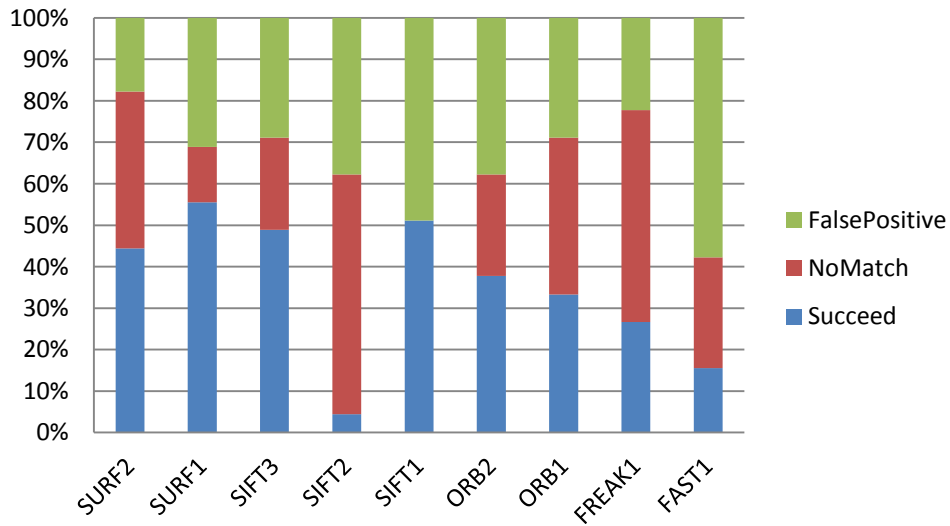
## 0,3 Megapixels Phone



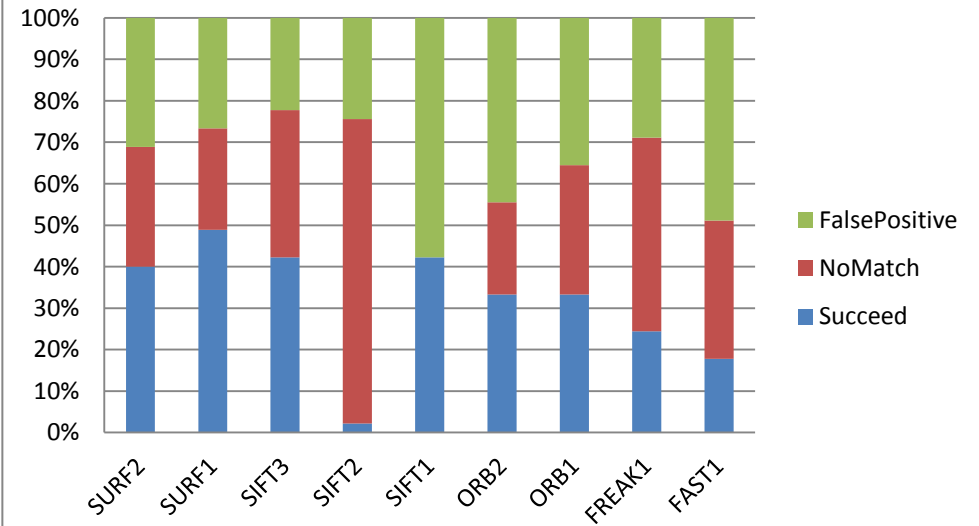
## 6 Megapixels Phone



## 0,3 Megapixels Tablet

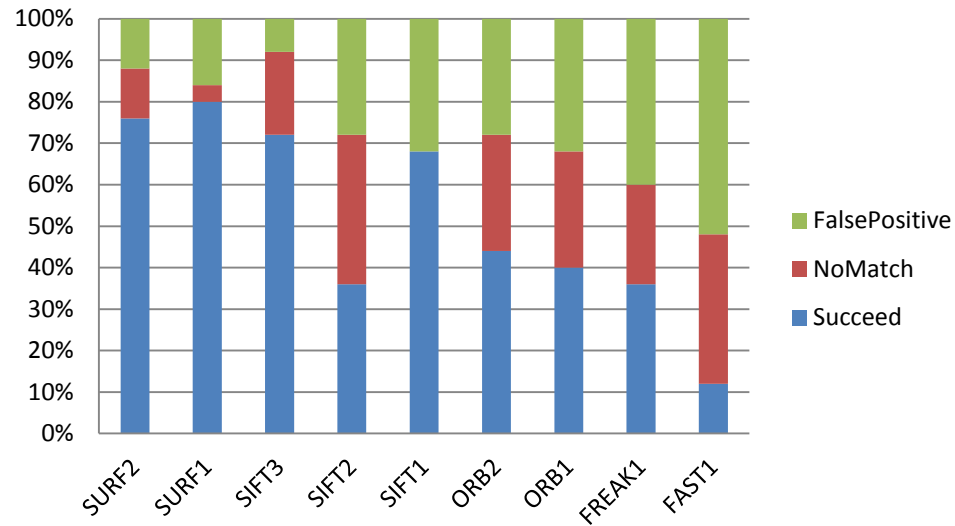


## 0,3 Megapixels Tablet Cut

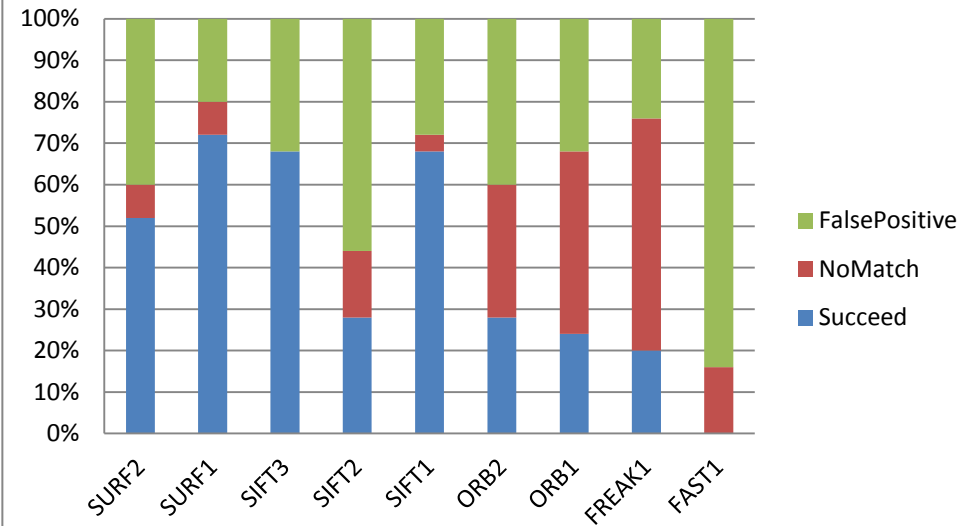


# Match Results (NoExtreme)

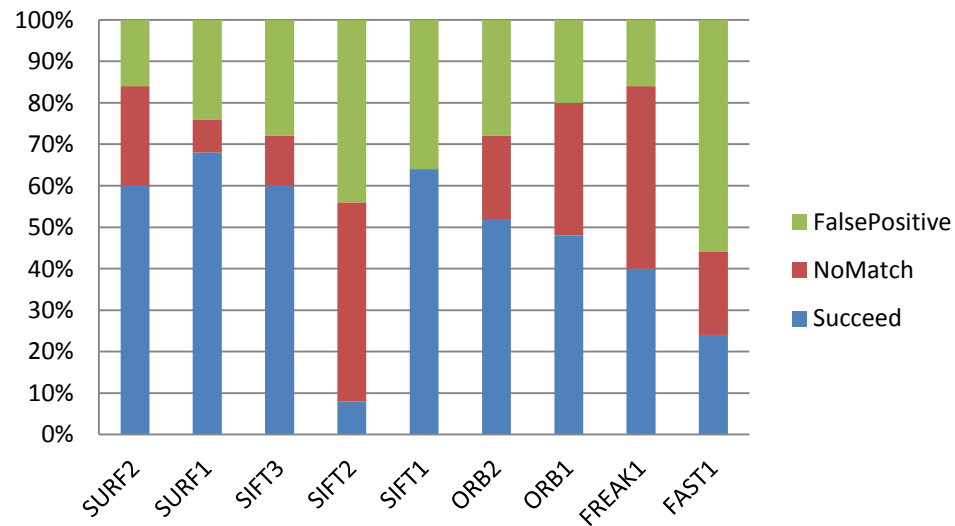
## 0,3 Megapixels Phone



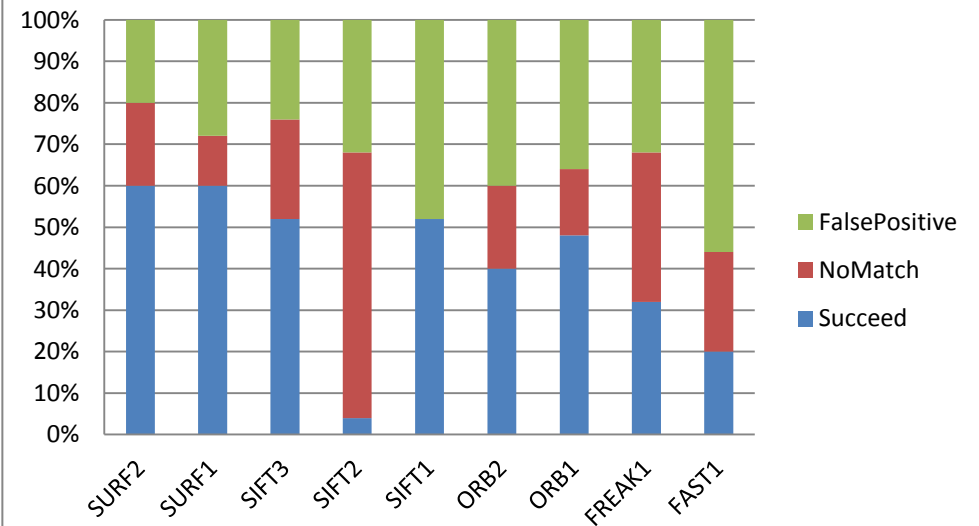
## 6 Megapixels Phone



## 0,3 Megapixels Tablet



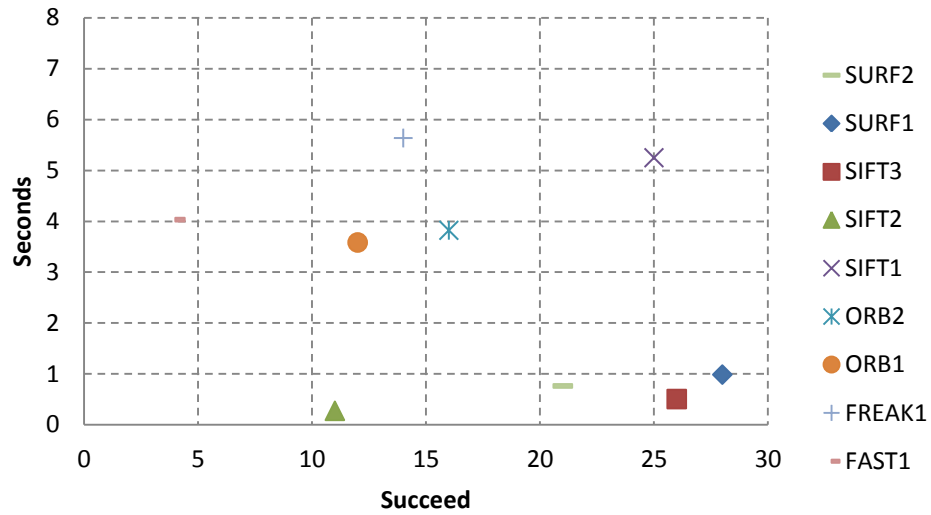
## 0,3 Megapixels Tablet Cut



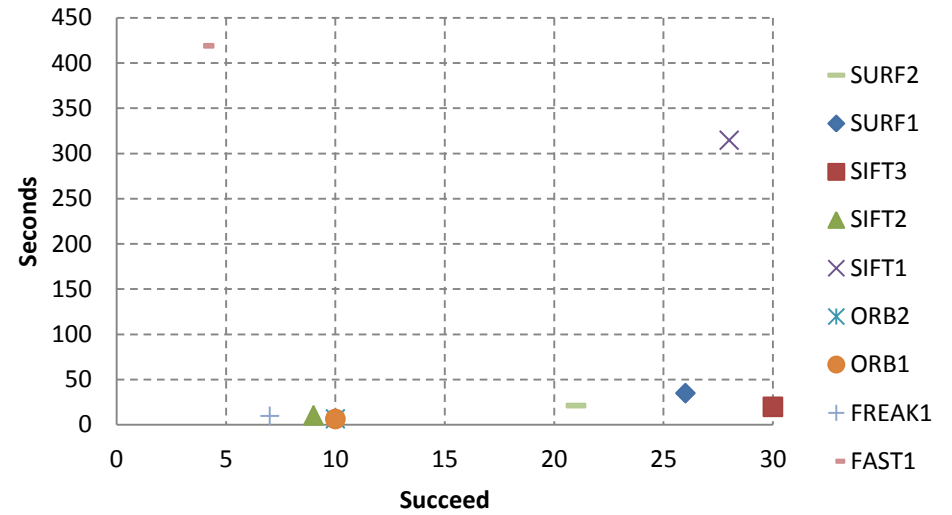


# Time Results

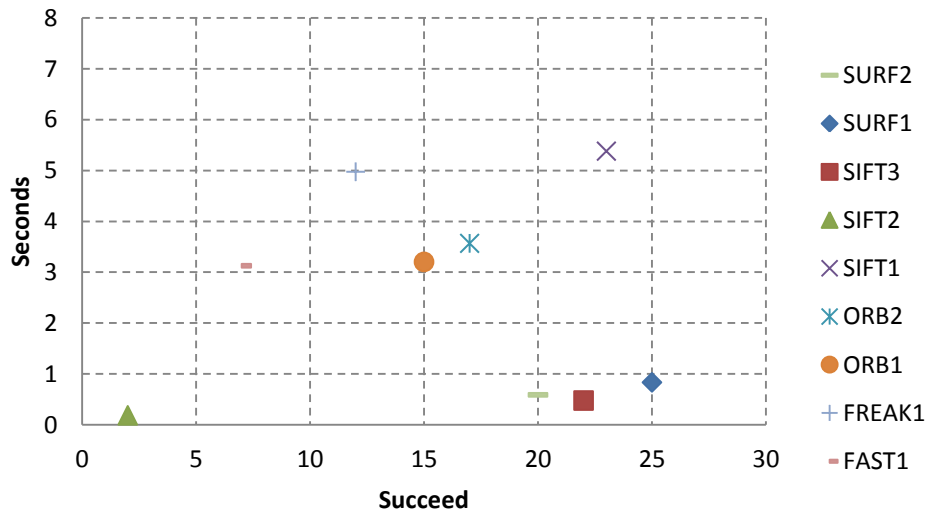
## 0,3 Megapixels Phone



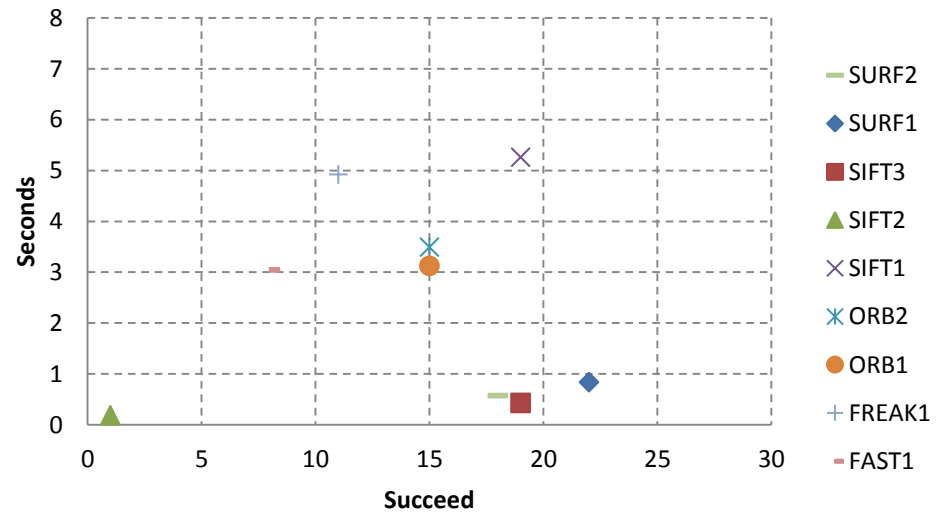
## 6 Megapixels Phone



## 0,3 Megapixels Tablet

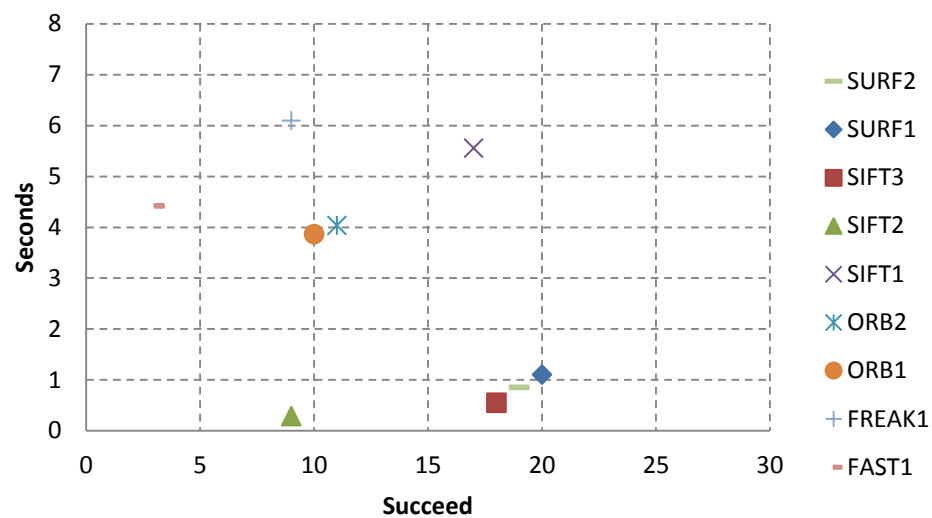


## 0,3 Megapixels Tablet Cut

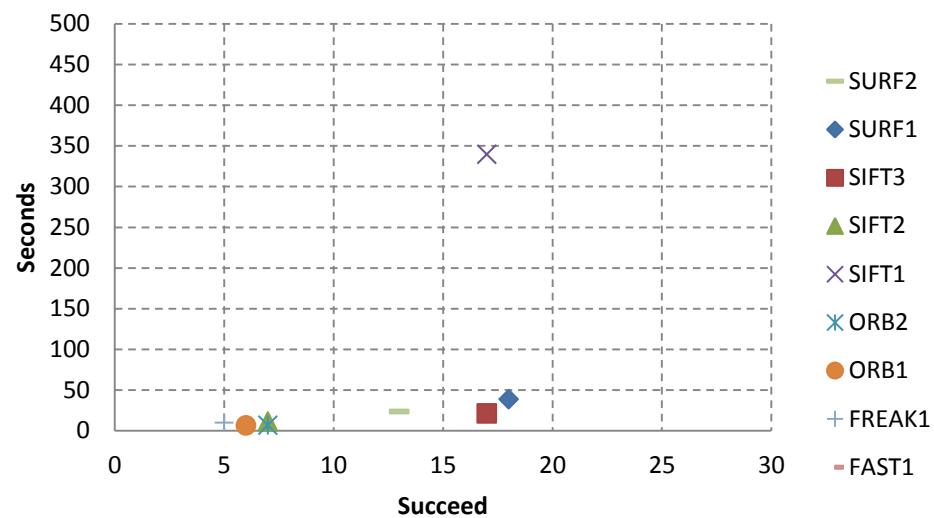


# Time Results (NoExtreme)

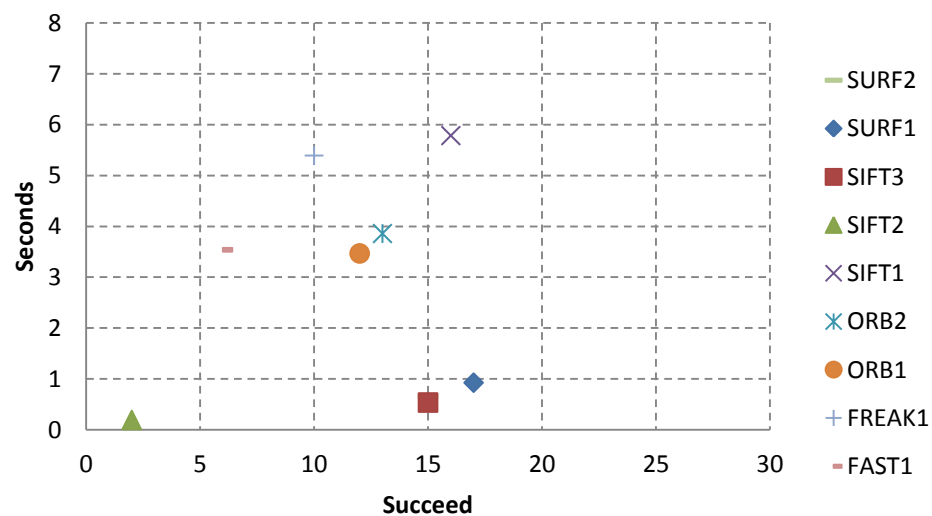
## 0,3 Megapixels Phone



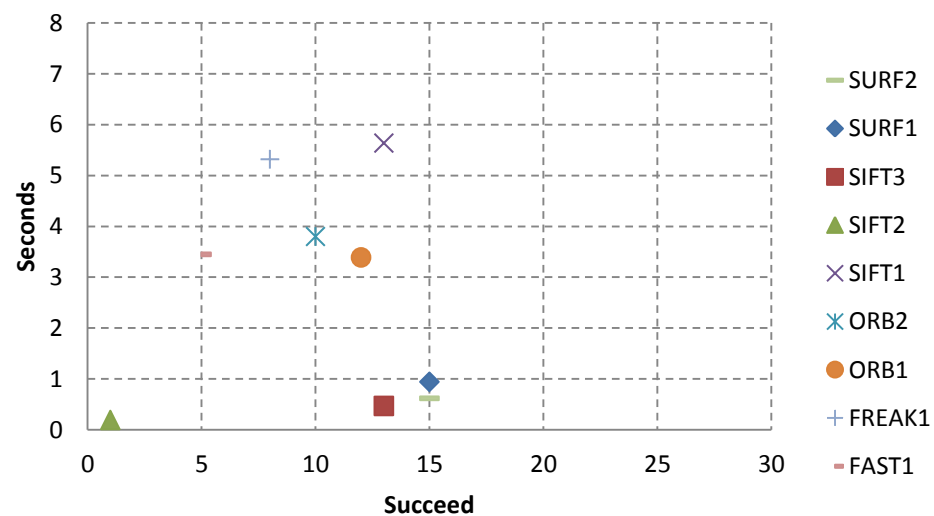
## 6 Megapixels Phone



## 0,3 Megapixels Tablet

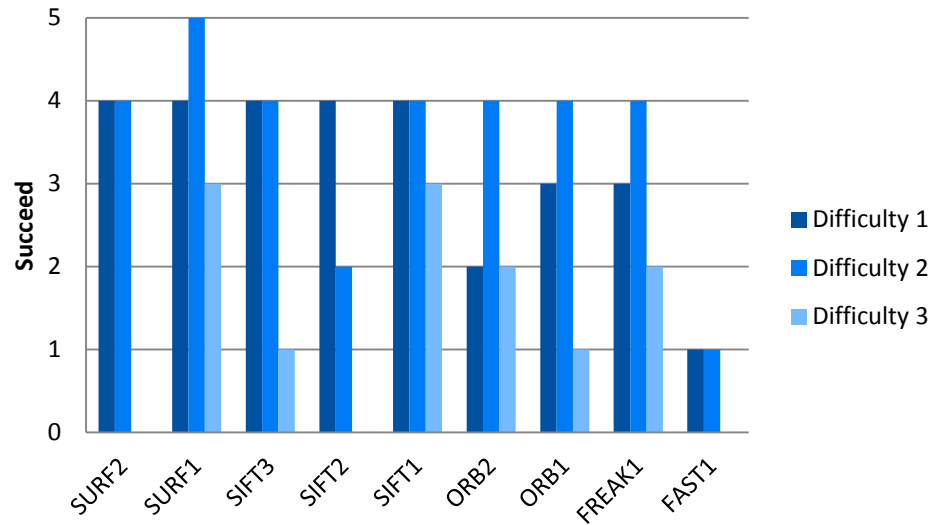


## 0,3 Megapixels Tablet Cut

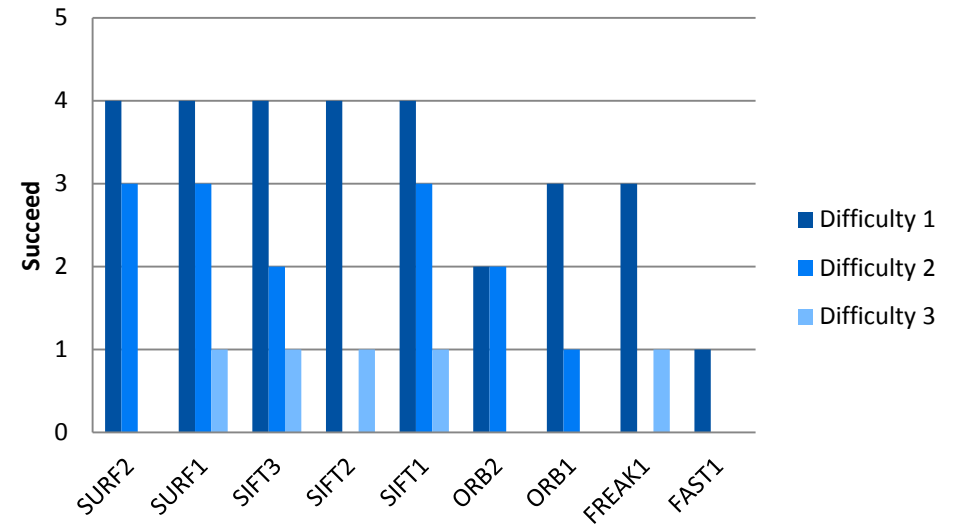


## Case Results

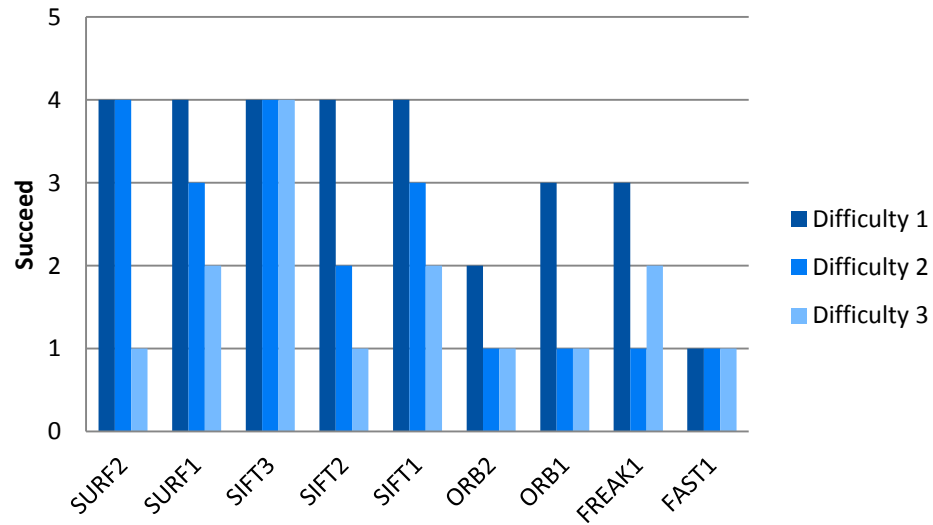
### 0,3 Megapixels Phone - Light



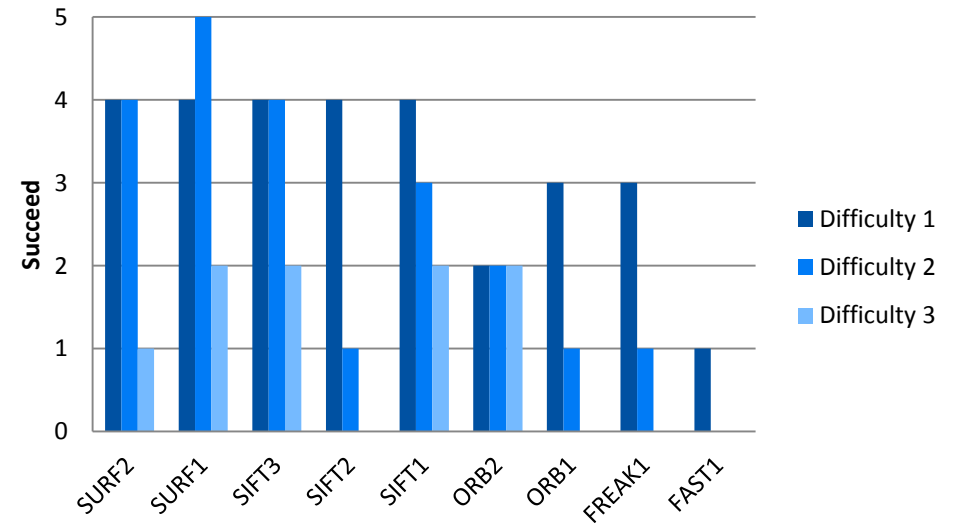
### 0,3 Megapixels Phone - Angle



### 0,3 Megapixels Phone - Rotation

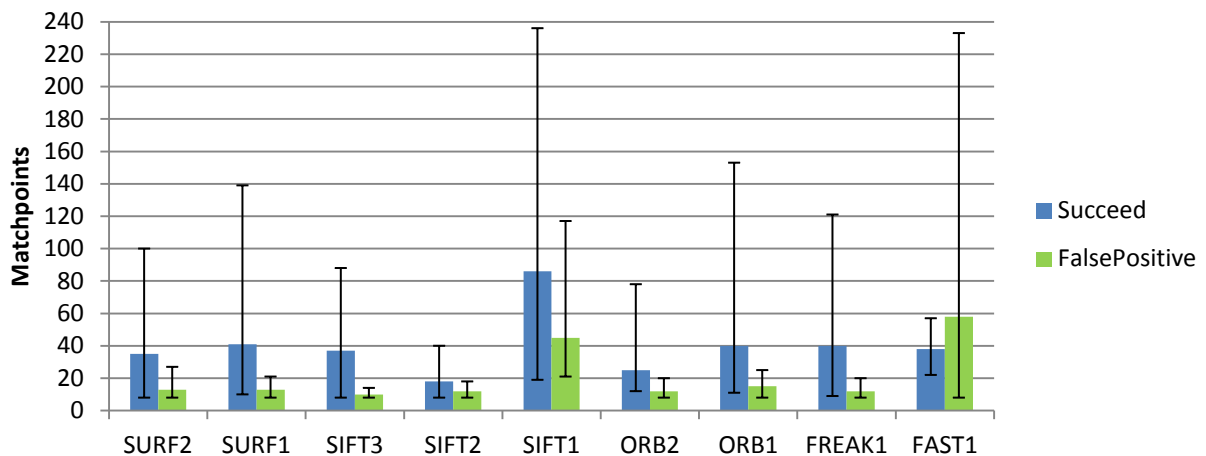


### 0,3 Megapixels Phone - Scale

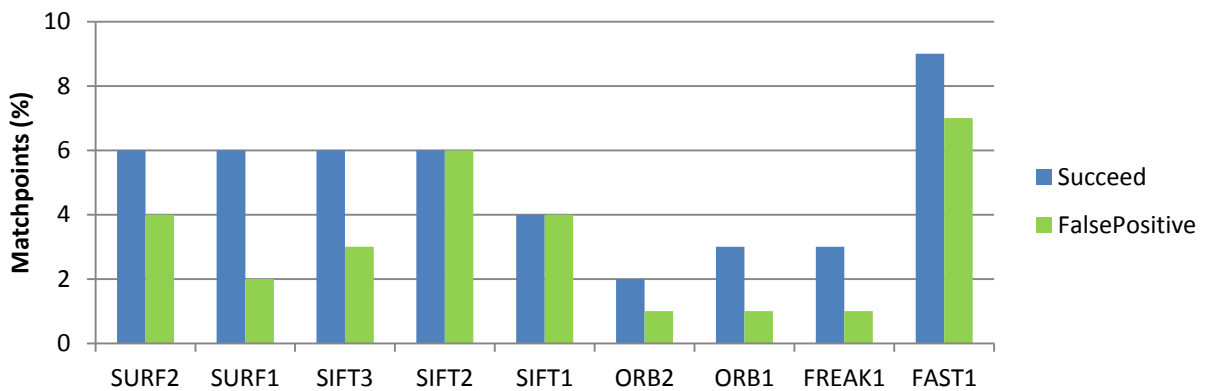


# Matchpoint results

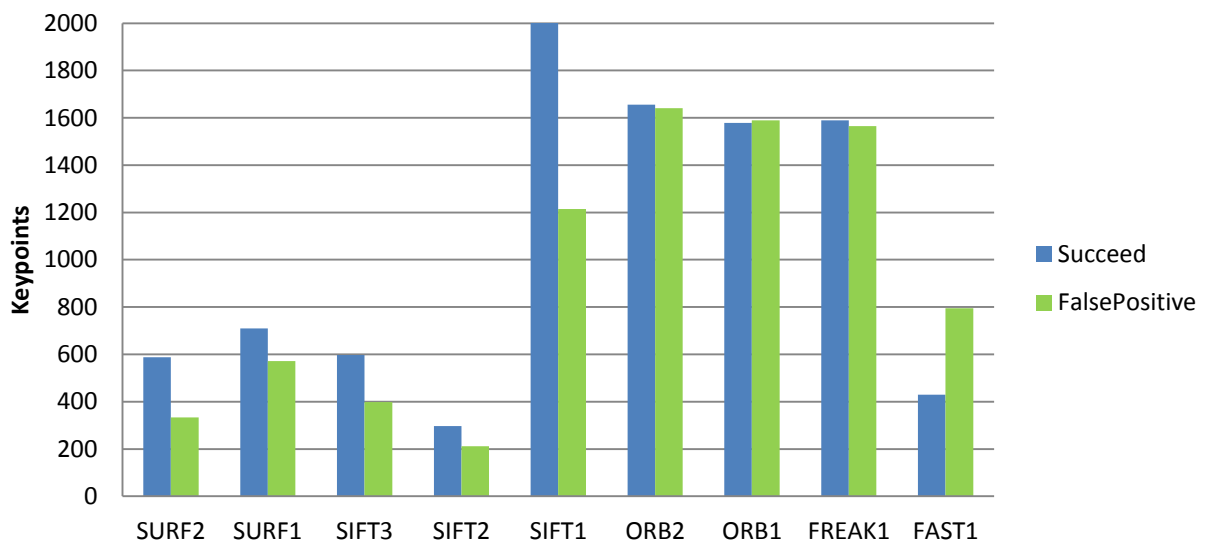
## 0,3 Megapixels Phone - Matchpoint results



## 0,3 Megapixels Phone - Matchpoints percentage results

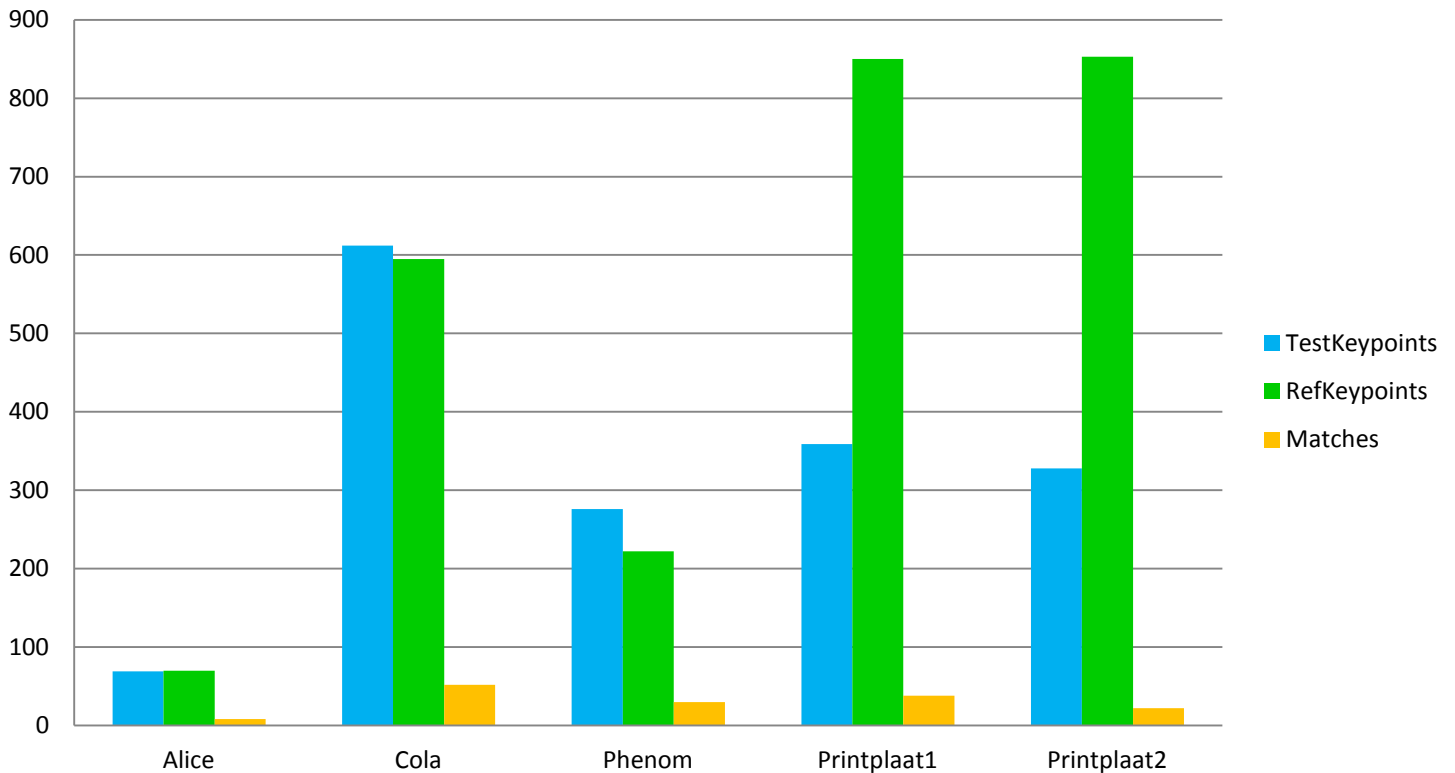


## 0,3 Megapixels Phone - RefKeypoints results



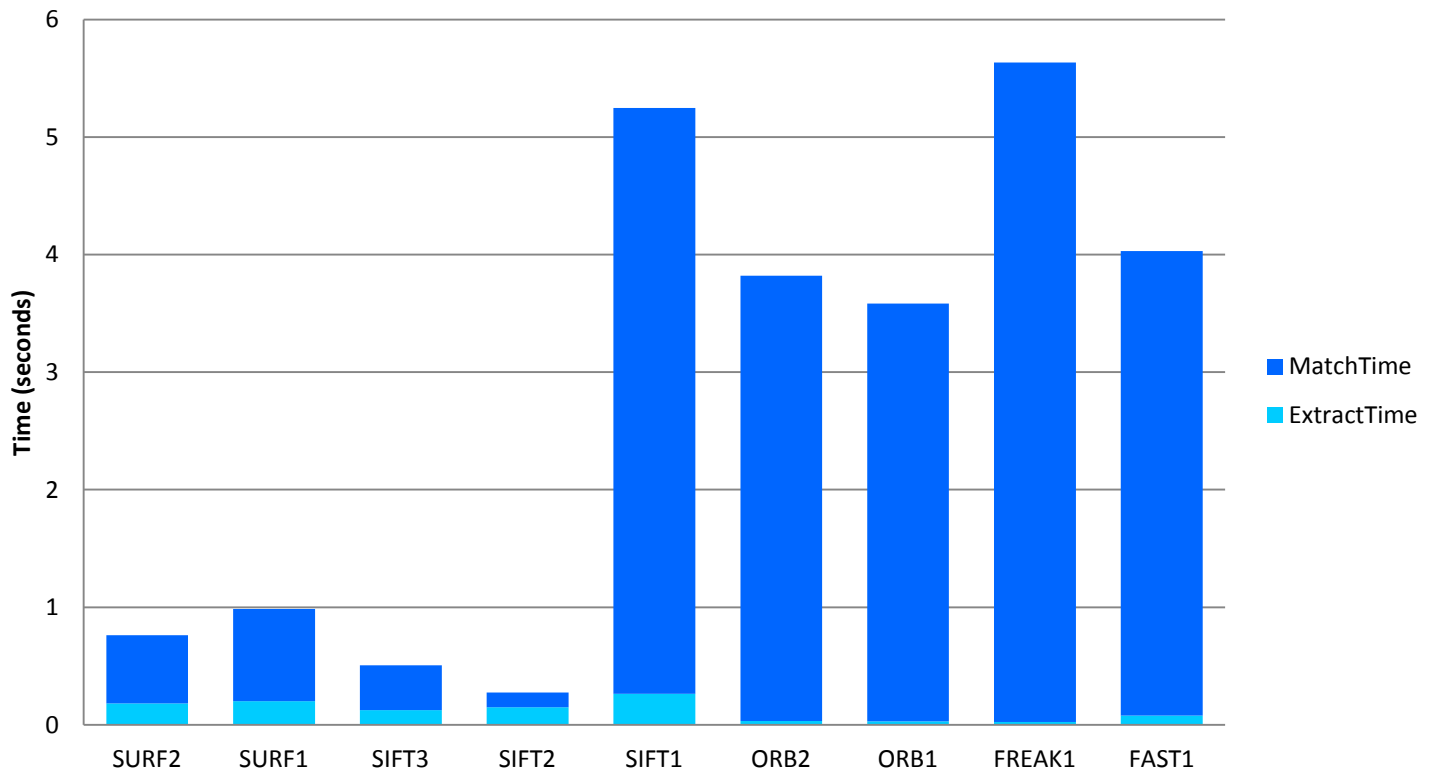
## Product results

### 0,3 Megapixels Phone - SIFT3 Products succes result



## DiffTime results

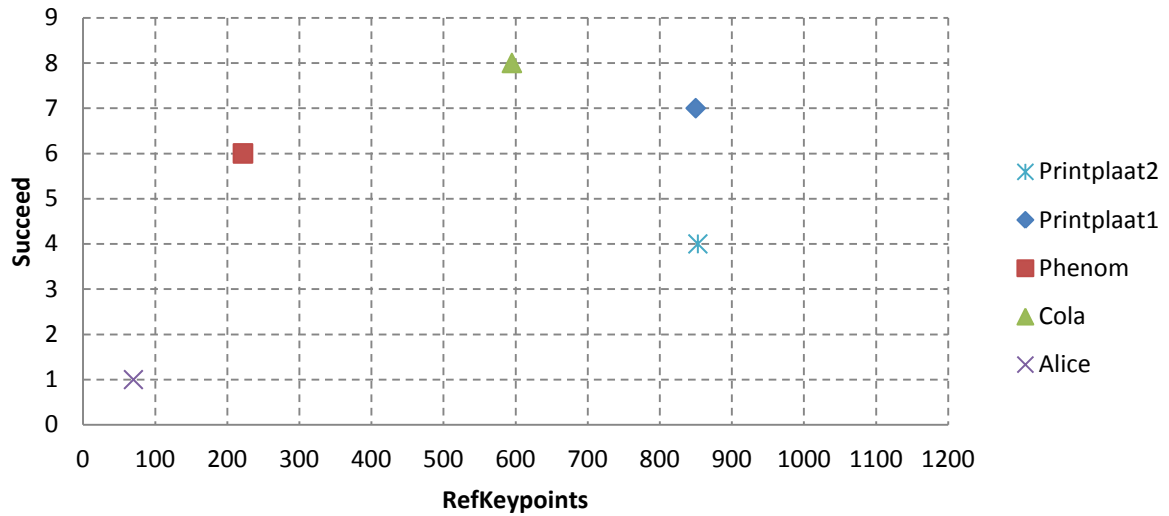
### 0,3 Megapixels Phone - DiffTime



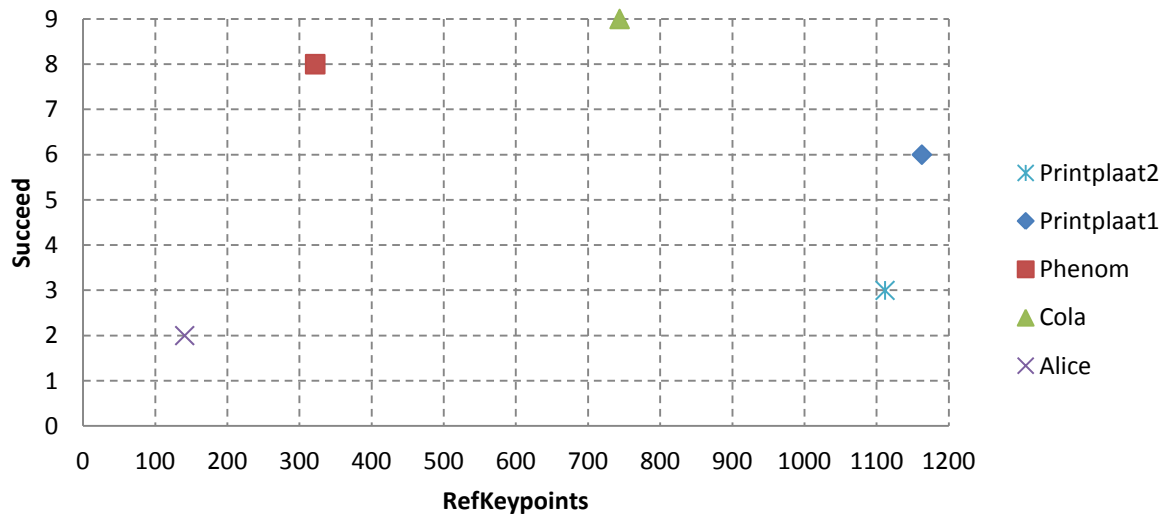


# Product succes results

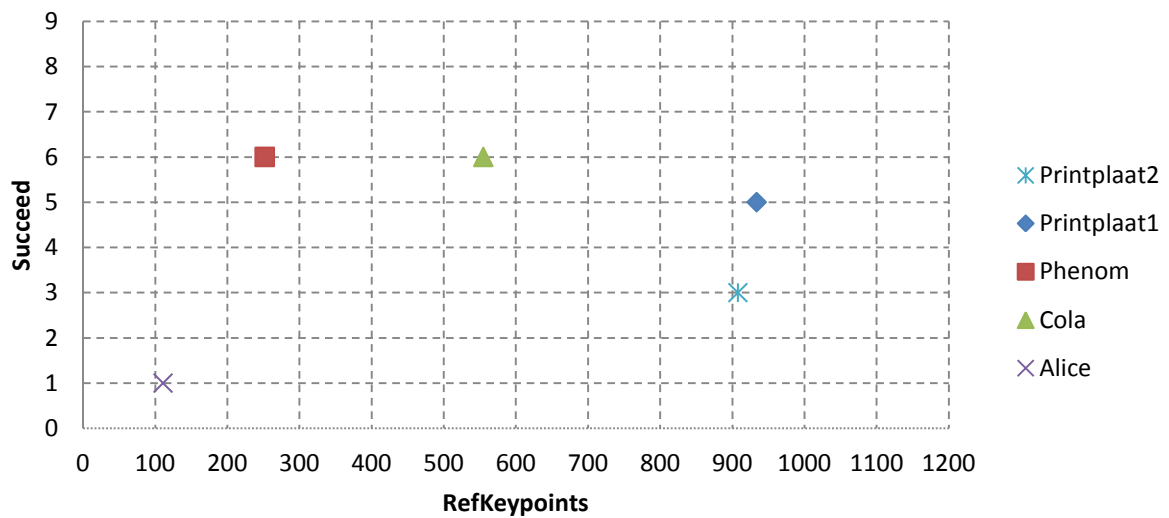
## 0,3 Megapixels Phone - SIFT3 Product succes



## 0,3 Megapixels Phone - SURF1 Product succes

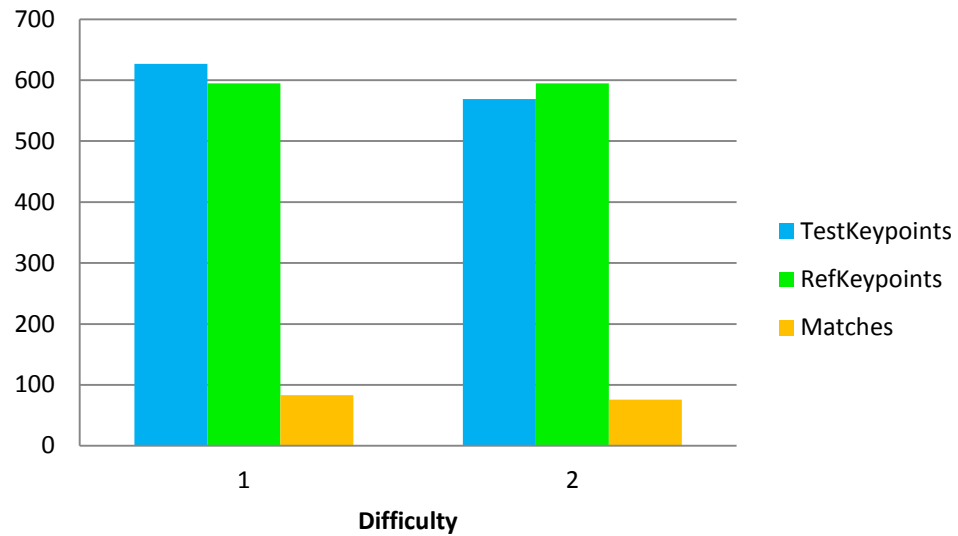


## 0,3 Megapixels Phone - SURF2 Product succes

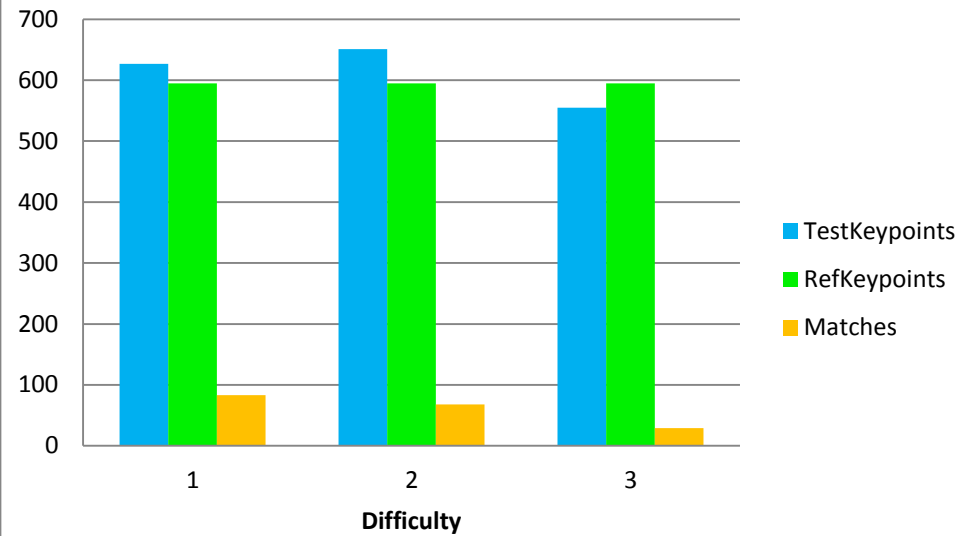


# Product difficulty results

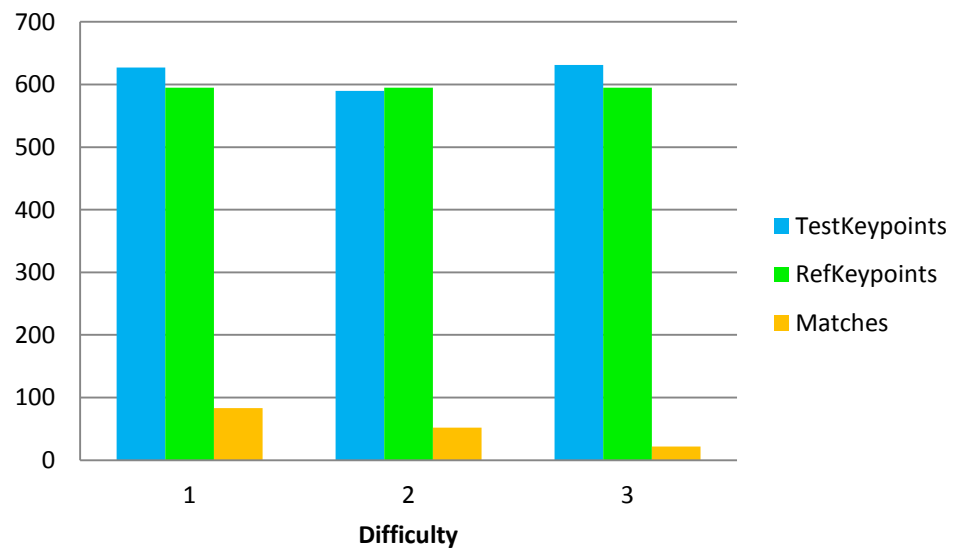
0,3 Megapixels Phone - SIFT3, Succes, Light, Cola



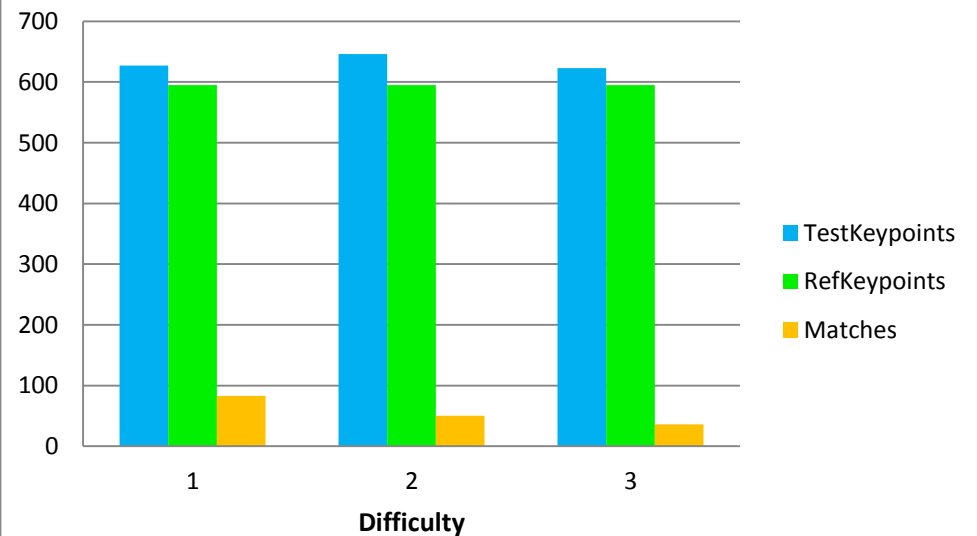
0,3 Megapixels Phone - SIFT3, Succes, Angle, Cola



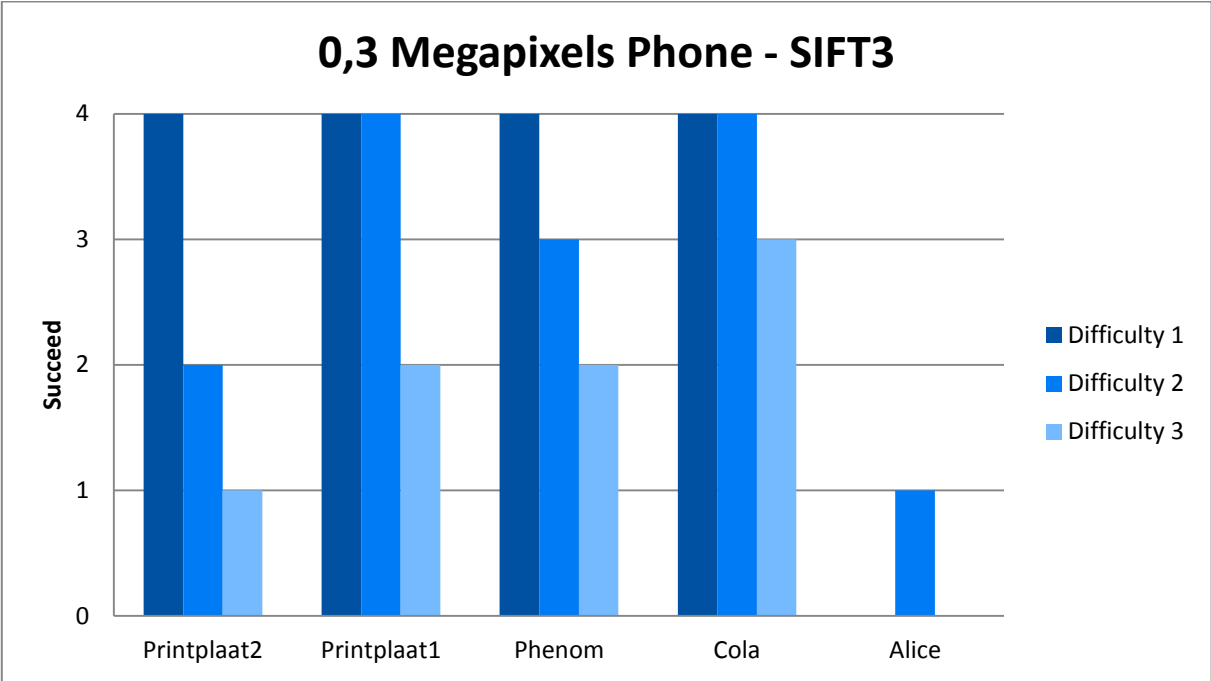
0,3 Megapixels Phone - SIFT3, Succes, Rotation, Cola



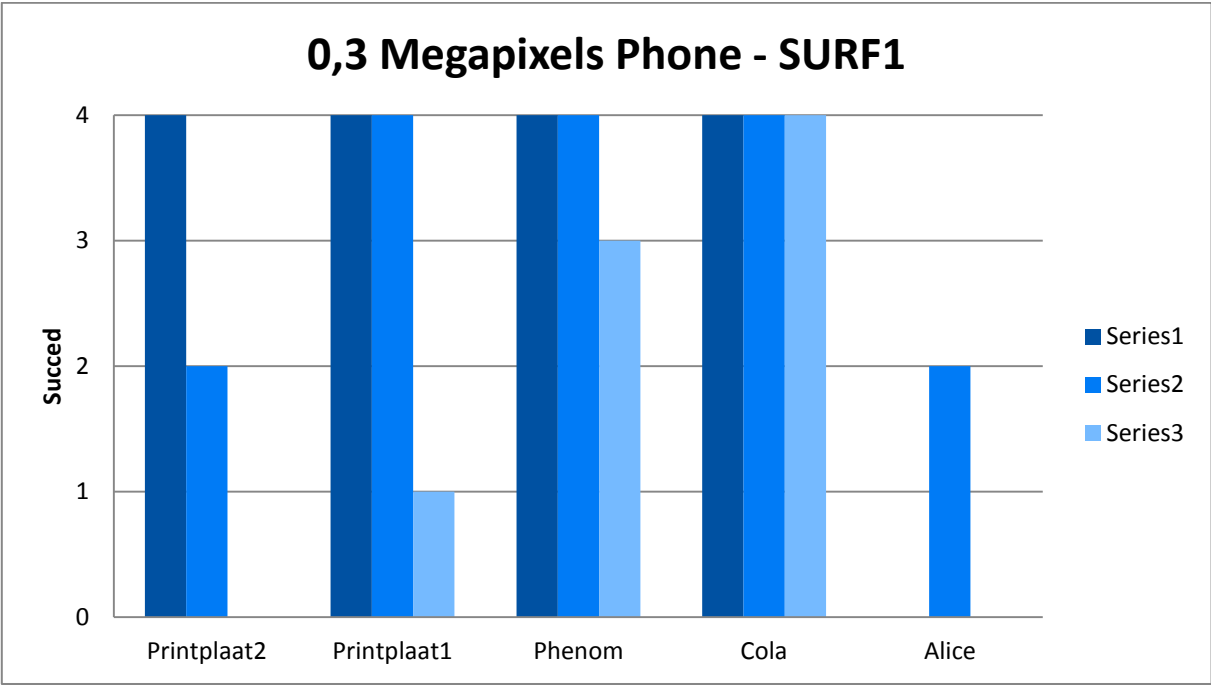
0,3 Megapixels Phone - SIFT3, Succes, Scale, Cola



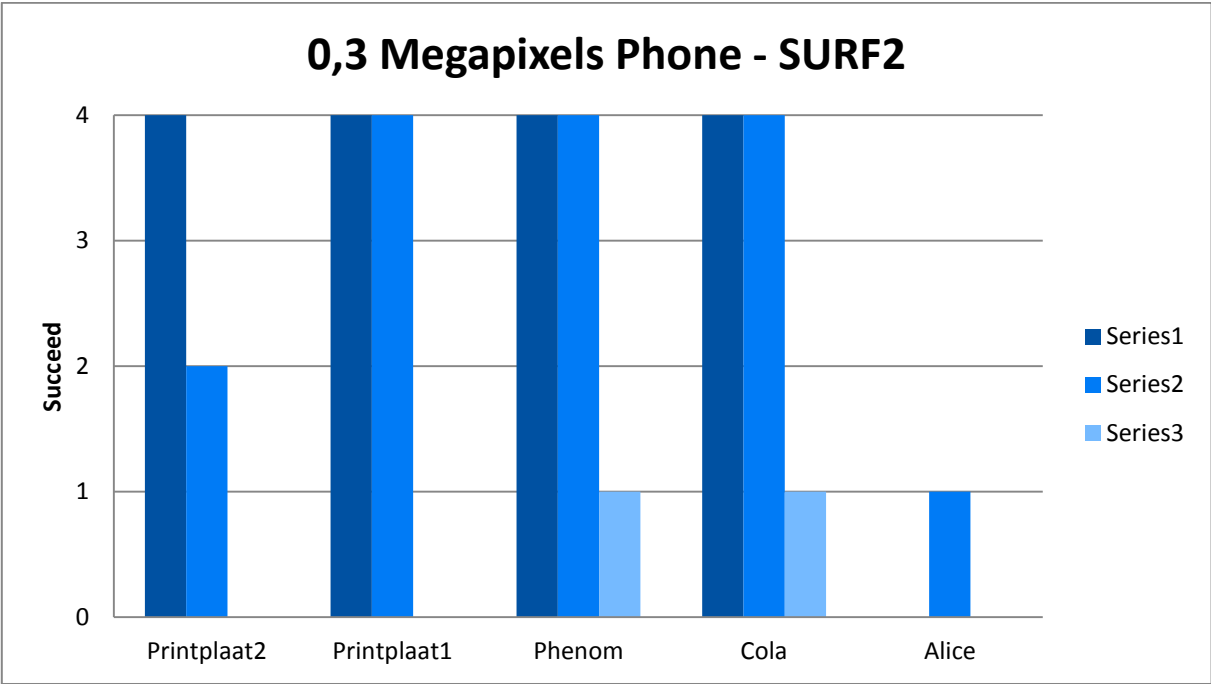
# Product case results



SIFT3 0,3 Megapixels Phone					
Product	Difficulty	Light	Angle	Rotation	Scale
Printplaat1	1	X	X	X	X
	2	X	X	X	X
	3	X		X	
Printplaat2	1	X	X	X	X
	2	X		X	
	3			X	
Phenom	1	X	X	X	X
	2	X		X	X
	3			X	X
Cola	1	X	X	X	X
	2	X	X	X	X
	3		X	X	X
Alice	1				
	2				X
	3				



SURF1 0,3 Megapixels Phone					
Product	Difficulty	Light	Angle	Rotation	Scale
Printplaat1	1	X	X	X	X
	2	X	X	X	X
	3	X			
Printplaat2	1	X	X	X	X
	2	X			X
	3				
Phenom	1	X	X	X	X
	2	X	X	X	X
	3	X		X	X
Cola	1	X	X	X	X
	2	X	X	X	X
	3	X	X	X	X
Alice	1				
	2	X			X
	3				



SURF2 0,3 Megapixels Phone					
Product	Difficulty	Light	Angle	Rotation	Scale
Printplaat1	1	X	X	X	X
	2	X	X	X	X
	3				
Printplaat2	1	X	X	X	X
	2	X		X	
	3				
Phenom	1	X	X	X	X
	2	X	X	X	X
	3			X	
Cola	1	X	X	X	X
	2	X	X	X	X
	3				X
Alice	1				
	2				X
	3				

## **BIJLAGE E**

### Testplan



# **PROJECT: EXPERIENCE CENTER APP / INITIATIEFASE**

## **TESTPLAN**



SOURCE OF YOUR TECHNOLOGY

## Documenthistorie

### Revisies

Versie	Status	Datum	Wijzigingen
0.1	Concept	16-10-2012	Geen wijzigingen
0.2	Concept	24-10-2012	Feedback stagebegeleider
0.3	Concept	25-10-2012	Feedback stagebegeleider
1.0	Release	03-01-2013	Testplan uitgevoerd

### Goedkeuring

Dit document behoeft de volgende goedkeuringen:

Versie	Datum goedkeuring	Naam	Functie	Paraaf
		Erik van Rijswik	Product owner	

### Distributie

Dit document is verstuurd aan:

Versie	Datum verzending	Naam	Functie
0.1	16-10-2012	Rico Huijbers	Stagebegeleider
0.2	24-10-2012	Rico Huijbers	Stagebegeleider
0.3	25-10-2012	Rico Huijbers	Stagebegeleider

## Inhoudsopgave

<b>1 INLEIDING</b>	<b>4</b>
<b>2 ITEM PASS/FAIL CRITERIA</b>	<b>4</b>
2.1 Algemeen	4
2.2 Pass criteria	4
2.3 Fail criteria	4
2.4 Acceptance	4
<b>3 TESTOMGEVING</b>	<b>5</b>
3.1 Server met extern scherm	5
3.2 Tablet	5
3.3 Producten van Sioux Experience Center	5
<b>4 TEST CASES</b>	<b>6</b>
4.1 Test case 1	6
4.2 Test case 2	6
4.3 Test case 3	7
4.4 Test case 4	7
4.5 Test case 5	8
<b>5 TEST COVERAGE MATRIX</b>	<b>9</b>

# 1 INLEIDING

Dit document beschrijft de test aanpak van het product. Er wordt vermeld wat er wel en niet wordt getest, hoe het wordt getest en de eisen waar de test aan moet voldoen.

## 2 ITEM PASS/FAIL CRITERIA

### 2.1 Algemeen

Bij het testen is het belangrijk om te weten wanneer een feature is geslaagd of gefaald voor een test. Om ervoor te zorgen dat iedereen die betrokken is bij het testproces deze voorwaarden weet is dit hoofdstuk geschreven.

**Belangrijk:**

Alle tests vereisen dat de pre condities precies zijn zoals beschreven in de testcase want we willen enkel de functionaliteit van een feature testen en niet de wijze waarop fouten worden afgehandeld. Als er twijfel bestaat over het slagen of falen van een test zal de stagiair dit melden bij de bedrijfsbegeleider. Samen zullen zij bepalen of de test geslaagd of gefaald is, of wat te doen.

### 2.2 Pass criteria

Een test kan als geslaagd beschouwd worden indien het resultaat van de test die werd uitgevoerd overeenkomt met het resultaat zoals verwacht.

### 2.3 Fail criteria

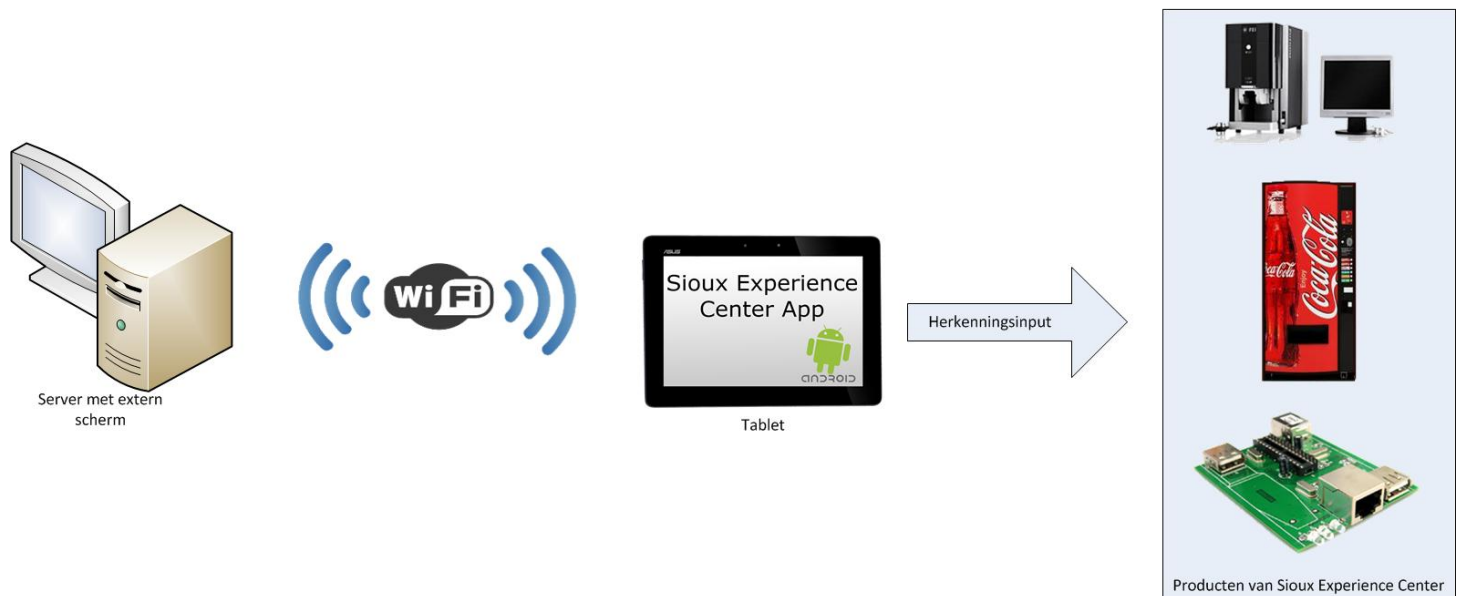
Een test kan als afgekeurd worden beschouwd indien deze niet aan de pass criteria voldoet.

### 2.4 Acceptance

Het product is geaccepteerd wanneer alle requirements die als 'MUST' staan genoteerd zijn geslaagd voor de tests cases.

### 3 TESTOMGEVING

Dit hoofdstuk beschrijft de omgeving waarbinnen de tests uitgevoerd gaan worden. De afgebeelde elementen gebruikt worden voor het uitvoeren van de tests.



Bovenstaande afbeelding geeft de gebruikte test opstelling weer. In deze test opstelling zijn diverse systemen opgenomen.

#### 3.1 Server met extern scherm

Deze server wordt gebruikt voor het afspelen van extra content, zoals een video.

#### 3.2 Tablet

Op de tablet draait een Android applicatie waarmee producten herkend kunnen worden en content weergegeven kan worden, zoals tekstuele informatie van een product.

#### 3.3 Producten van Sioux Experience Center

Dit zijn de producten in het Sioux Experience Center die herkend moeten gaan worden door de Android applicatie.

## 4 TEST CASES

### 4.1 Test case 1

Test goal ID	Test doel	
Test1	Content toevoegen	
<b>Omschrijving</b> De beheerder kan een product toevoegen aan het systeem		
<b>Requirements</b> <ul style="list-style-type: none"><li>- 1.3 Makkelijk producten beheren</li></ul>		
<b>Pre condities</b> <ul style="list-style-type: none"><li>• Het systeem is opgestart</li></ul>		
Test procedure	Verwacht Resultaat	Passed / Failed
1. Voeg herkennensinput van het te herkennen product toe aan de database met bijbehorende content (tekst en video)	1. Er wordt terugkoppeling gegeven dat de herkenningsinput en de content toegevoegd is.	

### 4.2 Test case 2

Test goal ID	Test doel		
Test2	Herken een product		
<b>Omschrijving</b> Het systeem herkend een product en laat content zien			
<b>Requirements</b> <ul style="list-style-type: none"><li>- 1.0 Product herkenning</li><li>- 1.1 PowerPoint presentatie</li><li>- 1.2 Automatische informatie</li><li>- 1.5 Direct bruikbaar</li><li>- 1.6 Android</li><li>- 1.9 Afstand</li></ul>			
<b>Pre condities</b> <ul style="list-style-type: none"><li>• De database bevat minimaal één product en bijbehorende PowerPoint presentatie</li></ul>			
Test procedure		Verwacht Resultaat	Passed / Failed
1.	Start de applicatie op	1. De applicatie is in een toestand waarin producten herkend kunnen worden.	
2.	Positioneer de tablet met een afstand van 20 cm richting het te herkennen product	2. Het object wordt herkend op de tablet en de PowerPoint presentatie wordt zichtbaar op het tablet scherm	



### 4.3 Test case 3

Test goal ID	Test doel		
Test3	Direct bruikbaar en betrouwbaarheid		
<b>Omschrijving</b> De applicatie is direct bruikbaar en betrouwbaar			
<b>Requirements</b> <ul style="list-style-type: none"><li>- 1.0 Product herkenning</li><li>- 1.1 PowerPoint presentatie</li><li>- 1.2 Automatische informatie</li><li>- 1.6 Android</li><li>- 1.7 Betrouwbaar</li><li>- 1.9 Afstand</li></ul>			
<b>Pre condities</b> <ul style="list-style-type: none"><li>• Het systeem is opgestart</li><li>• De database bevat producten en bijbehorende content</li></ul>			
Test procedure		Verwacht Resultaat	Passed / Failed
2.	Positioneer de tablet met een afstand van 20 cm richting het te herkennen product	2. Het object wordt herkend op de tablet en de PowerPoint presentatie wordt zichtbaar op het tablet scherm	
3.	Herhaal stap 2 totdat deze twintig keer is uitgevoerd	3. De applicatie herkent minstens 90% van de keer het te herkennen product	

### 4.4 Test case 4

Test goal ID	Test doel		
Test4	Nauwkeurigheid		
<b>Omschrijving</b> De applicatie is kan producten van elkaar onderscheiden			
<b>Requirements</b> <ul style="list-style-type: none"><li>- 1.0 Product herkenning</li><li>- 1.1 PowerPoint presentatie</li><li>- 1.8 Nauwkeurigheid</li><li>- 1.9 Afstand</li></ul>			
<b>Pre condities</b> <ul style="list-style-type: none"><li>• Het systeem is opgestart</li><li>• De database bevat producten en bijbehorende content</li><li>• Er liggen 2 producten 5 cm naast elkaar, deze staan allebei in de database</li></ul>			
Test procedure		Verwacht Resultaat	Passed / Failed
1.	Positioneer de tablet richting het te herkennen product met een afstand van 40 cm	1. Beide producten worden herkend, er wordt een keuze mogelijkheid zichtbaar voor de gebruiker op het tablet scherm	
2.	Selecteer een product uit de keuze mogelijkheid van de applicatie	2. Het object wordt herkend op de tablet en de PowerPoint presentatie wordt zichtbaar op het tablet scherm	

## 4.5 Test case 5

Test goal ID	Test doel		
Test5	Extern scherm		
<b>Omschrijving</b> Het systeem herkend een product en laat verschillende content zien			
<b>Requirements</b> <ul style="list-style-type: none"><li>- 1.0 Product herkenning</li><li>- 1.1 PowerPoint presentatie</li><li>- 1.2 Automatische informatie</li><li>- 1.4 Extern scherm</li><li>- 1.6 Android</li><li>- 1.9 Afstand</li></ul>			
<b>Pre condities</b> <ul style="list-style-type: none"><li>• Het systeem is opgestart</li><li>• De database bevat producten en bijbehorende content en video</li></ul>			
Test procedure		Verwacht Resultaat	Passed / Failed
2.	Positioneer de tablet met een afstand van 20 cm richting het te herkennen product	2. Het object wordt herkend op de tablet en de PowerPoint presentatie wordt zichtbaar op het tablet scherm	
3.	Selecteer afspelen video op externe scherm	3. Video wordt afgespeeld op het externe scherm	

## 5 TEST COVERAGE MATRIX

	Requirement	Test Case 1	Test Case 2	Test Case 3	Test Case 4	Test Case 5
1.0	Product herkenning		x	x	x	x
1.1	PowerPoint presentatie		x	x	x	x
1.2	Automatische informatie		x	x		x
1.3	Makkelijk producten beheren	x				
1.4	Extern scherm					x
1.5	Direct bruikbaar		x			
1.6	Android		x	x		x
1.7	Betrouwbaar			x		
1.8	Nauwkeurigheid				x	
1.9	Afstand		x	x	x	x