

Automatische detectie van patiëntengedrag in hun slaap- omgeving

Nele ANNAERT

Promotor(en): Prof. Dr. Ir Toon Goedemé

Co-promotor(en): Ir Jasmien Vanvooren

Masterproef ingediend tot het behalen van
de graad van master of Science in de
industriële wetenschappen: Elektronica - ICT
ICT

©Copyright KU Leuven

Zonder voorafgaande schriftelijke toestemming van zowel de promotor(en) als de auteur(s) is overnemen, kopiëren, gebruiken of realiseren van deze uitgave of gedeelten ervan verboden. Voor aanvragen i.v.m. het overnemen en/of gebruik en/of realisatie van gedeelten uit deze publicatie, kan u zich richten tot KU Leuven Technologicampus De Nayer, Jan De Nayerlaan 5, B-2860 Sint-Katelijne-Waver, +32 15 31 69 44 of via e-mail iiw.denayer@kuleuven.be.

Voorafgaande schriftelijke toestemming van de promotor(en) is eveneens vereist voor het aanwenden van de in deze masterproef beschreven (originele) methoden, producten, schakelingen en programma's voor industrieel of commercieel nut en voor de inzending van deze publicatie ter deelname aan wetenschappelijke prijzen of wedstrijden.

Automatische detectie van patiëntengedrag in hun slaap- omgeving

Nele ANNAERT

Promotor(en): Prof. Dr. Ir Toon Goedemé

Co-promotor(en): Ir Jasmien Vanvooren

Masterproef ingediend tot het behalen van
de graad van master of Science in de
industriële wetenschappen: Elektronica - ICT
ICT

©Copyright KU Leuven

Zonder voorafgaande schriftelijke toestemming van zowel de promotor(en) als de auteur(s) is overnemen, kopiëren, gebruiken of realiseren van deze uitgave of gedeelten ervan verboden. Voor aanvragen i.v.m. het overnemen en/of gebruik en/of realisatie van gedeelten uit deze publicatie, kan u zich richten tot KU Leuven Technologicampus De Nayer, Jan De Nayerlaan 5, B-2860 Sint-Katelijne-Waver, +32 15 31 69 44 of via e-mail iiw.denayer@kuleuven.be.

Voorafgaande schriftelijke toestemming van de promotor(en) is eveneens vereist voor het aanwenden van de in deze masterproef beschreven (originele) methoden, producten, schakelingen en programma's voor industrieel of commercieel nut en voor de inzending van deze publicatie ter deelname aan wetenschappelijke prijzen of wedstrijden.

Dankwoord

Deze masterproef vormt het besluit tot mijn opleiding. Graag zou ik enkele personen bedanken. Het is niet mogelijk om een masterproef alleen te schrijven zonder hulp. In eerste instantie gaat mijn dank uit naar mijn promotor vanuit school, Toon Goedemé. Zonder zijn hulp en commentaar had mijn werk nooit de vorm gekregen die het nu heeft. Verder wil ik ook Jasmien Vanvooren en Alphontrics bedanken voor het leveren van het onderwerp en de informatie.

Verder wil ik Maarten Vandersteegen bedanken voor het leveren van de code om de Seek Thermal ook op de computer te laten werken. Steven Puttemans wil ik ook bedanken voor het oplossen van een paar problemen in OpenCv.

Verder wil ik mijn ouders bedanken die mij dit jaar gesteund hebben met veel goede raad, het nalezen van mijn werk en mij de mogelijkheid hebben gegeven om verder te studeren.

Ik wil ook graag Thomas Van Hoof bedanken die dit jaar meerdere errors heeft helpen oplossen en mijn tekst heeft nagelezen.

Als laatste wil ik iedereen bedanken die ik niet met naam heb genoemd, maar die mij dit jaar toch gesteund en gemotiveerd hebben of aangeboden om te helpen.

Abstract

Door het opkomend succes van visie gestuurde surveillance systemen, is het interessant om hier verder onderzoek naar te doen. Deze masterproef heeft als doel de ontwikkeling van een alarm-systeem op basis van camerabeelden in bijvoorbeeld ziekenhuizen, dat automatisch een alarm laat genereren als een persoon uit bed stapt en niet binnen een bepaalde tijd terug in het bed komt. Verder moeten ook de lichten aangaan langs de kant van het bed waarlangs de persoon aanstalten maakt om uit het bed te stappen. Als een mogelijke uitbereiding kunnen we ook een alarm implementeren dat afgaat zodra een persoon uit het bed valt.

Het onderzoek gebeurt in het kader van Alphonics, een bedrijf in Lokeren dat reeds meerdere communicatie systemen beheert in de zorgsector.

Uit de literatuur blijkt dat er veel verschillende mogelijke camera's en persoon detectie algoritmes zijn. We hebben het algoritme in verschillende stappen ontwikkeld. Het eerste deel gaat een masker creëren op basis van kleursegmentatie. Nadien gaan we kijken waar het hoofd zich bevindt ten opzichte van het bed. Indien er geen persoon in het bed waargenomen wordt, gaan we zien waar de persoon zich bevindt zodat de lichten aan de juiste zijde aangaan. We maken gebruik van een Seek Thermal Compact om de beelden te maken. Ons systeem detecteert als de persoon uit het bed probeert te stappen of er geen persoon meer in het bed is en geeft een melding langs welke zijde dit gebeurt. Verder wordt er een alarm gegenereerd als de persoon te lang weg is.

Na het uitvoeren van dit onderzoek, kunnen we besluiten dat ons algoritme werkt. Verder onderzoek naar het gebruik van andere cameratypes moet nog gebeuren. De mogelijke uitbereiding van de valdetectie is eveneens nog niet gebeurd.

Abstract

Due to the upcoming success of vision controlled surveillance systems, is it interesting to conduct research in this area. This thesis has the goal to develop an alarm system designed to work with images. This system can be used in hospitals. The goal is to generate an alarm when a person comes out of bed and doesn't return in an certain amount of time. We also detect the side form which the person is leaving the bed, so the lights on that side turn on.

This research happens in collaboration with Alphatronics, a company in Lokeren that owns multiple systems in the medical sector.

recent studies shows us that there are many person detection algorithms we can use and that there are multiple types of cameras we can use. First we use colour segmentation to find the person in the images. After that we are going to determine where the head is located in the bed. If we don't detect a person in the frame, we are going to test on which side of the bed the person is, so the lights can turn on. If a person isn't back in time, the alarm is generated. Our system detects when a person is leaving the bed and also tells us on which side this happens. If the person doesn't return in time an alarm is generated.

After our research we can conclude that our system works but there is still room for improvement. The research for the use of other camera types and the fall detection have yet to happen.

Short summary

This thesis explains the development of an algorithm that a person detects. We use information from an infrared camera. The detection consists of three parts. The first part does a colour segmentation resulting in a binary image. The second part searches for the head in the image and determines its location. The third part is only used if there is no person in the bed. Then we are going to determine on which side of the bed the person is, so lights switch on at the right side. The source code is developed in C++ using the OpenCv library for the manipulations on the images.

Algorithm

The input data is delivered in a sequence of frames. Routines process one frame at a time. The algorithm consists of three parts.

Colour segmentation

The first part of the algorithm is colour segmentation. Our frames are in greyscale. We are going to segment the light pixels. These are the pixels where it is warmer. In normal situations these are the regions where a person is. The result of the segmentation is shown in image1. Before this image is created we erode and dilate the image to remove noise.

Head detection

The second part is the detection of the position of the head. We use blob detection to split the person into pieces. Then we check if there is a part of the person that is on the head part of the



Figuur 1: Mask created by colour segmentation

bed. If there is a piece of a person there, then there is someone in the bed. We use the head because when the patient uses a blanket, this is the only part that is visible.

Side of the bed and alarm generation

The third part is only necessary if there isn't a person in the bed. First we determine on which side of the bed the person is. Then we generate the alarm. If in the previous frame also wasn't a person in the bed, then the alarm value is incremented. If the value reaches a threshold the alarm will go.

Results

The detection of the person, the detection of the side and the alarm generation is satisfying. The persons are detected if they are there. If the bed is empty we get the message of the side so we know which lights to turn on. The alarm is also generated.

Inhoudsopgave

| | | |
|----------|---|-----------|
| 1 | Situering en doelstelling | 1 |
| 1.1 | Situering | 1 |
| 1.2 | Doelstelling | 1 |
| 1.3 | Overzicht | 2 |
| 2 | Literatuurstudie | 3 |
| 2.1 | Reeds bestaande systemen | 3 |
| 2.2 | Types van surveillance systemen | 3 |
| 2.2.1 | Traditionele surveillance systemen | 4 |
| 2.2.2 | Tweede generatie surveillance systemen | 4 |
| 2.2.3 | Intelligente surveillance | 4 |
| 2.3 | Verschillende types van camera's | 5 |
| 2.3.1 | Infraroodcamera | 5 |
| 2.3.2 | Night vision camera | 6 |
| 2.3.3 | IP camera | 7 |
| 2.3.4 | IP en PTZ camera's | 8 |
| 2.4 | Algoritmes voor persoonsherkenning | 9 |
| 2.4.1 | Thresholden van kleuren | 9 |
| 2.4.2 | Voorgrond- achtergrond segmentatie | 10 |
| 2.4.3 | Vormgebaseerde benadering | 19 |
| 2.5 | Besluiten uit de literatuurstudie | 19 |
| 3 | Methode en uitwerking | 20 |
| 3.1 | Werkingschema | 20 |
| 3.2 | IR camera | 21 |
| 3.2.1 | Seek Thermal Compact | 21 |
| 3.2.2 | Plaatsing van de camera | 22 |
| 3.3 | Gebruikte technieken voor de gedragsanalyse | 26 |
| 3.3.1 | Vorbereidende technieken | 27 |
| 3.3.2 | Ontwikkelen van de gedragsanalyse | 28 |
| 3.3.3 | Persoon in bed? | 29 |

| | | |
|----------|--|-----------|
| 3.3.4 | links naast bed? | 37 |
| 3.3.5 | Timer | 37 |
| 3.4 | Overige methodes | 38 |
| 3.4.1 | checkMovement | 38 |
| 3.5 | Besluit over de methode | 38 |
| 4 | Experimenten | 39 |
| 4.1 | Experiment met live beelden | 39 |
| 4.1.1 | Doel van het experiment | 39 |
| 4.1.2 | Uitwerking van het experiment | 39 |
| 4.1.3 | Besluiten na het experiment | 40 |
| 4.2 | Pixelwaarde zonder normalisatie | 40 |
| 4.2.1 | Doel van het experiment | 40 |
| 4.2.2 | Uitwerking van het experiment | 40 |
| 4.2.3 | Bespreking resultaten en besluiten na het experiment | 41 |
| 4.3 | Hoofd gebaseerde detectie inleidend experiment | 46 |
| 4.3.1 | Doel van het experiment | 46 |
| 4.3.2 | Uitwerking van het experiment | 46 |
| 4.3.3 | Resultaten en besluiten van het experiment | 46 |
| 4.4 | Hoofd gebaseerde detectie | 48 |
| 4.4.1 | Doel van het experiment | 48 |
| 4.4.2 | Uitwerking van het experiment | 48 |
| 4.4.3 | Resultaten van het experiment en besluit | 48 |
| 5 | Algemeen besluit | 51 |

Lijst van figuren

| | | |
|------|---|------|
| 1 | Mask created by colour segmentation | viii |
| 1.1 | Logo alphasatronics | 1 |
| 2.1 | Traditionele flow voor het verwerken van beelden in een surveillance systeem | 5 |
| 2.2 | Afbeelding van een night vision camera | 6 |
| 2.3 | Hardware architectuur IP camera bestaande uit 3 delen: (a) Sensor eenheid, (b) Verwerkings eenheid, (c) Communicatie eenheid. | 7 |
| 2.4 | Systeem ontwerp met twee IP camera's en een PTZ camera | 8 |
| 2.5 | Voorbeeld voorgrond/achtergrond segmentatie | 10 |
| 2.6 | Blokschema temporeel verschil van drie Frames | 16 |
| 3.1 | Werkingsschema op hoog niveau | 20 |
| 3.2 | Image of Seek Thermal Compact | 21 |
| 3.3 | Schets van camera recht boven bed | 23 |
| 3.4 | Schets van camera niet recht boven bed | 24 |
| 3.5 | Schets van camera niet bevestigd aan bed | 25 |
| 3.6 | Opgeslagen beelden Seek Thermal Compact | 27 |
| 3.7 | Foto van het bed met warme objecten op de hoeken | 28 |
| 3.8 | Flowchart van de methode setValuesAuto | 30 |
| 3.9 | Flowchart van de methode createMask | 31 |
| 3.10 | Voorbeeld van masker: (links) meegegeven afbeelding (rechts) verkregen masker . . | 32 |
| 3.11 | Verkregen masker voor leeg opgewarmd bed | 32 |
| 3.12 | Flowchart van de methode tempDifference | 33 |
| 3.13 | Voorbeeld van een bewegingsmasker | 34 |
| 3.14 | Flowchart van de methode createMaskNew | 35 |
| 3.15 | Frame en masker gemaakt met methode createMaskNew | 36 |
| 3.16 | frame (links) en bijhorend bewegingsmasker (rechts) aangepast temporeel verschil . | 36 |
| 3.17 | frame (links) en bijhorend 'foutief' bewegingsmasker (rechts) aangepast temporeel verschil | 36 |
| 4.1 | Schets experiment met live beelden | 44 |
| 4.2 | Flowchart detectie uit bed stappen met live beelden | 45 |

| | | |
|-----|---|----|
| 4.3 | frame (links) en bijhorend masker (rechts) bepaald door niet genormaliseerde waarden | 45 |
| 4.4 | Grafiek met de coördinaten van het hoofd (eerste set frames), met op de horizontale as de framenummers, de verticale as de pixelwaarden | 47 |
| 4.5 | Grafiek met de coördinaten van het hoofd (tweede set frames), met op de horizontale as de framenummers, de verticale as de pixelwaarden | 47 |
| 4.6 | Grafiek met de coördinaten van het hoofd (derde set frames), met op de horizontale as de framenummers, de verticale as de pixelwaarden | 47 |
| 4.7 | Grafiek met de coördinaten van het hoofd (vierde set frames), met op de horizontale as de framenummers, de verticale de waarden | 49 |
| 4.8 | Flowchart hoofdgebaseerde detectie | 50 |

Lijst van tabellen

| | | |
|-----|---|----|
| 3.1 | Berekende waarden voor camera niet recht boven hoofdeinde | 25 |
| 3.2 | Berekende waarden camera niet aan bed bevestigd | 26 |
| 4.1 | Gemeten niet genormaliseerde waarden leeg bed, eerste poging | 41 |
| 4.2 | Gemeten niet genormaliseerde waarden bed met persoon, eerste poging | 42 |
| 4.3 | Gemeten niet genormaliseerde waarden leeg bed, tweede experiment | 42 |
| 4.4 | Gemeten niet genormaliseerde waarden bij een bed met persoon zonder laken, tweede experiment | 43 |
| 4.5 | Gemeten niet genormaliseerde waarden bij een bed met persoon met laken, tweede experiment | 44 |

Hoofdstuk 1

Situering en doelstelling

Dit hoofdstuk heeft een inleidende functie. In sectie 1.1 bespreken we de situering van deze masterproef. De doelstelling wordt besproken in sectie ???. Op het einde van dit hoofdstuk, in sectie ???, formuleren we een overzicht van de verdere inhoud van de masterproef.

1.1 Situering



Figuur 1.1: Logo alphontronics

Alphontronics is een bedrijf uit Lokeren met meerdere jaren ervaring als fabrikant in toegangscontrole. Dit bedrijf is actief in verschillende sectoren waaronder zorgcentra en ziekenhuizen. Alphontronics is momenteel volop aan het innoveren in de zorgcentrum. Het is het kader van deze innovatie dat het onderzoek naar het patiënten gedrag in hun slaapomgeving is gedaan.

1.2 Doelstelling

Deze masterproef behandelt een onderzoek naar het gebruik van camera beveiliging in een slaapomgeving van patiënten in de zorgsector. Om het comfort van de patiënten te bevorderen en het verplegend personeel te ontlasten gaan we een systeem ontwikkelen dat detecteert wanneer een patiënt uit bed probeert te stappen. Als dit gebeurt begint er een timer te lopen en gaan de lichten aan langs de zijde van het bed waar de patiënt zich bevindt. Indien de persoon niet op tijd terug in bed is, gaat er bij de nachtverpleegster een alarm. Dan weten zij dat er iets mis is. Deze masterproef bestaat uit volgende activiteiten

- Literatuurstudie
- Bestuderen en ontwikkelen van de gedragsanalyse

- Experimenten

De onderzoeks vraag luidt als volgt: Is het mogelijk om door middel van een visiesysteem, te detecteren langs welke zijde patiënten uit bed stappen en hoelang ze wegblijven? Een belangrijke beperking op ons systeem is dat de patiënten onherkenbaar moet zijn op de beelden.

1.3 Overzicht

De masterproef is als volgend geconstrueerd: eerst geven we meer informatie over reeds bestaande systemen, verschillende types van surveillance systemen, verschillende types van camera's en algoritmen voor persoonsdetectie in hoofdstuk 2. Hoofdstuk 3 bestaat uit een verduidelijking van het werkingsschema, de gebruikte code en de gebruikte camera. Nu we de methode besproken hebben gaan we verder met de verklaring van de verschillende experimenten in hoofdstuk 4. We sluiten deze masterproef af met een besluit in hoofdstuk ??.

Hoofdstuk 2

Literatuurstudie

In dit hoofdstuk gaan we nakijken of er reeds werken geschreven zijn die voor ons onderzoek van toepassing zijn. Voor we beginnen aan het maken van het project, gaan we eerst op zoek naar reeds bestaande systemen, deze worden beschreven in sectie 2.1, we bekijken welke types van surveillance systemen er bestaan en bestuderen welke eventueel interessant kunnen zijn voor ons project in sectie 2.2, dit wordt gevolgd door sectie 2.3, deze omvat een beschrijving van verschillende types van camera's. Als laatste deel van de literatuurstudie bestuderen we reeds bestaande algoritmes voor persoonsherkenning in sectie 2.4. Dit hoofdstuk wordt afgesloten in sectie 2.5 met een besluit over de literatuurstudie.

2.1 Reeds bestaande systemen

Het eerste wat we onderzochten was of er al gelijkaardige systemen bestonden. Onze zoekacties op het internet leverden aanvankelijk geen resultaten op. Al blijkt dat er in bijvoorbeeld de slaapkliniek van A.Z. Monica ook gewerkt wordt met een infraroodcamera. Daarover was verder geen informatie te vinden. Gedurende de rest van het jaar bleven we verder zoeken naar soortgelijke systemen, omdat de mogelijkheid altijd bestaat dat er nieuwe informatie beschikbaar komt.

2.2 Types van surveillance systemen

In dit deel van de literatuurstudie gaan we onderzoeken welke types van surveillance systemen er zijn. We gaan eveneens bestuderen welke types zinvol zijn voor ons project. Waarom zouden we werken met camerabeveiliging? Je zou ook een systeem kunnen maken door gebruik te maken van druk sensoren in een mat naast het bed, of onder de matras zelf. Het voordeel van camera gestuurde surveillance is dat we kunnen zien wat er op het moment zelf gebeurt, maar dat we ook achteraf gebeurde incidenten kunnen reconstrueren. Het nadeel van camera gestuurde surveillance is dat we incidenten moeten kunnen identificeren en herkennen op het moment dat ze gebeuren [30]. Er zijn drie verschillende generaties van surveillance systemen. De eerste generatie van surveillance systeem dat we gaan bestuderen is het traditionele surveillance systeem, meer informatie hierover is terug te vinden in sectie 2.2.1, daarna gaan we verder met de tweede generatie van surveillance systemen in sectie 2.2.2, dit wordt gevolgd in sectie 2.2.3 door een bespreking van de intelligente surveillance systemen. Een surveillance systeem kan door middel van verschillende parameters geëvalueerd worden. Bijvoorbeeld door een studie van de tijdsvertraging

tussen het voorvallen van een gebeurtenis en de detectie. Er wordt momenteel ook nog steeds onderzoek gedaan op gebied van de verschillende types van camera gestuurde surveillance.

2.2.1 Traditionele surveillance systemen

De traditionele surveillance systemen zijn de eerste van drie generaties van surveillance systemen. Ze maken gebruik van analoge technologie doorheen het volledige systeem, dit is een van de grootste nadelen van dit type van systeem [22]. Analoge camera's maken in dit type van systeem, de beelden van de scène. De signalen worden verzonden over communicatielijnen naar de toestellen, waar de data weergegeven en opgeslagen worden [20]. In een traditioneel surveillance systeem is de kwaliteit en de kost recht evenredig met het aantal gebruikte camera's. Voor een grote omgeving, worden er voor een systeem meerdere camera's gebruikt, om een hele omgeving te bestuderen. Bijgevolg wordt er een grote hoeveelheid aan data verzameld die achteraf verwerkt moet worden. In onze toepassing, moet er enkel het bed bestudeerd worden. Hierdoor hebben wij een kleine omgeving en volstaat het om één camera te hebben voor de detectie van het patiënten gedrag. Indien er ook valdetectie gedaan wordt, zijn er twee camera's nodig om een dieptebeeld te creëren. Meestal is er een persoon nodig die altijd naar de monitor kijkt [32]. Omdat wij meerdere patiënten in het oog willen houden, is dit voor ons niet mogelijk. In sommige gevallen geeft deze methode zeer goede resultaten.

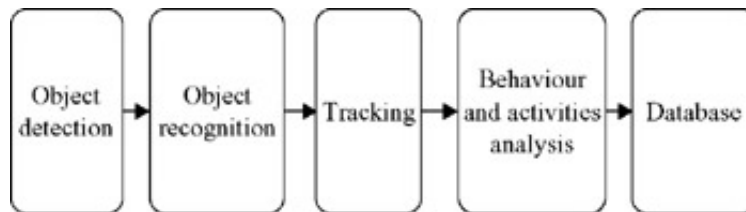
2.2.2 Tweede generatie surveillance systemen

In de tweede generatie van surveillance systemen wordt er gebruik gemaakt van digitale componenten. De overstap van analoge naar digitale technologie is nog niet compleet. Er wordt een combinatie van beide technologieën gebruikt. Automatische visuele surveillance wordt hier mogelijk door een combinatie van computer visuele technologie met Closed Circuit Television (CCTV) systemen [22]. Hierdoor is de detectie van gebeurtenissen eenvoudiger voor de gebruiker. Deze dient eens het programma geschreven is en de detecties automatisch gebeuren, zelf niets meer te doen. De alarmtijd wordt korter. Doordat een computer de detectie doet, en niet meer een persoon, kunnen we elke kamer uitrusten met een surveillance systeem. Bij traditionele surveillance systemen zou dit gelijk zijn als voor elke kamer een persoon te betalen om de kamers te bewaken. Door de automatisatie kunnen we meerdere kamers in het oog gehouden worden. De kwaliteit van het surveillance systeem is beter [20], dit doordat de reactie tijd hoger is. Eén van de nadelen is dat je algoritmes nodig hebt voor de gedragsanalyses [22]. Het ontwikkelen van deze gedragsanalyses gaat tijdens de ontwikkel fase veel tijd in beslag nemen. Eens deze ontwikkelt zijn en goede resultaten hebben, gaan zij bijdragen tot de betere resultaten van het surveillance systeem ten opzichte van de traditionele systemen. Het ontwikkelen van dergelijk gedragsanalyse omvat het grootste deel van het onderzoek dat in dit werk gedaan wordt.

2.2.3 Intelligente surveillance

Dit is het derde en eveneens laatste generatie van beeld gestuurde surveillance systemen. Hier is de overgang naar een volledig digitaal systeem compleet [20]. Daarom wordt het vaak een intelligent surveillance systeem genoemd. Het kan automatisch zeer grote regio's verwerken. Een groot voordeel is dat er verschillende soorten van sensoren kunnen gecombineerd worden waardoor er zeer veel informatie beschikbaar is. Het opzetten van een dergelijk systeem is moeilijker dan het opzetten bij de voorgaande generaties. men moet het systeem trainen om de incidenten te

detecteren. Men kan dit zowel manueel als computergestuurd doen. Hoe meer data je nodig hebt om het systeem te trainen, hoe meer men het proces moet automatiseren [26]. Het verwerken van de beelden die gemaakt worden gebeurt in verschillende stappen. Welke stappen deze zijn en in welke volgorde ze gebeuren, zijn te zien in figuur 2.1 [22].



Figuur 2.1: Traditionele flow voor het verwerken van beelden in een surveillance systeem

2.3 Verschillende types van camera's

In het tweede deel van de literatuurstudie doen we onderzoek naar verschillende types van camera's. En bekijken we of deze al dan niet bruikbaar zijn voor ons project. Waarom zijn niet alle camera's geschikt? Aangezien het project gaat over het bestuderen van het slaapgedrag van patiënten, gaan onze beelden gemaakt worden in een donkere omgeving. Hierdoor zal er met sommige camera's bijna niets te zien zijn. Het eerste type van camera dat we bespreken is de infraroodcamera in sectie 2.3.1, dit type camera werkt met warmte-verschillen. Een ander type van camera dat we mogelijk ook kunnen gebruiken is de night vision camera, deze wordt besproken in sectie 2.3.2. Dit is een type van camera speciaal ontworpen om in donkere omgevingen te werken. Een derde mogelijk type van camera wordt besproken in sectie 2.3.3, namelijk een IP-camera. Nadat we de werking van de IP camera bekijken, gaan we deze uitbereiden met PTZ-camera's, dit is terug te vinden in sectie 2.3.4.

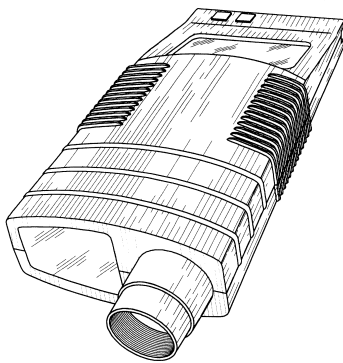
2.3.1 Infraroodcamera

We hebben het vermoeden dat de infrarood (IR) camera interessant kan zijn omdat deze warmte-verschillen weergeeft. In tegenstelling tot camera's die opereren in de zichtbare band van het spectrum werkt de IR camera met een lange golflengte (8-12 μm). Een IR sensor gaat de elektromagnetische golven, die zich in het spectrum van de camera bevinden, uitgestraald door objecten in een scène, als een thermische afbeelding weergeven,. Waarvan elke pixelwaarde een temperatuur voorstelt [11]. We gaan dus de persoon, indien deze zich binnen het bereik van de camera bevindt waarnemen, ongeacht of het dag of nacht is. Dit komt doordat de temperatuur van een patiënt, normaal gezien, hoger is als deze van de omgeving [18]. De eerste bedenkingen hierbij zijn of het mogelijk is om een persoon te detecteren als deze gebruik maakt van een elektrisch warmtedeken. Verder gaat het bed ook opwarmen. We moeten dus ook bekijken of er geen valse persoonsdetecties gebeuren op de warme lakens als de persoon in kwestie niet meer in het bed ligt. Verder is de camera ongevoelig voor lichtinval. Hierdoor gaat het bijvoorbeeld geen schaduwen maar ook geen kleuren van kleding waarnemen. Een voordeel van dit type camera is dat een persoon al onherkenbaar is op de beelden. Dit is één van de vereisten van ons systeem. Een probleem doet zich voor wanneer een patiënt naar de badkamer gaat en er vervolgens een ander persoon de kamer binnen komt. Dit zal in de beelden zeer weinig verschil veroorzaken en er kun-

nen valse detecties gebeuren. Een ander probleem dat opduikt is dat er kringen verschijnen rond gebieden met een zeer hoge of zeer lage temperatuur [8]. De sterkte en grootte van deze kringen is afhankelijk van het actuele temperatuurverschil tussen de persoon en de omgeving, en de contrast/gain instellingen van de camera [10]. Een van de grote voordelen is de uitzonderlijk lage signal to noise ratio (SNR). In het begin werd de IR camera voornamelijk in militaire toepassingen gebruikt. Door de daling van de prijs van een infrarood camera wordt deze tegenwoordig steeds vaker gebruikt in verschillende toepassingen zoals bijvoorbeeld industriële inspectie, door de politie en ook in surveillance systemen[10]. Op het internet zijn er ook veel verschillende manieren te vinden om van een oud fototoestel zelf een infraroodcamera te maken. Al blijken deze niet altijd goed te werken. Dankzij de vele toepassingen en de dalende prijs, zijn er ook veel bedrijven die met IR camera's op de markt komen die door ons gebruikt kunnen worden. Hieronder is een lijst te vinden met een paar voorbeelden van dergelijke bedrijven.

- Heimann Sensor
- Excelitas met CoolEye sensor
- Panasonic in samenwerking met AS electronic
- Infinitegra
- Seek Thermal

2.3.2 Night vision camera



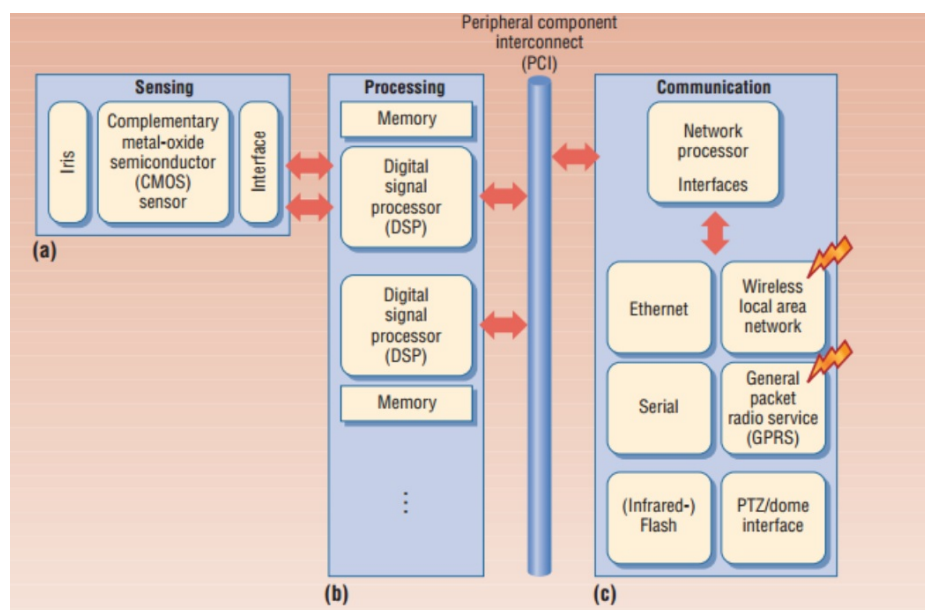
Figuur 2.2: Afbeelding van een night vision camera

Een ander type van camera dat ook mogelijkheden biedt, is een night vision camera. Een afbeelding van zo een camera, is te vinden in figuur 2.2[15]. Dit type van camera wordt gebruikt in een omgeving waar onvoldoende licht is voor een standaardcamera en wordt voornamelijk bij zonsopgang, 's nachts of zonsopgang gebruikt. Dit type van camera wordt in veel toepassingen gebruikt waaronder surveillance, astronomie en gedragsonderzoek van dieren [28]. Aangezien wij 's nachts onze beelden maken en deze zouden gebruiken voor een surveillance systeem, vallen wij in het gebruiksgebied van deze camera. Er wordt gebruik gemaakt van SWIR (Short Wavelength InfraRed), met een golflengte van $(1-1.7 \mu m)$. Deze SWIR gaat het overgebleven atmosferisch licht, dat 5 tot 7 keer groter is dan het licht van de sterren, waarnemen. Het maken van de beelden in SWIR gebeurt op een vergelijkbare manier dan in een standaardcamera, het is het resultaat

van een reflectie principe . De dure versies gaan beelden gemaakt met LWIR (Lange golflengte IR) combineren met SWIR beelden [28]. Het nadeel van deze camera is dat de persoon wel herkenbaar is op de beelden die rechtstreeks komen van de camera. Terwijl in de vereisten van ons systeem staat dat de persoon onherkenbaar moet zijn op de beelden. Dit kan natuurlijk achteraf door middel van beeldverwerking aangepast worden, door bijvoorbeeld een masker te creëren door huidsegmentatie toe te passen op de beelden. Deze camera's zijn duurder dan de eerder besproken IR camera's in sectie 2.3.1.

2.3.3 IP camera

IP staat voor Internet Protocol. Het is een camera voorzien van de mogelijkheid om verbinding te maken met het internet. De camera functioneert op je computer netwerk. De camera wordt aangesloten aan een router, switch of draadloos netwerk. De camera combineert de traditionele camera en netwerk video technologie. De IP camera kan live video comprimeren in de camera zelf wordt de video data omgezet naar digitale data, die al voor een deel verwerkt is. Enkel een abstracte beschrijving van het beeld wordt verder gestuurd [21]. De data wordt over het internet verzonden zonder daarvoor een computer te gebruiken. De basis architectuur van de IP camera ziet u in figuur 2.3 [20]. De camera is opgebouwd uit drie delen. Het eerste deel is de sensor



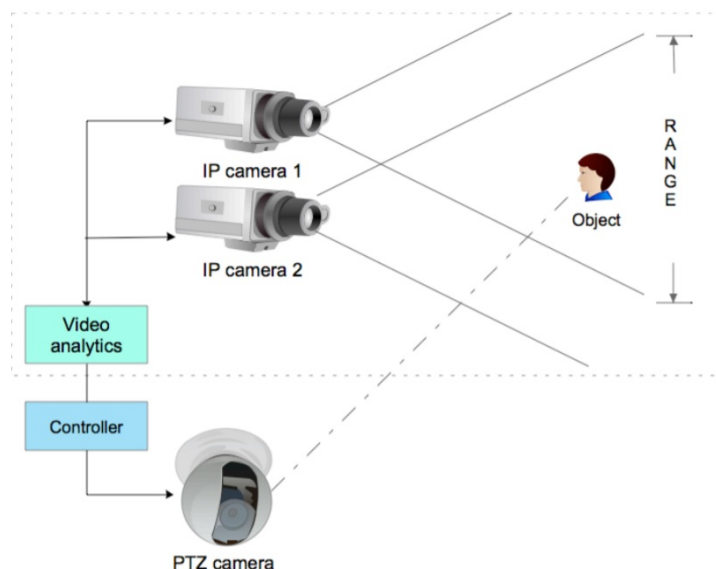
Figuur 2.3: Hardware architectuur IP camera bestaande uit 3 delen: (a) Sensor eenheid, (b) Verwerkings eenheid, (c) Communicatie eenheid.

eenheid, deze bestaat meestal uit een zeer dynamische monochrome complimentary metal-oxide semiconductor (CMOS) afbeelding sensor. In sommige gevallen komen hier ook andere sensoren bij. Het tweede deel van de architectuur, is de afhandelingseenheid. Deze bestaat uit Digital Signal Processor (DSP). Zij bieden een goede compromis tussen prestatie, vermogen verbruik en flexibiliteit. De verbinding met de netwerk processor wordt tot stand gebracht door Perpiheral Component Interconnect (PCI) bus. De netwerk processor gaat zorgen voor de interconnectie tussen de communicatie- en afhandelingseenheden. Indien er nog verdere afhandeling van de beelden nodig is, zijn er een paar vaste stappen: object detectie en herkenning, volgen, gedragsanalyse en

ophalen van de data [22]. Er zijn verschillende soorten van IP camera's, je hebt er voor gebruik binnen en buiten, of speciaal uitgerust voor gebruik 's nachts, al dan niet met infrarood. De camera die wij nodig hebben is er één voor binnen, die uitgerust is voor 's nachts. Het prijskaartje hiervan kan al snel oplopen tot een paar honderden euro's. Dit type camera wordt vooral gebruikt voor bewakingssystemen, ook in de medische sector, verder vormen ze ook de kern van surveillance in de toekomst.

2.3.4 IP en PTZ camera's

De uitbereiding van twee IP camera's met een PTZ camera wordt gedaan, om een dieptebeeld te kunnen vormen. Dit is nodig voor een mogelijke uitbereiding van dit onderzoek, namelijk de detectie van een patiënt die uit bed valt. We beginnen met twee IP camera's om de positie en locatie van een persoon te bepalen, deze worden automatisch uit de beelden van de camera's gehaald. De positie en locatie worden dan doorgestuurd naar een Pan-Tilt-Zoom (PTZ) camera. De camera wijst en zoomt naar de juiste locatie in de ruimte. In ons geval dus naar de patiënt. Hierdoor wordt van de persoon een goed kwalitatief beeld gevormd. Het ontwerp van zo een systeem is te zien in figuur 2.4 [24]. De grootste moeilijkheid bij dergelijke constructie is dat je de goniometrische relatie tussen de verschillende camera's moet kennen. Om ervoor te zorgen dat dit in orde is kan je een kalibratie doen. Het is door deze goniometrische relatie dat de coördinaten bepaald worden waar de PTZ camera's op gaan inzoomen. [30]. Een groot nadeel van deze opstelling is natuurlijk de extra kost van een IP camera en een PTZ camera. We moeten ook de afweging maken of het verkrijgen van een nauwkeuriger beeld van de patiënt door de PTZ camera het maken van de extra kost waard is. Anders zouden wij ook de opstelling zonder PTZ camera kunnen gebruiken om de diepte te kunnen bepalen voor de valdetectie.



Figuur 2.4: Systeem ontwerp met twee IP camera's en een PTZ camera

2.4 Algoritmes voor persoonsherkenning

In dit hoofdstuk bespreken we algoritmes voor persoonsherkenning. Deze hebben we nodig om de patiënt uit het frame te halen. Er zijn veel verschillende methodes ontwikkeld. In dit hoofdstuk bespreken we er een paar die we onderzocht hebben. De eerste techniek is het thresholden van kleuren, alle informatie hierover is te vinden in sectie 3.3.3. Hierna wordt background estimation techniek of voorgrond- achtergrondsegmentatie besproken in sectie 3.3.3. Deze techniek vaak gebruikt voor het volgen van mensen en bewegingen. Ze zijn vooral bruikbaar bij een stationaire camera [23]. Aangezien wij in ons onderzoek een stationaire camera hebben, is dit voor ons een zeer interessante techniek. Een nadeel is dat elke vorm van verandering wordt gedetecteerd als de patiënt, waardoor het soms kan lijken dat er meerdere patiënten in een kamer zijn en er dus valse detecties kunnen optreden. Een oplossing hiervoor is een vorm gebaseerde benadering, die we zullen bespreken in sectie 2.4.3 [23]. In de meeste gevallen worden verschillende van deze methoden gecombineerd op uiteindelijk goede resultaten te krijgen.

2.4.1 Thresholden van kleuren

Dit algoritme is gebaseerd op de kleuren dat een object heeft op de beelden. Deze techniek kan voor veel applicaties gebruikt worden, zoals het detecteren van personen, zoals ook onze bedoeling is, of bijvoorbeeld het detecteren van het rood uit verkeersborden. Deze methode is pixel gebaseerd, voor elke pixel wordt de kleurwaarde vergeleken met een regio in het kleurenspectrum. Zo kunnen we in onze toepassing pixel per pixel zien of het al dan niet gaat om een persoon [31]. Er zijn verschillende systemen om kleuren te benoemen, bijvoorbeeld:

RGB Rood Groen en Blauw

CMY Cyaan, Magenta and Yellow: De primaire kleuren van de kleurpigmenten

HSI Hue, Saturation and Intensity

YUV Y = helderheid en UV chrominantie (kleurcomponenten)

We kunnen in het algoritme spelen met de verschillende systemen, zo kunnen we zien waar de persoon het makkelijkst uit het beeld te halen is. Indien we dit algoritme gebruiken, gaan we gebruik maken van OpenCV. Om de ruis achteraf te gaan verminderen, kunnen we erosie en dilatie, een median filter of een andere methode gebruiken.

Jones and Rehg

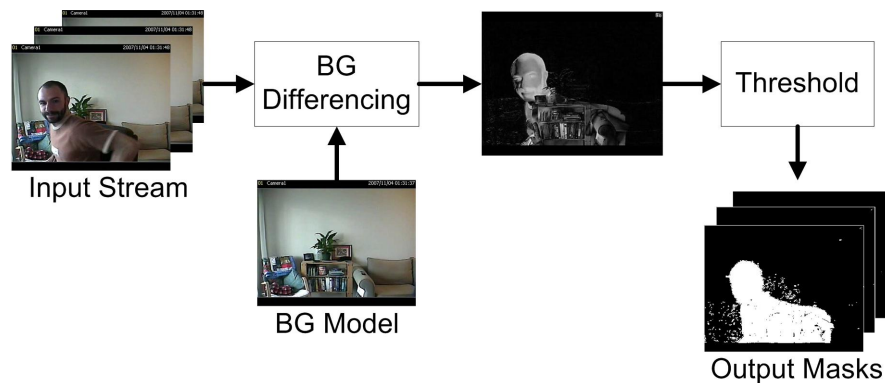
Ze maakten een model op basis van internet afbeeldingen. Ze onderzochten welke kleuren in welke verhoudingen voorkomen in de huidskleur op de verschillende afbeeldingen. Nadat dit model af was hebben ze een ander model toegevoegd dat huid- en niet huidafbeeldingen scheidt. Ze maakten een huidhistogram waarin ze de kleur verhoudingen van huid gelabelde afbeeldingen staken, ze maakten ook een niet-huidhistogram waarin ze de kleur verhoudingen van niet-huid gelabelde afbeeldingen staken. Dankzij deze histogrammen kan men de kans berekenen dat huid voorkomt in een bepaalde pixel [31]. Doordat wij onze beelden 's nachts maken en er dan weinig lichtinval is, gaan in onze toepassing, de kleurwaarden anders zijn. Verder gaan wij ook op beelden gemaakt met een infraroodcamera geen huidskleur zien. Bijgevolg is dit dus een techniek die niet van toepassing is voor ons onderzoek.

Dawod et al.

Dit model staat ook gekend als het RGB-H-CbCr model. Dit is een pixel gebaseerde methode. De beelden worden bekeken in drie verschillende kleurruimtes. De eerste kleurruimte is het RGB domein. Vervolgens wordt gebruik gemaakt van het HSV-domein en als laatste het YUV-domein. Een andere naamgeving voor het YUV domein is H-Cb-Cr. Hierbij geeft H de waarde van de gereflecteerde luminantie. Cb staat voor chrominantie blauw en geeft het verschil aan tussen een referentiewaarde en de blauwe component. Cr is chrominantie rood en geeft het verschil weer tussen een referentiewaarde en de rode component. Deze kleurruimte wordt typisch gebruikt voor huiddetectie [1]. In elke kleurruimte wordt er beslist of een pixel al dan niet huid is. Achteraf worden de drie ruimtes samengevoegd en besloten of een pixel al dan niet huid is, de huidpixels moeten een aaneensluitende regio vormen [31]. Doordat we onze beelden 's nachts maken gaan de waarden van de kleuren veranderd zijn. Hierdoor kunnen we ook deze techniek niet gebruiken voor onze toepassing.

2.4.2 Voorgrond- achtergrond segmentatie

Deze techniek wordt ook background estimation genoemd. De techniek is gebaseerd op het feit dat een beeld bestaat uit een achtergrond die niet verandert en een voorgrond die dus alle bewegende en veranderde delen bevat. Eens de achtergrond bepaald is, wordt deze van het nieuwe frame afgetrokken. De verschilwaarde wordt dan vergeleken met een drempelwaarde. Als de verschilwaarde groter is dan de drempelwaarde, wordt ze op de resultaten afbeelding op 1 gezet, indien ze kleiner is dan de drempelwaarde wordt die pixel op de resultaten afbeelding 0. Ze krijgt men een binair beeld. Dit is terug te vinden in figuur 2.5 [25].



Figuur 2.5: Voorbeeld voorgrond/achtergrond segmentatie

Voorgrond- achtergrond segmentatie wordt vaak gebruikt in video surveillance [6]. Deze techniek kan je splitsen in twee verschillende types namelijk recursief en niet-recursief. Eerst bespreken we de recursieve technieken. Dit wordt gevolgd door de niet-recursieve technieken. Elk van deze technieken kan op verschillende manieren gebeuren en heeft zijn voordelen en nadelen op vlak van geheugen, snelheid en moeilijkheid. De eerste stap van de deze techniek is het bepalen van de achtergrond. Dit kan op drie verschillende manieren gebeuren, namelijk per pixel per regio en per frame [17]. Dit gebeurt in de achtergrondmodelleringsfase. De voorgrond- achtergrond segmentatie techniek heeft 2 verschillende nadelen. Het eerste nadeel is dat hij licht gevoelig is, als je het licht aan doet, verandert het beeld. Hierdoor kunnen in bepaalde toepassingen valse detecties optreden [17]. Het tweede nadeel is dat de techniek gevoelig is voor de beweging van de camera.

Je kan deze techniek dus enkel toepassen op beelden gefilmd met een statische camera [14] of door de beweging in het algoritme te reduceren. Een derde nadeel is dat het niet mogelijk is om goede resultaten te bekomen bij een niet statische achtergrond of in geval van ruis [4]. Als allerlaatste bespreken we een manier om voorgrond/achtergrond segmentatie toe te passen op thermische beelden [9],[12],[19]. Vaak worden verschillende technieken gecombineerd om betere resultaten te krijgen.

Voorgrond/achtergrond segmentatie technieken moeten met 3 zaken rekening houden om succesvol te zijn [5]:

- Wat is het model en hoe gedraagt het zich?
- Hoe wordt het model geïnitieerd?
- Hoe wordt het model geüpdatet?

Doordat in onze toepassing de camera en de achtergrond statisch is. En de beelden 's nachts gemaakt worden, waardoor er weinig invloed is van het licht. Zijn de drie grote nadelen van deze methode hier niet van toepassing. Hierdoor kunnen we mogelijk één van de verschillende soorten voorgrond- achtergrond segmentatie gebruiken in onze toepassing.

Recursieve technieken

Het achtergrond model wordt op recursieve wijze bij elke frame geüpdatet. Dit wordt eveneens een adaptieve techniek genoemd. Omdat de achtergrond die bij het eerste frame geschat wordt, bij elke frame aangepast wordt, totdat de achtergrond bijna niet meer wijzigt. Het voordeel hiervan is dat je geen extra frames moet bewaren in een buffer. Het nadeel hiervan is dat als er zich een fout voordoet in het achtergrond model dit ook voor lange tijd in het achtergrond model bewaard blijft. We bespreken vier verschillende recursieve methodes om de achtergrond te bepalen. De eerste techniek is running average, de tweede is de approximated median filtering gevolgd door mixture of gaussian en om af te sluiten bespreken we $\Sigma - \Delta$ achtergrond segmentatie.

Running average Het achtergrond model wordt voor deze techniek bepaald aan de hand van volgende formule [12]:

$$A_{i+1} = \alpha F_i + (1 - \alpha)A_i$$

met:

A_i De nieuwe achtergrond die bepaald wordt

A_i De huidige achtergrond

α De leersnelheid, vaak 0.05, met een waarde tussen 0 en 1

F_i Het huidige frame

Dit wilt zeggen dat het achtergrond model gemaakt wordt door voor elke pixel de gemiddelde waarde te nemen van de waarde in de vorige achtergrondschatting en de huidige framewaarde. De achtergrond gaat dus mee veranderen in de loop van de tijd.

Men kan de voorgaande formule generaliseren door het te schrijven in incrementele vorm [16]:

$$A_{i+1} = A_i + \delta_i F_i$$

met:

A_i De nieuwe achtergrond die bepaald wordt

A_i De huidige achtergrond

δ_i increment functie, afhankelijk van het huidige frame

F_i Het huidige frame

De voordelen van deze methode zijn

- Eenvoudige berekeningen die niet veel CPU tijd vragen
- Er wordt niet veel geheugen gevraagd
- De bepaling van de achtergrond gaat heel snel in verhouding tot andere technieken
- De methode is eenvoudig te implementeren.
- Houdt rekening met veranderingen in de achtergrond

De nadelen van deze methode zijn

- Het resultaat is een schatting en is dus niet precies
- α moet geregeld worden om zo nauwkeurig mogelijke resultaten te krijgen.
- De nauwkeurigheid van de achtergrond-schatting is afhankelijk van de snelheid waarmee het object door het beeld beweegt.

Er is ook een niet recursieve tegenhanger, deze noemt men exponential smoothing [16]. Doordat een persoon tijdens het slapen niet veel beweegt, gaat op de plaats waar de persoon slaapt, de persoon als achtergrond gelabeld worden. Als de persoon vervolgens beweegt, gaat er een geest, een persoon die er niet is, ontstaan op de slaapplek. Wij gaan op dat moment twee personen detecteren terwijl er maar één is. Als we gebruik maken van deze techniek, moeten we in het algoritme met deze geest rekening houden en zien dat we de reële persoon gebruiken.

Approximated Median Filtering Deze methode heeft ook een niet recursieve tegenhanger. Omdat ze zo'n goede resultaten opleveren en de berekening zeer eenvoudig is. Het principe is als volgt: als de grijswaarde van de pixel die men bekijkt groter is dan de grijswaarde van de geschatte achtergrond, wordt de grijswaarde van deze pixel in de geschatte achtergrond geïncrmenteerd met 1. Dezelfde redenatie voert men ook uit als de grijswaarde van de pixel kleiner is dan de grijswaarde van de overeenkomstige pixel in de geschatte achtergrond. De waarden van de achtergrond gaan convergeren naar de effectieve mediaan [26]. Deze techniek werkt omdat men ervan uit gaat dat de achtergrondpixels gedurende het grootste deel van de tijd zichtbaar zijn. Daardoor is de effectieve mediaan de achtergrond. Dit resulteert in weinig rekenwerk.

De voordelen van deze methode zijn:

- Zeer eenvoudige implementatie
- Hoge nauwkeurigheid

- Niet veel geheugen nodig
- De techniek is robuust

De nadelen van deze techniek zijn:

- Als er een wijziging is in de achtergrond, past hij zich maar traag aan

Deze techniek heeft hetzelfde nadeel als de voorgaande, de persoon wordt tijdens het slapen als achtergrond aangeduid. Als de persoon dan gaat bewegen wordt er een geest gecreëert. Het achtergrond model gaat zich wel aanpassen en de geest zal verdwijnen, maar het gaat lang duren.

Mixture of Gaussian Mixture of gaussian ook wel MOG genoemd, zal aan de hand van een model bepalen of een pixel tot de voor- of achtergrond hoort. Als de waarde van de pixel in het huidige frame hoort binnen de waarden van de achtergrond, zal deze tot de achtergrond gerekend worden. Het updaten van de achtergrond is afhankelijk van de waarschijnlijkheid van de waarden van het nieuwe frame. De waarden waarmee geïncrementeerd worden zijn eveneens afhankelijk van de dichtheid van de achtergrond. Als de dichtheid groter is, zal de incrementeer waarde ook groter zijn [16]. Het aantal Gausische distributies die gebruikt worden tijdens de berekening, weergegeven door k , is afhankelijk van de hoeveelheid beschikbaar geheugen.

De formule is hieronder te vinden [3]:

$$P(X_t) = \sum_{i=1}^k w_{i,t} \cdot \eta(X_t, \mu_{i,t}, \Sigma_{i,t})$$

met:

k Het aantal Gausische distributies

$w_{i,t}$ een schatting van het gewicht van de i^{de} Gaussian op ogenblik t

$\mu_{i,t}$ de gemiddelde waarde van de i^{de} Gaussian op ogenblik t

$\Sigma_{i,t}$ De covariantie matrix van de i^{de} Gaussian op ogenblik t

η Gausische probaliteits functie

X_t De pixelwaarde op het ogenblik t

De formule van de Gausische probaliteitsfunctie, is hieronder weergegeven [3]:

$$\eta(X_t, \mu_{i,t}, \Sigma_{i,t}) = \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma_{i,t}|^{\frac{1}{2}}} e^{-\frac{1}{2}(X_t - \mu_{i,t})^T \Sigma_{i,t}^{-1} (X_t - \mu_{i,t})}$$

met:

X_t De pixelwaarde op het ogenblik t

$\mu_{i,t}$ de gemiddelde waarde van de i^{de} Gaussian op ogenblik t

Σ De covariantie matrix van de i^{de} Gaussian op ogenblik t

T Schatting van de minimale hoeveelheid data dat tot de achtergrond gerekend wordt

Voordelen van de methode:

- Er is geen vaste drempelwaarde
- Deze methode is nauwkeuriger dan de running average en approximated median filtering methodes die eerder beschreven werden.
- Kan tegen lange termijn veranderingen en herhalende bewegingen.

Nadelen van deze methode:

- Deze methode is traag.
- Veel moeilijke berekeningen nodig om de achtergrond te bepalen

De nadelen wegen zwaarder op dan de voordelen. Daardoor wordt deze techniek minder vaak gebruikt [16], [3].

$\Sigma - \Delta$ achtergrond segmentatie De $\Sigma - \Delta$ achtergrond segmentatie is een eenvoudige, niet lineaire methode om de achtergrond te bepalen [16]. Het wordt vaak gebruikt in ingebedde toepassingen [5]. Om de achtergrond te bepalen wordt aangenomen dat de achtergrond voor het grootste deel van de tijd zichtbaar is [27]. Het is gebaseerd op vergelijken en elementaire optellingen en verschillen. Door middel van een functie bepaalt men de waarschijnlijkheid van de waarde van de pixel van het volgende frame, dit noemt men het dichtheidsmodel. Hoe hoger de waarschijnlijkheid van een pixel, hoe kleiner de nood om het te gaan updaten, hierdoor is de geschatte achtergrond stabiel. [27] Men gaat dan de pixelwaarde vergelijken met de waarde van het dichtheidsmodel en de waarde van de achtergrond aanpassen. Een mogelijke manier om de geschatte waarde te bepalen is door gebruik te maken van de Zipf law, dit is een hyperbolisch afnemende functie. Ook hier treedt het aperture probleem op, dit wilt zeggen dat de middelste delen van bewegende delen niet mee gedetecteerd worden. De relevantie van deze techniek is te vergelijken met die van MOG [16].

Voordelen van de methode [27],[2]:

- Eenvoudig te berekenen
- robuuste en accurate detectie van bewegende objecten

Nadelen van de methode [2]:

- Veel parallele berekeningen nodig
- Trage beweging, kan mee tot achtergrond gerekend worden

Het tweede nadeel van deze methode, geeft eveneens de reden dat deze methode voor onze toepassing niet in aanmerking komt.

Niet recursieve technieken

De niet-recursieve technieken maken gebruik van een sliding window benadering voor het bepalen van de achtergrond. De laatste frames worden opgeslagen in de buffer en de achtergrond wordt bepaald door variatie van elke pixel in de tijd. Het voordeel van deze methode is dat ook personen die lang in dezelfde houding op het bed liggen ook gedetecteerd worden. Het nadeel hiervan is dat de buffer een groot deel geheugen in beslag neemt. Er zijn verschillende methodes om het achtergrond model te bepalen. We bespreken er drie. Als eerste bespreken we temporeel verschil, de tweede methode is temporeel verschil met drie frames, als laatste bespreken we median filter.

Temporeel verschil Dit is een zeer eenvoudige model. Deze methode wordt gebruikt om bewegende objecten uit een reeks van afbeeldingen te halen. De gedetecteerde oppervlakken omvatten de doel oppervlakken en kleine niet-interessante oppervlakken (bijvoorbeeld kleine veranderingen in de achtergrond) [33]. Je neemt twee opeenvolgende frames, neemt daarna het verschil, dit verschil wordt dan vergeleken met een drempelwaarde. [22]. Aangezien een patiënt in zijn slaap heel weinig gaat bewegen, is dit voor ons geen goede methode. Een mogelijke oplossing hiervoor is een vlag zetten als de persoon het frame binnen komt en de vlag te verwijderen als de persoon uit het frame verdwijnt. Een nadeel van deze techniek is dat men gebruik maakt van een sequentie van beelden en dat de detectie dus een paar frames later gebeurt. [23]. De techniek kan zeer goed tegen dynamische veranderingen van de omgeving. Maar geeft niet alle relevante pixels als oplossing, er ontstaan gaten in bewegende objecten [7].

$$|frame_i - frame_{i-1}| < T$$

met:

frame_i het huidige frame

frame_{i-1} het vorige frame

T de drempelwaarde

Als het verschil groter is dan de drempelwaarde, dan hoort die pixel bij de voorgrond [6]. De drempelwaarde wordt empirisch bepaald.

Voordelen [33]:

- Weinig rekenkracht nodig.
- De methode is zeer adaptief.
- Er wordt zeer weinig extra geheugen gebruikt, doordat er twee frames gebruikt worden met men enkel het vorige bijhouden.
- Het model wordt snel bijgewerkt als er iets verandert in de achtergrond.

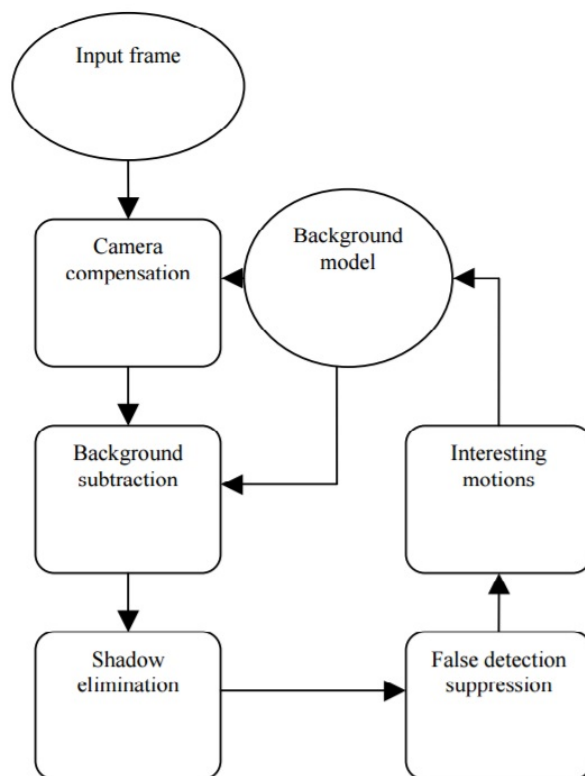
Nadelen [33]:

- Als een persoon niet beweegt, verdwijnt deze mee in de achtergrond.
- Als een object een gelijkmatig verdeelde intensiteit waarden heeft, de middelste delen mee verdwijnen in de achtergrond

- Deze methode is zeer gevoelig voor ruis.
- Het is moeilijk om de juiste drempelwaarde te vinden.
- Een kleine verandering in de achtergrond wordt mee als voorgrond gezien

Deze techniek is goed om de bewegende delen in het frame te vinden. Met deze methode gaan we geen hele persoon detecteren. Via deze methode kunnen we wel elke beweging van de patiënt detecteren.

Temporeel verschil met drie frames Deze methode is een uitbereiding op de methode die we in de vorige paragraaf hebben besproken. Het grootste verschil is dat er drie opeenvolgende frames gebruikt worden om de achtergrond te berekenen. Het blok diagram van deze methode is te vinden in figuur 2.6 [33]. Zoals in de figuur te zien is, zijn er verschillende stappen in het algoritme. Hieronder gaan we ze stap voor stap uitleggen.



Figuur 2.6: Blokschema temporeel verschil van drie Frames

Camera compensatie Hier worden eventuele kleine bewegingen van de camera gecompenseerd. Omdat temporeel verschil heel gevoelig is aan kleine veranderingen in het beeld. Dit gebeurt door middel van een algoritme dat enkele standaard pixels in het frame neemt en vervolgens de beweging schat door middel van een matching methode. Vervolgens wordt het volledige inputframe aangepast. Indien men werkt met een statische camera, zoals wij in onze applicatie, hoeft men deze stap niet te doen [33].

Achtergrond bepaling Vervolgens wordt het temporeel verschil gebruikt om snel de bewegende objecten uit het input beeld te halen. De drie frames worden verdeeld in twee groepen. De eerste groep bevat de twee voorgaande frames, de tweede groep bevat de input frame en het voorgaande frame. Van deze twee groepen wordt dan het temporeel verschil zoals besproken in de voorgaande paragraaf berekend. Vervolgens kan men door de intersectie te nemen van de twee berekende verschillen eenvoudig de beweging berekenen van het vorige frame, deze komt namelijk in de twee berekeningen voor. Vervolgens neemt men het verschil van het temporeel verschil van de tweede groep en de juist berekende intersectie. Nu houd je de bewegende delen over van het laatste frame.

Schaduweliminatie en valse detectie onderdrukking De stappen na de berekening van het temporeel verschil worden gebruikt om de twee problemen die optreden door het gebruik van de methode ongedaan te maken. De eerste is dat het onmogelijk is om altijd het volledig bewegende object te detecteren, dit doordat er kleine gaten ontstaan in het bewegend object door de berekening van het temporeel verschil. Het tweede is dat deze techniek heel gevoelig is voor kleine veranderingen en er dus ook oninteressante regio's zijn die gedetecteerd worden [33].

Algemene voor en nadelen Deze techniek gaat een accuratere achtergrond bepalen dan het klassieke temporeel verschil. Het gaat wel veel meer rekenwerk vragen. Verder gaat er meer geheugen gebruikt worden.

Median Filter Deze techniek wordt zeer vaak toegepast. Hij zegt dat de mediaan van elke pixel van de n laatste frames in de buffer de achtergrond is. De redenatie hierachter is dat de achtergrond in de helft van de frames ongewijzigd gaat blijven.

Voordelen:

- Weinig rekenkracht voor nodig
- Levert nauwkeurige resultaten

De nadelen van deze methode zijn

- Veel geheugen capaciteit nodig, omdat je elke keer de volledige frames moet bijhouden

Per pixel proces

Men kijkt naar elk pixel signaal als een onafhankelijk proces. Deze methode wordt tegenwoordig het meeste gebruikt [5], omdat men hiervoor weinig rekenkundige kracht nodig heeft. Al de methodes die hierboven besproken werden zijn per pixel processen. Men vergelijkt de achtergrond pixel per pixel met het huidige frame. Bij het gebruiken van deze techniek, wordt er eveneens vanuit gegaan dat er een ruimtelijke consistentie is. Dit kan resulteren in lokale mis classificaties [5].

Per regio proces

Men gaat de regio rond de pixel mee bekijken. Dit wordt gedaan om valse positieve signalen te minimaliseren. Deze methode levert een robuustere versie op van het model. Men neemt aan dat de omgeving van een pixel ook tot het zelfde object in de achtergrond hoort. Er wordt ook vanuit

gegaan dat als de waarde van een pixel verandert, er soortgelijke veranderingen optreden in de aanliggende pixels. Daarom vergelijkt men de ogenblikkelijke pixel waarde met de waarden van de omgeving in de achtergrond. Deze methode zorgt wel voor problemen bij pixels waarvan de omgeving op de rand ligt met meerdere achtergrond object [5].

Per frame proces

Bij deze methode kijkt men naar het gehele frame. Dit wordt gebruikt om bijvoorbeeld de gevoeligheid voor het aandoen van het licht te verkleinen. Men gebruikt deze techniek om plotse en globale veranderingen te detecteren [5]. In onze toepassing zou dit bijvoorbeeld het verschijnen van een patiënt in het frame zijn, nadat deze uit het frame verdwenen was.

Voorgrond/Achtergrond segmentatie voor thermische beelden

Door de kringen die optreden rond de personen in thermische beelden zijn de klassieke methoden voor voorgrond- achtergrondsegmentatie hier niet van toepassing. Deze methode steunt niet op vorige vorm modellen of bewegingsinformatie. Het algoritme werkt in drie stappen, de eerste is het bepalen van een interessante regio. De tweede is de contour detectie. Als laatste wordt de contour gesloten [9]. Een verdere uitleg over deze drie stappen, ziet u in de volgende paragrafen. De beste resultaten voor persoonsdetectie in thermische beelden gebeurt door gebruik te maken van trainbare, klasse specifieke object detectors [14].

Regio detectie We zoeken interessante regio's (ROI) die de personen en hun halo bevatten. Met een halo bedoelen we een kring die rondom een persoon verschijnt, omdat deze warmte uitstraalt. Hiervoor kunnen verschillende technieken gebruikt worden zoals het standaard gausisch model dat eerder besproken werd en een standaard intensiteit gebaseerde statistische achtergrond segmentatie techniek [10]. Om de ROI uit de beelden te halen wordt er gebruik gemaakt van een 5x5 dillatie, gevolgd door een algoritme om de aaneengesloten componenten te extraheren. Als de ROI te klein is wordt het automatisch weer verwijderd [9].

Contour detectie We bekijken elke ROI individueel, zodat we de persoon van zijn halo kunnen scheiden. Op elke regio wordt een Contour Saliency Map (CSM) toegepast. De waarde van elke pixel in de CSM bepaalt of de pixel behoort tot de persoon of de omgeving. Een CSM wordt bepaald door een vermenigvuldiging van de genormaliseerde voorgrond gradiënt magnitudes met de genormaliseerde voorgrond-achtergrond gradiënt verschillen magnitude in de ROI. Vervolgens gaan we de representatie van de CSM verkleinen, vervolgens wordt er een threshold toegepast om de beste pixels te selecteren. Daarna gaan we terug kijken naar de ROI's als het een zeer warm persoon was, gaat de detectie die we net gedaan hebben in de kring vallen, dan wordt het contour sterker getekend [9].

Contour sluiten De oplossing van de contour detectie is open. In de laatste stap wordt de contour aangevuld door per pixel te zoeken waar de dichtstbijzijnde contourpixels zijn [9].

2.4.3 Vormgebaseerde benadering

Deze techniek gaat personen herkennen aan de hand van hun lichaamsbouw. Hierdoor zouden we wel personen kunnen blijven detecteren, als ze weinig bewegen en bij achtergrond segmentatie in de achtergrond zouden verdwijnen. Verder wordt hier ook het ontstaan van geesten tegen gewerkt omdat deze niet de vorm van een persoon zullen hebben. Het grootste probleem voor deze techniek is dat patiënten in veel verschillende vormen kunnen voorkomen [23]. Dit ligt bijvoorbeeld aan slaaphouding en/of lichaamsbouw. Dit kan men oplossen door personen op te delen in de verschillende lichaamsdelen. Dit is op zichzelf al een zeer moeilijke taak. Het principe is zeer eenvoudig, als eerste wordt er een template in de vorm van een persoon ingeladen. Dan wordt de template over de afbeelding geschoven terwijl er gezocht wordt naar een overeenkomst [8].

2.5 Besluiten uit de literatuurstudie

In dit hoofdstuk hebben we veel verschillende technieken besproken. In deze sectie trekken we onze besluiten uit deze technieken.

Als eerste hebben we gezien dat er reeds bestaande systemen op de markt zijn, maar hier is geen informatie over te vinden. Verder zijn er nog niet veel systemen beschikbaar, dit wijst erop dat dit een zeer actueel onderzoek is.

Uit het bestuderen van de verschillende types van surveillance systemen, hebben we gezien dat wij een tweede of derde generatie systeem gaan ontwikkelen. Dit omdat wij de gedragsanalyse van patiënten gaan bepalen. We hebben ook gezien dat dit het belangrijkste maar ook meest tijdrovende deel is van het ontwerp van het surveillance systeem.

Uit de verschillende types van camera kunnen we besluiten dat er veel camera's zijn die in aanmerking kunnen komen voor ons systeem. We moeten een goede afweging maken tussen prijs en kwaliteit van het beeld om te bepalen welke camera het meest geschikt is voor onze toepassing. Omdat IR camera's goedkoop zijn, gaan we hiermee beginnen. Indien we valdetectie toevoegen aan ons systeem moeten we een tweede camera toevoegen. Dit kan eenzelfde type camera zijn, of we kunnen een ander type camera toevoegen zodat we meer informatie over de patiënt kunnen krijgen.

Als laatste hebben we algoritmes voor persoonsherkenning bekeken. De algoritmes op basis van kleurherkenning kunnen interessant zijn. Al kunnen we geen gebruik maken van methodes die gebaseerd zijn op huidskleur omdat wij onze beelden 's nachts maken. Maar we kunnen wel zien in welke kleur onze patiënten in het frame te zien zijn en op deze kleur gaan segmenteren. Voorgrond - achtergrondsegmentatie is niet interessant voor onze toepassing. Dit omdat onze patiënt stil ligt tijdens het slapen en dan mee in de achtergrond gaat verdwijnen. Dit is een goede techniek om een persoon of object te vinden die de hele tijd in beweging is. Uit het stuk over achtergrond bepaling van thermische beelden zijn er wel bepaalde stukken die mogelijk voor ons van toepassing zijn, zoals het omgaan met halo's. Vorm gebaseerde benadering is niet interessant omdat een persoon verschillende, soms vreemde, houdingen aanneemt in zijn slaap. Verder gaan er vaak delen van een persoon onzichtbaar zijn door een deken.

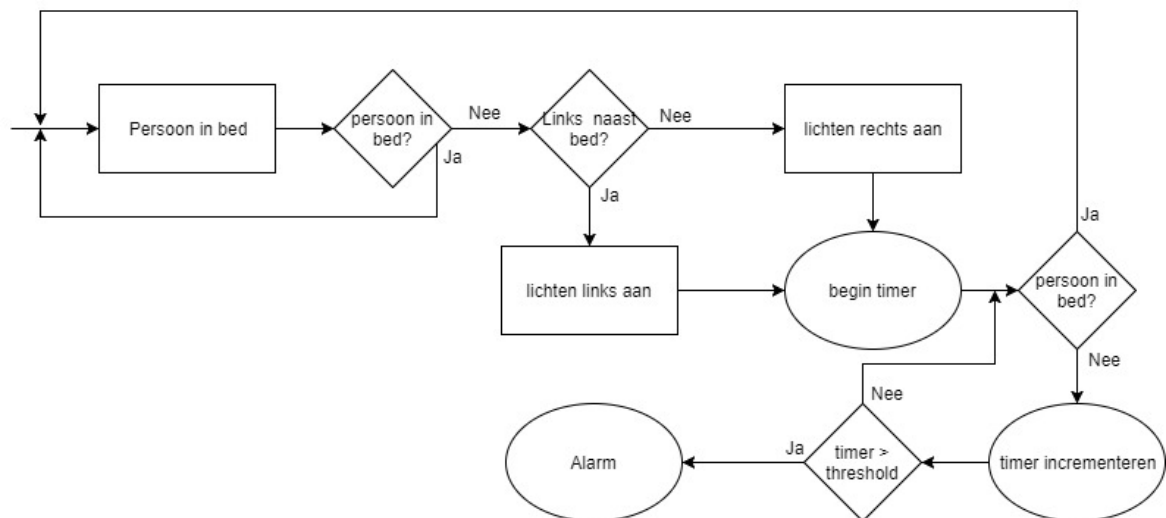
In hoofdstuk 3, gaan we de methode bespreken die we gebruikt hebben tijdens ons onderzoek.

Hoofdstuk 3

Methode en uitwerking

In dit hoofdstuk bespreken we de gebruikte werkwijzen, technieken en materialen. We beginnen met een algemeen blokschema van het te ontwerpen systeem is te vinden in sectie 3.1. Voor ons systeem hebben we ook een camera nodig. Welk type van camera we gebruiken en de plaatsing van deze camera bespreken we in sectie 3.2. Vervolgens gaan we aan de hand van dit blokschema de manieren bespreken waarmee we de gedragsanalyse geconstrueerd hebben, dit gebeurt in sectie 3.3. Na de bespreking van de gedragsanalyse, bespreken we de overige methoden van onze code en waarvoor ze gebruikt kunnen worden, dit gebeurt in sectie 3.4.

3.1 Werkingsschema



Figuur 3.1: Werkingsschema op hoog niveau

Dit onderzoek heeft als doel het ontwikkelen van een visiesysteem dat kan detecteren langs welke zijde een patiënt uit bed stapt en kan meten hoelang de patiënt wegblijft. Als mogelijke uitbereiding kunnen we ook val detectie toevoegen. In figuur 3.1 is het werkingsschema op hoog niveau te zien. We beginnen links bij persoon in bed. We gaan er vanuit dat er een persoon in het bed ligt. Vervolgens gaan we testen of er delen van het lichaam zich uit het bed bevinden, op welke manier

we dit doen, wordt later besproken in sectie 3.3.2. Indien er een persoon in het bed gevonden wordt, blijven we deze test herhalen. Is er geen persoon in het bed gedetecteerd, gaan we kijken naar welke zijde. Vanaf het moment dat de persoon het bed heeft verlaten, wordt er een timer gestart. Vervolgens gaan we terug kijken of er een persoon in het bed is. Als er een persoon is, dan gaan we terug naar de startpositie. Als er geen persoon in het bed is, wordt de waarde van de timer geïncrmenteerd. Daarna gaan we de waarde van de timer vergelijken met een threshold. Is de waarde van de teller groter dan de threshold, dan gaat er een alarm afgaan, deze persoon is te lang uit het bed. Indien de waarde van de timer kleiner is dan de threshold, gaan we terug kijken of er een persoon in het bed is. De waarde van de threshold kan in het programma aangepast worden als het toilet zich ver van het bed bevindt. Als de persoon terug in het beeld komt, stopt de teller en bevinden we ons weer in de eerste toestand.

3.2 IR camera

Voor de uitwerking van ons project maken we gebruik van een infrarood camera. Deze hebben we eerder al besproken in onze literatuurstudie 2.3.1. We gebruiken deze camera omdat deze ter beschikking werd gesteld door de school en dit ons ook een logische keuze leek. Dit omdat de persoon door de infraroodbeelden al minder snel herkenbaar zijn, op de beelden, wat een zeer belangrijk deel is van de specificaties. Verder is dit in verhouding met de andere besproken types van camera's in de literatuurstudie 2.3, een goedkope camera. Verder is in onze toepassing het niet nodig om een gedetailleerd beeld van de patiënt te verkrijgen. Hierdoor is de constructie van IP en PTZ camera's hier niet van toepassing. Andere types van camera zouden ook kunnen gebruikt worden. Maar dit onderzoek, is door bovenvermelde redenen, niet door ons gedaan. We bespreken de gebruikte camera, namelijk die Seek Thermal Compact in sectie 3.2.1. Voor de plaatsing van onze camera hebben we een aantal berekeningen gedaan. Dit om te zien waar en op welke hoogte onze camera het beste geplaatst wordt. Deze berekeningen zijn te vinden in sectie 3.2.2.

3.2.1 Seek Thermal Compact



Figuur 3.2: Image of Seek Thermal Compact

Dit is een zeer kleine camera die bedoeld is om te gebruiken met een Android smartphone of met een iPhone. Een afbeelding van de camera is terug te vinden in figuur 3.2 [13]. Wij hebben gebruik gemaakt van de versie die normaal op Android gestuurde systemen werkt. Om deze camera te

kunnen gebruiken op een computer, hebben we gebruik gemaakt van code die ons beschikbaar is gesteld via Eavise [29]. Via deze code kunnen we van de camera binnen halen en bekijken. Deze camera is gebruikt om alle testbeelden te maken tijdens de experimenten die besproken worden in hoofdstuk 4. Hieronder bespreken we de specificaties van de Seek Thermal Compact voor gebruik op Android toestellen [13]:

- μ USB Thermal Camera for Android devices
- Werkt met de meeste Android toestellen die werken met 4.3 of hoger. (Op de website, <http://www.thermal.com/products/compact/> is een lijst met compatibele modellen te vinden.
- 206 * 156 Thermal Sensor
- 12 μ Pixel Pitch
- Vanadium Oxide Microbolometer
- Chalocogeninde Lens
- 36° zicht
- Magnesium behuizing
- Lange golf infrarood 7.2 - 13 μ m
- spectrum van -40 tot 330 °c
- 9Hz
- Bevat een waterdichte cover
- Model: UW-AAA

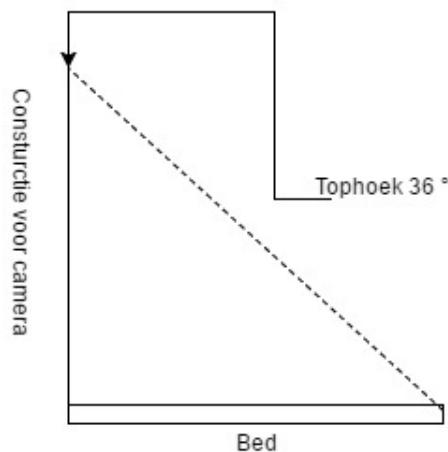
Een nadeel van deze camera is dat hij via een μ USB connectie met de computer verbonden moet worden. Hierdoor mag de camera niet te ver staan van de computer waarop het programma met de gedragsanalyse draait, of er moet onderzoek gedaan worden naar een mogelijkheid om de beelden draadloos door te sturen.

3.2.2 Plaatsing van de camera

Er zijn verschillende manieren, om de camera te plaatsen. Maar aangezien we het bed in de code moeten kunnen detecteren, willen we dat de positie van het bed hetzelfde blijft ten opzichte van de camera. Hierdoor moeten we een gebruik maken van een statische opstelling. Dit kunnen we doen door de camera via een constructie aan het bed te bevestigen. Een andere manier om een statische opstelling te verkrijgen, is door de camera te bevestigen aan het plafond of één van de muren van de kamer. We weten de openingshoek van de camera. Dankzij deze kunnen we de nodige hoogte van de camera berekenen. We nemen de lengte van het bed (2m) als de bepalende factor voor de hoogte. De camera mag altijd hoger bevestigd worden, zo gaat er meer van de omgeving waargenomen worden.

Camera recht boven hoofdeinde

De eerste berekening die we doen, is een situatie waar de camera recht boven het hoofdeinde geplaatst is. Deze kan zowel via aan constructie aan het bed gemonteerd worden als rechtstreeks aan het plafond. De camera hangt in het midden van de breedte van het bed. Op afbeelding 3.3 is een schets van de situatie te zien. De berekening van de hoogte is in dit geval heel eenvoudig. Het



Figuur 3.3: Schets van camera recht boven bed

gaat hier over een rechthoekige driehoek. We kunnen hier gebruik maken van de goniometrische regels voor rechthoekige driehoeken.

$$\tan(\alpha) = \frac{o}{x}$$

$$\tan(36^\circ) = \frac{2}{x} \Rightarrow x = \frac{2}{\tan(36^\circ)} = 2,75$$

met

- α is de openingshoek van de camera in uitgedrukt graden
- o is de lengte van het bed uitgedrukt in meter
- x is de gevraagde hoogte in meter

Uit de berekeningen blijkt dat de camera 2,75 meter boven het bed moet hangen. Dit is onmogelijk, omdat de gemiddelde kamer ongeveer een 2,5 meter hoog is. Hierdoor zullen we de camera moeten hangen op een andere locatie. Hieronder volgen de berekeningen voor de situatie waarbij de camera niet recht boven het hoofdeinde hangt.

Camera niet recht boven hoofdeinde

Nadat we uit de voorgaande berekening hebben kunnen besluiten dat het onmogelijk is het bed volledig op het frame te krijgen als we de camera recht boven het hoofdeinde hangen. Gaan we de nodige hoogte bepalen als de camera boven het bed komt te hangen, maar niet recht boven het hoofdeinde. figuur 3.4 geeft een schets weer van de situatie. De camera hangt in het midden van de breedte van het bed. De drukletters in de figuur 3.4 zijn de verschillende hoeken. Deze

worden ook in de berekening gebruikt. Door te rekenen met hoeken in driehoeken en het gebruik van de sinusregels voor willekeurige- en rechthoekige driehoeken, kunnen we de gezochte hoogte berekenen. Alle hoeken worden uitgedrukt in graden en de lengtes in meter. Voor het horizontale gedeelte van de constructie nemen we een paar keer een andere waarde, zo kan u zien hoe de hoogte evolueert. Bepaling waarden voor hoeken:

$$C = 36^\circ$$

$$B = E$$

$$A + B + c = 180^\circ \rightarrow A = 144^\circ - E$$

Vervolgens gaan we de sinusregel toepassen voor niet rechthoekige driehoeken:

$$\frac{2}{\sin(36^\circ)} = \frac{s}{\sin(144^\circ - E)}$$

Door gebruik van de cosinusregel in de rechthoekige driehoek verkrijgen we:

$$\cos(E) = \frac{x}{s}$$

Uit de twee voorgaande formules kunnen we hoek E berekenen als we de waarde voor x kennen:

$$\cos(E)\sin(144^\circ - E) = \frac{x \cdot \sin(36^\circ)}{2}$$

Nu we de waarde van de hoek E hebben bepaald, kunnen we door gebruik te maken van de tangens regel de hoogte berekenen:

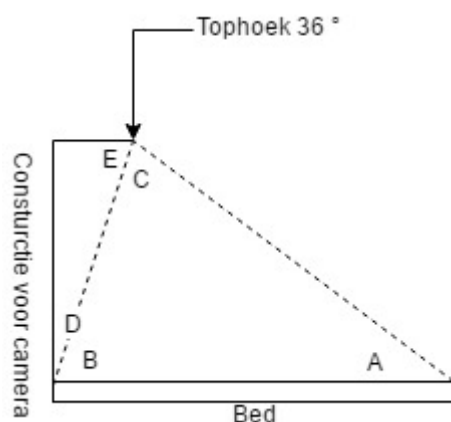
$$h = \tan(E) * x$$

met:

A, B, C, D, E de hoeken zoals weergegeven in figuur 3.4

s de schuine zijde in de driehoek (tussen B en C)

x de horizontale afstand tussen het hoofdeinde en de plaats van de camera



Figuur 3.4: Schets van camera niet recht boven bed

h de hoogte waarop de camera moet komen

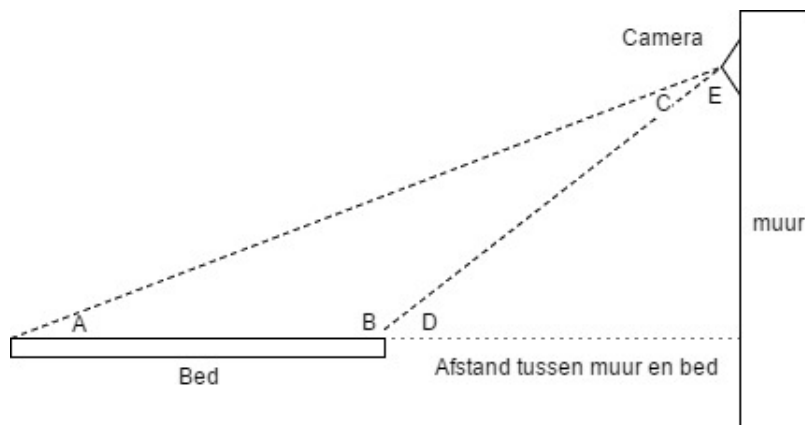
Een overzicht van de berekende waarden, is terug te vinden in tabel 3.1. Uit deze waarden kunnen we besluiten dat het onmogelijk is om deze camera in een normale ruimte aan het bed te bevestigen.

Tabel 3.1: Berekende waarden voor camera niet recht boven hoofdeinde

| x | E | h |
|------|----|------|
| 0.1 | 88 | 2.86 |
| 0.25 | 85 | 2.85 |
| 0.5 | 81 | 3.15 |
| 0.75 | 76 | 3.01 |
| 1 | 72 | 3.08 |

Camera niet bevestigd aan het bed

De voorgaande berekeningen hebben ons getoond dat we de camera niet aan het bed kunnen monteren. Nu gaan we de hoogte berekenen die we nodig hebben, als de camera achter het bed tegen de muur hangt, op een afstand van het voeteneinde. Een schets van de situatie is terug te vinden in de afbeelding 3.5. Net zoals bij de vorige berekening vindt u de namen van de hoeken



Figuur 3.5: Schets van camera niet bevestigd aan bed

terug in de afbeelding. Bij deze berekening nemen we aan dat de camera zelf geen dikte heeft. Dus de hoek E zit tussen de denkbeeldige lijn die de openingshoek van de camera weer geeft en de muur. Dit om de berekeningen iets eenvoudiger te houden. Over de hoeken hebben we volgende gegevens:

$$C = 36^\circ$$

$$A + B + C = 180^\circ$$

$$B + D = 180^\circ \Rightarrow A = D - 36^\circ$$

Uit de sinus regel voor niet rechthoekige driehoeken kunnen we volgende formule afleiden:

$$\frac{2}{\sin(36^\circ)} = \frac{s}{\sin(D - 36^\circ)}$$

Door de cosinusregel in rechthoekige driehoeken, geldt volgende formule

$$\cos(D) = \frac{x}{s}$$

Uit de twee voorgaande formules kunnen we de waarde van D berekenen als we de waarde voor x kennen:

$$\cos(D)\sin(D - 36^\circ) = \frac{x \cdot \sin(36^\circ)}{2}$$

Vervolgens gaan we de tangens regel voor rechthoekige driehoeken gebruiken om h te bepalen:

$$h = \tan(D) \cdot x$$

met:

A, B, C, D, E de hoeken zoals weergegeven in figuur 3.4

s de schuine zijde in de driehoek (tussen B en C)

x de afstand tussen het bed en de muur

h de hoogte waarop de camera moet komen

Een overzicht van de berekende waarden, worden weergegeven in tabel 3.2.

Tabel 3.2: Berekende waarden camera niet aan bed bevestigd

| horizontaal | D | hoogte |
|-------------|----|--------|
| 0.25 | 84 | 2.37 |
| 0.5 | 77 | 2.16 |

Aangezien een kamer ongeveer 2,5 meter hoog is, zien we dat het mogelijk is de camera op een hoogte te plaatsen waardoor het volledige bed zichtbaar is.

Besluiten bij de berekeningen

Uit de eerste twee berekeningen besluiten we dat het onmogelijk is om de gebruikte camera aan het bed te bevestigen en heel het bed te zien. Er zijn twee mogelijke oplossingen. We kunnen een andere camera nemen met een grotere openingshoek, of we kunnen de camera verder van het bed plaatsen zodat de benodigde hoogte kleiner gaat worden.

Uit de tweede berekening kunnen we besluiten dat het met de gebruikte camera wel mogelijk is om het volledige bed in het frame te krijgen. Als we de camera achter het bed aan de muur hangen.

3.3 Gebruikte technieken voor de gedragsanalyse

Deze sectie is opgebouwd uit twee verschillende delen. In het eerste deel bespreken we de dingen die we als eerste hebben gedaan in dit onderzoek, nog voor we begonnen zijn aan de effectieve gedragsanalyse. In het tweede deel bespreken we de gebruikte technieken aan de hand van de verschillende fasen in het werkingsschema, dat terug te vinden is in figuur 3.1 en dat we hebben besproken in sectie 3.1. Voor het maken van de gedragsanalyse en de code die we schrijven tijdens de voorbereidende technieken, wordt gebruik gemaakt van openCV, de gebruikte taal voor de code is c++. De code is terug te vinden op de cd die samen met deze tekst afgegeven is.

3.3.1 Voorbereidende technieken

Voor we begonnen zijn aan de gedragsanalyse, hebben we eerst bekeken of we de camera met de gekregen code, aan de praat kregen op de computer. Vervolgens hebben we nagekeken of we met deze camera een persoon in een bed kunnen waarnemen. Nadien hebben we twee klassen geschreven. De eerste is `SavelImage`, deze heeft één methode (`savelImage`) en heeft als doel het frame dat we ophalen met de gekregen code, weg te schrijven op het door ons gedefiniëerde pad. De tweede geschreven klasse is de klasse `GetImages`. Deze heeft eveneens één methode, namelijk `getImage`. Deze methode doet het frame op het door ons opgegeven pad weer ophalen. Figuur 3.6 geeft een aantal opgeslagen frames weer. Omdat we deze frames gebruiken om het persoonsdetectie op toe te passen, hebben we in deze beelden zo veel mogelijk variatie gestoken. Zo zijn er verschillende slaaphoudingen aangenomen en is er in sommige frames een deken aanwezig en in andere frames niet.



Figuur 3.6: Opgeslagen beelden Seek Thermal Compact

Opmerkingen openCV beschikt over standaardfunctie voor het wegschrijven en ophalen van afbeeldingen. Wij gebruiken deze, maar hebben de twee klassen geschreven omdat er voor het opslaan van de afbeelding een conversie moet gebeuren om de afbeeldingen om te zetten naar het juiste bestandstype. Anders zouden we de conversie elke keer apart moeten oproepen. Het bespaart elke keer een paar lijnen code en we verkleinen eveneens de kans dat iemand het vergeet of er een fout in getypt wordt.

Nadat deze fase afgelopen was, hebben we de Klassen `SavelImages` en `GetImages` samengevoegd tot één klasse, namelijk `SGImage`. De twee methodes worden vervolgens voor deze klasse geïmplementeerd. Dit omdat we nu nog maar één object moeten aanmaken om zowel afbeeldingen op te slaan als terug op te halen. Dit is efficiënter.

Het opslaan en ophalen van de afbeeldingen is ook nodig voor het maken van opeenvolgende testbeelden. Zo kunnen we overdag de ontwikkelde technieken laten lopen over dezelfde beelden en hier besluiten uit trekken. Verder maakt dit dat we ook niet elke keer als we iets willen testen we een persoon moeten zoeken die dan even in een bed wilt gaan liggen.

Wat ons ook is opgevallen tijdens het maken van de beelden, is dat de camera een klik geluidje

maakt elke keer dat er een frame wordt opgeslagen. Dit is heel vervelend om te slapen. Om het aangenamer te maken, gaan we maar om de paar seconden een frame opslaan. Hierdoor gaat het getik meer op de achtergrond verdwijnen en lijken op een tikkende klok.

Wat we ook zien op deze beelden, is dat de camera niet op een juiste plaats hangt, het bed is niet volledig te zien in het frame.

Verder moeten we met de persoonsdetectie ook rekening houden met het feit dat als er kleding gedragen wordt, of de persoon onder een deken ligt, deze minder goed zichtbaar is voor onze camera.

3.3.2 Ontwikkelen van de gedragsanalyse

Zoals eerder aangehaald, gaan we de gebruikte methodes van dit deel van het onderzoek overlopen aan de hand van het werkingsschema. Dit is terug te vinden in afbeelding 3.1. De gebruikte technieken zijn terug te vinden in de volgende paragrafen.

Persoon in bed



Figuur 3.7: Foto van het bed met warme objecten op de hoeken

Het eerste blokje in het werkingsschema is persoon in bed. We hebben een klasse Bed gemaakt. Een object van de klasse bed bestaat uit acht integers. Dit zijn de coördinaten van de hoekpunten van het bed. Dankzij deze kunnen we achteraf bepalen of een persoon in bed ligt, of aanstalten maakt om eruit te komen. Verder hebben we in dit stadium ook een mousecallback functie gemaakt deze wordt besproken in de eerstvolgende paragraaf, daarna volgt een bespreking van de Bed klasse

mouseCallback functie De functie gaat de punten waarop geklikt wordt, met de linkermuisknop opslaan in een vector. Door een klik met de rechtermuisknop, wordt het laatste punt in de vector verwijderd. Door op enter te klikken wordt de functie beëindigd.

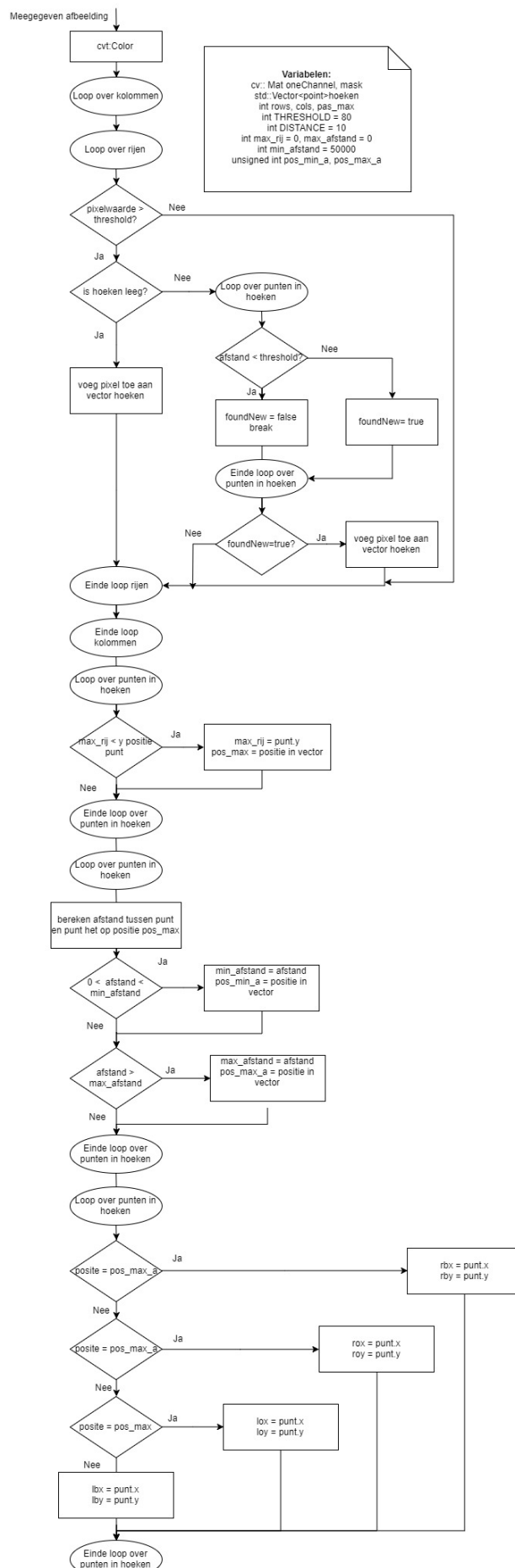
klasse Bed De klasse bezit drie constructoren. De eerste heeft geen argumenten. Indien deze opgeroepen wordt is het bed object een punt. De tweede constructor, heeft acht integers als argument. Deze methode heeft een paar nadelen. Je moet de hoekpunten van het bed op voorhand kennen. Deze zijn moeilijk om te berekenen. Om het voor de gebruikers van het systeem gemakkelijker te maken hebben we een andere constructor ontwikkeld. De derde constructor heeft

twee argumenten. Het eerste is een afbeelding en het tweede is een integer. De afbeelding is een frame waarop een leeg bed te zien is, waar op elke hoek van het bed een warm object geplaatst is. Een voorbeeld van zo een afbeelding is te zien in figuur 3.7. De camera gaat de warme objecten weergeven als lichte punten op een donkere achtergrond. Het tweede argument, de integer gaat bepalen op welke manier de coördinaten van de lichtere punten in de afbeelding toegekend worden aan de integers van het bed object. Als de integer 0 is, dan wordt er via de `mouseCallBack` functie, die eerder is besproken, gevraagd om de hoekpunten van het bed aan te klikken, beginnend bij het hoekpunt rechtsboven. Op elk punt moet dan één keer geklikt worden. De klik positie wordt toekend aan het bed object. Als de integer van het argument een andere waarde heeft als 0. Dan worden de locaties van de punten via blobdetectie automatisch uit de afbeelding opgehaald en toegekend. Deze constructor is veel gebruiksvriendelijker. Er moet wel eerst een afbeelding van het bed beschikbaar zijn. Deze kan eenvoudig gemaakt worden door het programma `saveBed` te gebruiken. `saveBed` gaat één frame ophalen en dit opslagen in het mapje van bed. Daar kan deze achteraf makkelijk opgehaald worden.

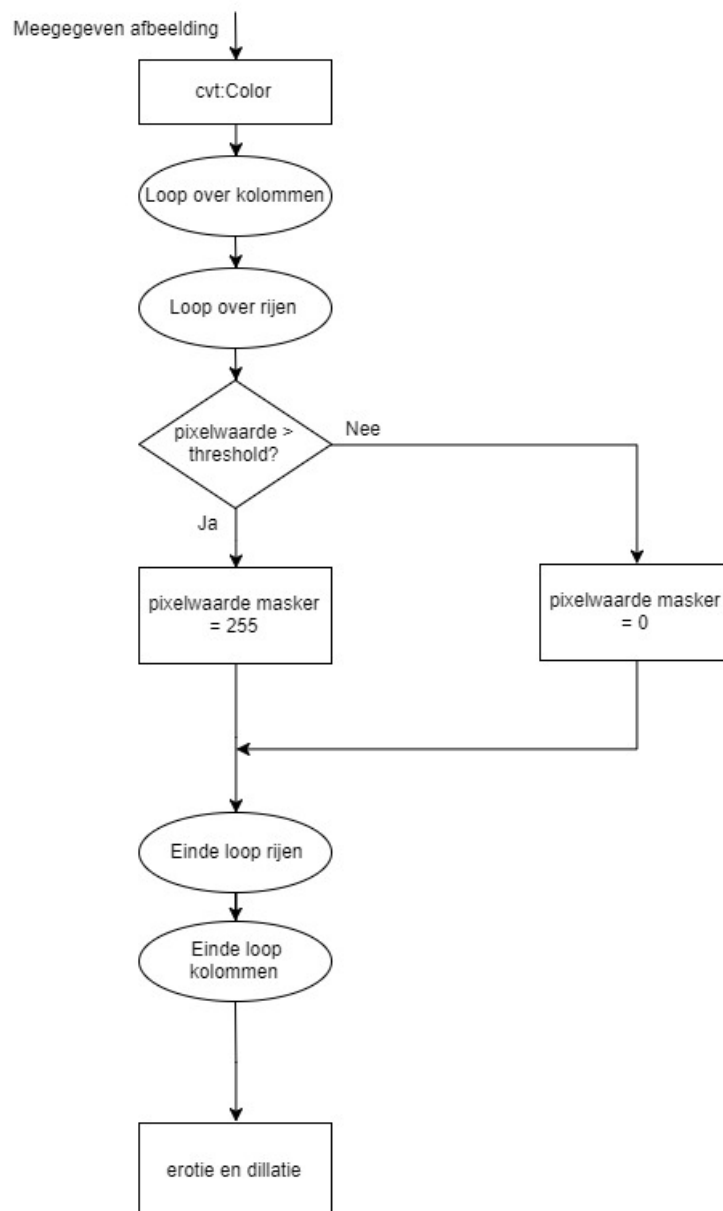
De klasse `bed` heeft een aantal andere methodes, de eerste is `setValues` deze heeft acht integers als argument en gaat deze toekennen aan de integers van het object. Een andere methode is `seValuesImg`, deze gaat via de `mouseCallBack` functie de waarden van de coördinaten toekennen aan het object. `setValuesAuto` gaat automatisch de hoekpunten bepalen aan de hand van de weergegeven afbeelding. de flowchart van deze methode is te zien in figuur 3.8. De methode werkt als volgt, er wordt een lege vector gecreërd. Vervolgens lopen we over alle pixels in de afbeelding, als de pixel licht is, wordt er gezien of we de waarde aan de vector moeten toevoegen. De pixelwaarde moet toegevoegd worden als de vector nog leeg is, of als er geen punt in de directe omgeving in de vector staat. Als we door alle pixels hebben gelopen, gaan we bepalen welk punt het meest onderaan staat. Dit is de linker onder hoek. Door gebruikt te maken van de onderlinge afstand van dit punt met de drie andere punten in de vector, worden de waarden aan de juiste hoeken toegekend. Een andere methode is `sidesOfBed`. Deze methode gaat aan de hand van de hoeken van het object, de zijanten van het bed bepalen. Dit zijn twee vergelijkingen van rechten. De coëfficiënten van deze vergelijkingen worden in een vector terug gegeven. De laatste methode van `Bed`, is de `headOfBed`. Deze gaat aan de hand van de hoekpunten de vergelijking van een lijn bepalen. Als de persoon in het bed ligt, gaat het hoofd boven de lijn liggen. De laatste twee methodes worden gebruikt tijdens de detectie van de persoon in het bed of het uit het bed stappen.

3.3.3 Persoon in bed?

Persoon in bed? is de tweede blok van het werkingsschema. Dit is het deel van de gedragsanalyse waar het meeste tijd in kruipt. Er zijn veel verschillende versies gebruikt om de detectie van een persoon te doen. Deze technieken zijn methodes van een nieuwe klasse. Deze klasse is de klasse `Detect`. De klasse `Detect` is opgebouwd uit een `Bed` object. De klasse `Detect` heeft twee constructoren. De eerste heeft acht integers als argument. Als deze opgeroepen wordt, wordt er een punt bed object gecreërd, nadien worden met de `setValues` methode van de klasse `bed` de coördinaten van de hoekpunten toegekend. De tweede constructor heeft een afbeelding als argument, na de creatie van het punt object, wordt door gebruik te maken van de methode `setValuesAuto` van de klasse `bed` de hoekpunten toegekend. In de volgende paragrafen bespreken we methodes die in deze klasse gemaakt zijn en gebruikt worden om de persoon in het bed te detecteren.

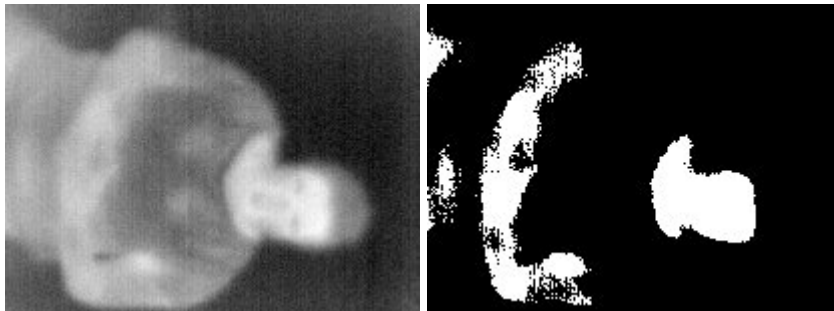


Figuur 3.8: Flowchart van de methode setValuesAuto

**Figuur 3.9:** Flowchart van de methode createMask

Eerste poging: createMask

Deze methode is gebaseerd op het thresholden van kleuren. Deze techniek is besproken in de literatuurstudie in sectie . Aangezien wij een grijswaarden afbeelding hebben, hebben we maar één threshold waarmee vergeleken moet worden. De flowchart van deze methode wordt weergegeven in figuur 3.9. De gekozen threshold is 180. Deze waarde is bekomen door middel van het principe van trial and error. Na afloop wordt erosie en dilatie toegepast. Hiermee gaan we de ruis verminderen. Figuur 3.10 geeft een frame en het verkregen masker weer. We zien dat het masker het gewenste resultaat vertoont.



Figuur 3.10: Voorbeeld van masker: (links) meegegeven afbeelding (rechts) verkregen masker

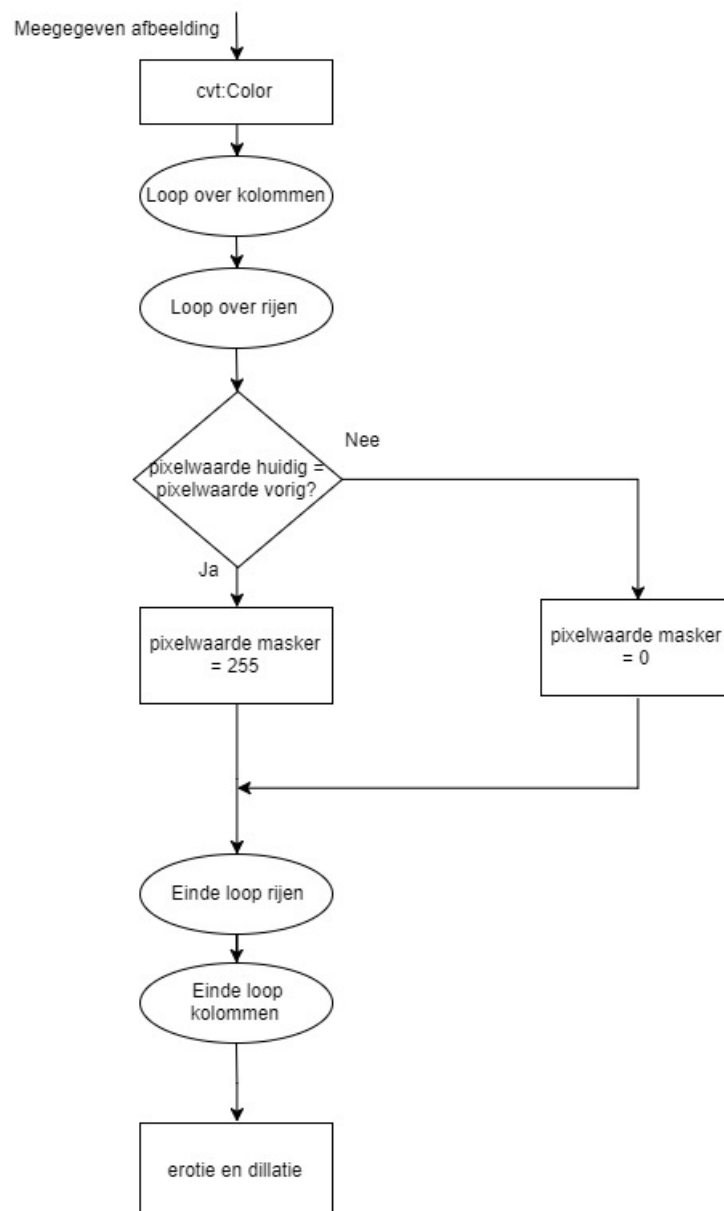


Figuur 3.11: Verkregen masker voor leeg opgewarmd bed

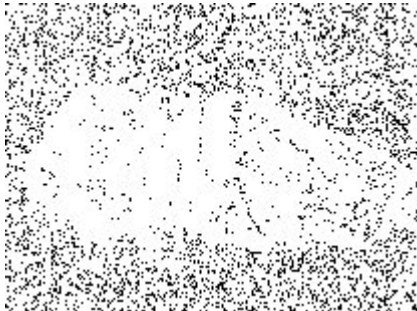
Opmerkingen Doordat de het bed opgewarmd wordt door de lichaamswarmte. Wordt de restwarmte, dit is de warmte die achterblijft in het bed nadat de persoon eruit gestapt is, ook gedetecteerd als een persoon. Een afbeelding hiervan wordt getoond in figuur 3.11. Verder worden er ook nog andere warme plekken gedetecteerd, die geen deel uitmaken van een persoon. Door deze valse persoon detecties hebben we een tweede manier van detectie gecreëerd. Deze wordt besproken in de volgende paragraaf.

Tweede poging: temporeel verschil

Om het probleem van de restwarmte op te lossen en om te controleren dat het alarm getriggerd wordt door bewegende delen. Hebben we het temporeel verschil geïntroduceert. Deze techniek is in de literatuurstudie besproken in sectie . De gedachte hier is eerst te gaan zien waar er beweging is, dit is waar de pixelwaarde voor twee opeenvolgende frames niet gelijk is. Dit wordt gedaan door

**Figuur 3.12:** Flowchart van de methode `tempDifference`

de methode `tempDifference` van de klasse `Detect`. Om de werking hiervan te verduidelijken is de flowchart toegevoegd. Deze is te zien in figuur 3.12. Het resultaat van deze methode noemen we vanaf nu het bewegingsmasker. Een voorbeeld van een bewegingsmasker is weergegeven in figuur 3.13. We zien dat er veel pixels wit zijn in zo een bewegingsmasker. Dit komt doordat alle

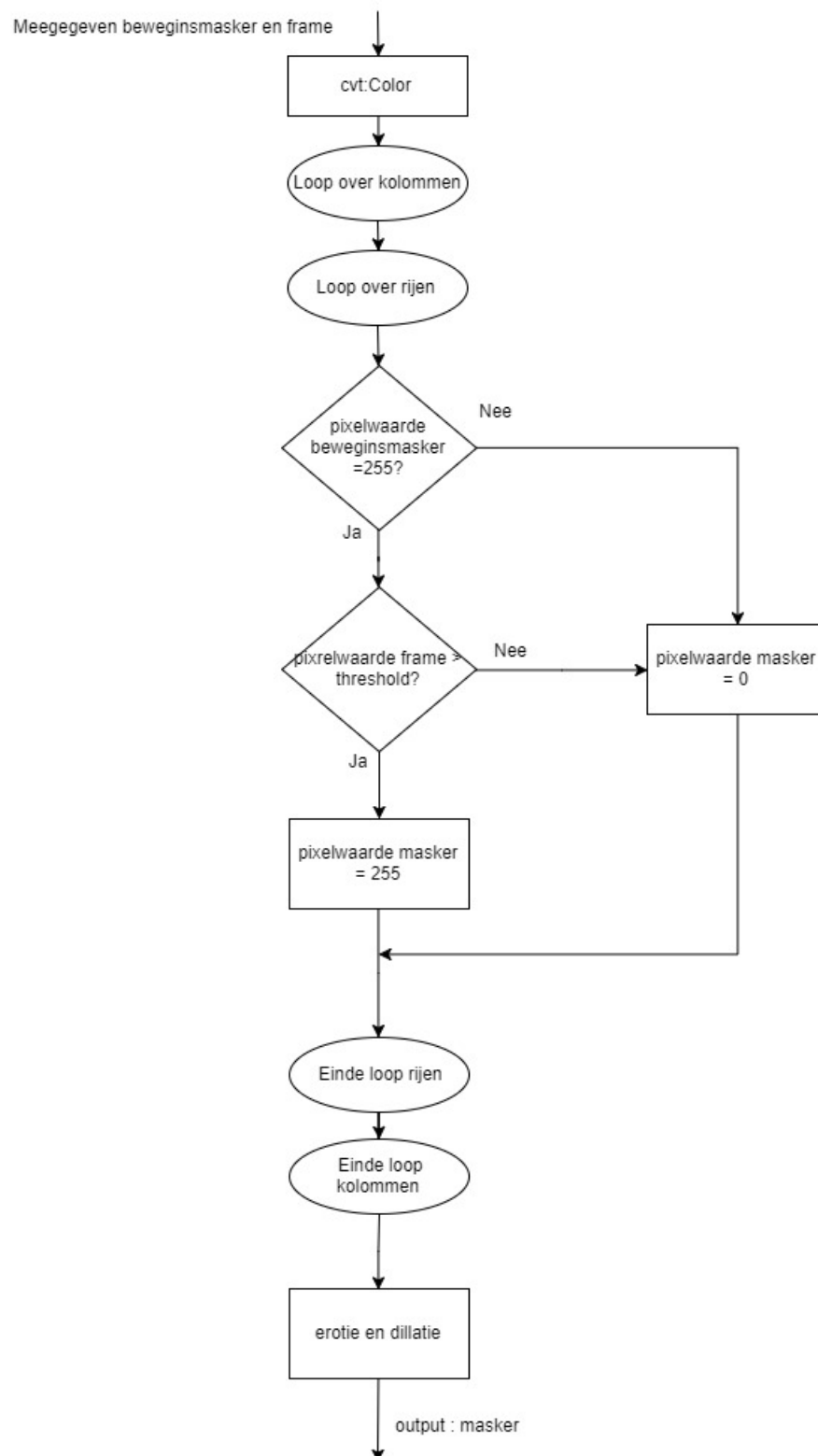


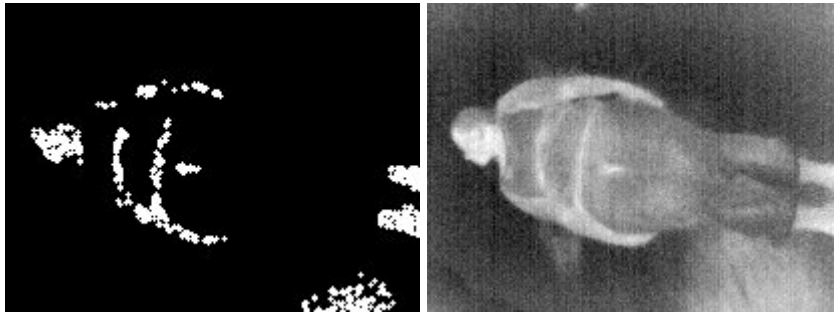
Figuur 3.13: Voorbeeld van een bewegingsmasker

pixels die kouder worden, zoals bijvoorbeeld een deel van het bed dat afkoelt nadat een persoon hier gelegen heeft, ook mee gedetecteerd worden. Verder is een verandering van pixelwaarde met 1 ook voldoende om gedetecteerd te worden. Omdat er toch veel niet interessante regio's gedetecteerd worden. Hebben we de methode `createMask` aangepast. Deze heeft nu twee argumenten het bewegingsmasker en het huidige frame. De functie gaat nu een masker creëren waarbij witte pixels voorkomen op de plaatsen waar de pixelwaarde in het huidige frame boven de threshold waarde is en de overeenkomende pixelwaarde van het bewegingsmasker ook wit is. De werking van deze methode wordt verduidelijkt aan de hand van een flowchart, weergegeven in figuur 3.14 geeft een masker bepaald door de `creatMaskNew` methode en het huidige frame weer. Figuur 3.15 Er worden nog steeds punten gedetecteerd die voor ons niet van toepassing zijn. Verder is er ook nog veel restwarmte die tot valse persoonsdetecties kan leiden.

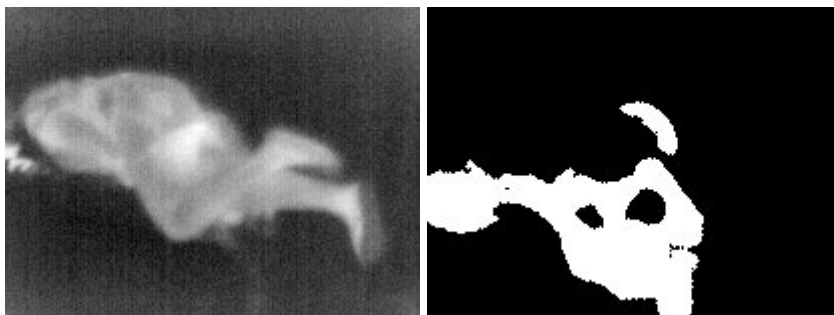
Derde poging: aangepast temporeel verschil

De twee voorgaande algoritmes hebben gefaald door het restwarmte probleem. Dit algoritme heeft als doel daar iets aan te doen. We hebben de methode `tempDiffNew` aangepast. Vanaf nu krijgt het bewegingsmasker de waarde 255 als de pixelwaarde van het nieuwe frame hoger is dan de pixelwaarde van het vorige frame. Als de persoon uit het bed stapt is dit bed nog warm, dit is dus restwarmte. Dit gaat in de komende tijd afkoelen tot kamertemperatuur. Daardoor zou dit niet meer gedetecteerd mogen worden in deze nieuwe methode. Verder wordt het algoritme op de zelfde manier opgebouwd als in de voorgaande poging. Figuur 3.16 toont een frame en een masker waarvoor deze methode correct werkt. Er zijn echter ook frames waarvoor deze methode niet werkt. Een voorbeeld hiervan is te zien in figuur 3.17. We zien dat de persoon hier zwart wordt weergegeven en de achtergrond wit. Dit is niet het gewenste resultaat. Dit probleem en het feit dat de restwarmte lang aansleept komt door de normalisatie van de pixelwaarden. Daarom hebben we dit algoritme ook toegepast op beelden voor de normalisatie optreedt. Buiten het weglaten van de normalisatie stap is het gehele algoritme behouden. Dit geeft geen goede resultaten omdat de camera een bereik van -40 tot 330 °C heeft. Het verschil tussen kamer temperatuur rond de 20 °C en de lichaamstemperatuur van 37 °C is niet groot genoeg om via threshold te detecteren. Op deze manier kunnen we geen correcte persoonsdetectie doen. Een voorbeeld van een masker gecreëerd zonder normalisatie is weergegeven in figuur 4.3

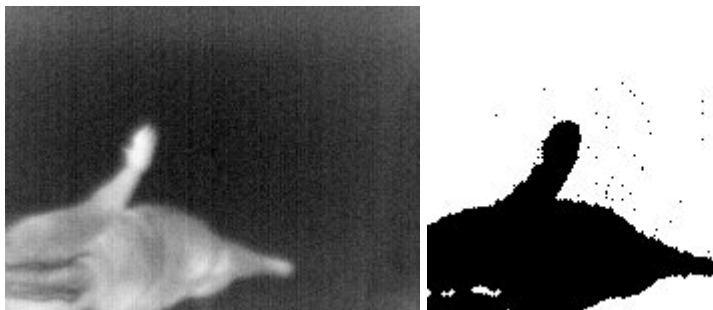
**Figuur 3.14:** Flowchart van de methode createMaskNew



Figuur 3.15: Frame en masker gemaakt met methode createMaskNew



Figuur 3.16: frame (links) en bijhorend bewegingsmasker (rechts) aangepast temporeel verschil



Figuur 3.17: frame (links) en bijhorend 'foutief' bewegingsmasker (rechts) aangepast temporeel verschil

Vierde poging: detectie van het hoofd

Als laatste algoritme voor de detectie van persoon in bed, werken we alleen met het hoofd. Dit is altijd zichtbaar, ook als een persoon onder een deken slaapt. Verder wordt dit ook niet bedekt door kleding. De methode die we hier gebruiken is `detectionHead`, er wordt een masker meegegeven. Het masker is een gewoon masker gecreëerd met `createMask`. We gaan zien of het hoofd zich bevindt het hoofdgedeelte van het bed. Het hoofdgedeelte van het bed wordt bepaald door de methode `headOfBed` van de klasse `Bed`. Praktisch doen we dit door een vorm van blobdetectie. We gaan over alle pixels van het masker lopen. Als een witte pixel gevonden wordt, gaan we indien de resultaten vector leeg is, de waarde in de vector opslaan. Indien de resultaten vector niet leeg is, gaan we zien of er al een punt uit de omgeving in de vector zit. Indien er al een punt uit de omgeving in de vector zit, worden de coördinaten van dit punt mee in de berekening het gemiddelde punt genomen. We vergelijken de gevonden punten met deze gemiddelde waarde bij het toevoegen in de vector. Indien er geen gemiddelde waarde in de omgeving van het punt ligt, wordt er een nieuw punt toegevoegd aan de vector. Nadat alle pixels doorlopen zijn, gaan we over de gemiddelde punten lopen in de vector. We vergelijken de positie van deze punten met de lijn die het hoofdeinde scheidt van de rest van het bed. Als er een hoofd in de juiste regio gevonden wordt, wordt er `true` terug gegeven, anders `false`.

3.3.4 links naast bed?

Indien we de persoonsdetectie doen aan de hand van het hoofd, gaan we de detectie van naast het bed doen als er `false` terug gegeven wordt na uitvoering van de methode `detectionHead`. We gaan dan het masker meegeven met de functie `checkBed` uit de klasse `Detect`.

Voor de andere algoritmes die besproken zijn in sectie ??, gaan we de verkregen maskers of bewegingsmaskers uit de vorige sectie, worden meegegeven met de methode `checkBed` uit de klasse `Detect`.

Het doel van deze methode is gaan kijken of er witte delen van het masker zich buiten het bed bevinden. Deze methode heeft één argument. Dit is het masker. We gaan over alle pixels lopen. Als de pixel van het masker wit is, gaan we kijken waar dit punt ligt ten opzichte van de zijkanen van het bed. De zijkanen worden bepaald door de methode `sidesOfBed` van de `Bed` klasse. Er wordt weergegeven langs welke zijde het witte deel zich bevindt. Als er geen witte delen buiten het bed zijn, dan wordt `None` terug gegeven.

Opmerkingen De methode `checkBed` kan eventueel ook geoptimaliseerd worden. In plaats van over alle pixels te lopen en te zien of het masker daar een waarde van 255 heeft, kan men ook enkel over de pixels lopen die geen deel uitmaken van het bed.

3.3.5 Timer

We hebben voor het bijhouden van hoelang de persoon uit het bed is, een nieuwe klasse `Alarm` gedefinieerd. Deze timer bevat een teller en een threshold. Deze klasse heeft één constructor, deze wordt opgeroepen als er voor de eerste keer geen persoon in bed gedetecteerd wordt. Deze zet de teller op 0. Als in het volgende frame geen persoon aanwezig is, dan wordt de methode `adOne` opgeroepen. Deze gaat de teller incrementeren. Vervolgens gaat hij de teller met de threshold vergelijken. Als de teller groter of gelijk is aan de threshold, dan wordt via de methode `timesUp` een alarm gegenereerd. Als de persoon terug in bed is of er is iemand van de verpleging langs

geweest, dan wordt via `endAlarm` het alarm gestopt. Er is nog één overige methode. Via `setThresh` kan je de `threshold` instellen indien de standaardwaarde niet goed is voor de desbetreffende kamer.

3.4 Overige methodes

3.4.1 `checkMovement`

Deze methode gaat 2 opeenvolgende frames vergelijken om te zien of het de moeite is om de volgende zwaardere berekeningen te doen. Indien er geen beweging is geweest, kan de persoon ook niet uit bed gestapt zijn, of eruit gevallen. Deze methode heeft als argument 2 opeenvolgende frames. Deze worden pixel per pixel vergeleken. Het aantal pixels waar er een verschillende waarde is wordt bijgehouden. Deze wordt vergeleken met een drempelwaarde. Indien het aantal verschillende pixels kleiner is dan de drempelwaarde, is er geen beweging geweest en wordt `false` terug gegeven. Indien er wel meer verschillende pixelwaarden zijn, heeft de persoon bewogen en wordt er `true` terug gegeven.

`erDil`

Deze methode heeft twee argumenten. De eerste is de afbeelding waarop de erosie en dilatie toegepast moet worden. Het tweede is de grootte van de matrix gebruikt voor het eroderen en dileren. Hoe groter de waarde, hoe meer details er van de afbeelding verdwijnen. Nadat de erosie en dilatie, wat men ook wel openen noemt, is toegepast wordt de resultaat afbeelding terug gegeven. Deze methode wordt tijdens het maken van een masker opgeroepen.

3.5 Besluit over de methode

Tijdens ons onderzoek maken we gebruik van een infraroodcamera. Er zijn echter nog andere types van camera mogelijk. Hiernaar hebben we geen onderzoek gedaan. Een mogelijk uitbereiding van dit onderzoek is om onze gedragsanalyse mits een paar aanpassingen, te gebruiken op beelden gemaakt met een andere type camera. Verder kunnen we besluiten dat er veel verschillende methodes van persoonsdetectie mogelijk zijn. Deze gaan echter niet allemaal tot een goed resultaat leiden. Verder zijn er een paar methodes waar er optimalisatie mogelijk is. Zoals bijvoorbeeld de `checkBed` methode. We hebben onze methode gebaseerd op het werkingsschema weergegeven in figuur 3.1. We kunnen dus besluiten dat het mogelijk is om een analyse volgens dit schema op te bouwen. Een andere mogelijke uitbereiding is het toevoegen van valdetectie. Om de code te gebruiken voor live beelden, moet de `cvtColor` stap in de methodes over geslagen worden. Tijdens het opslagen van de afbeeldingen en het vervolgens terug ophalen, gebeuren er een paar kleine aanpassingen. De data wordt op een andere manier weergegeven. `cvtColor` gaat deze aanpassingen ongedaan maken, zodat we weer met de beelden kunnen werken. We hebben een aantal experimenten met live beelden gedaan om te zien of onze analyse snel genoeg kan werken. In hoofdstuk 4 worden de verschillende experimenten besproken.

Hoofdstuk 4

Experimenten

Dit hoofdstuk behandelt de experimenten en hun resultaten. Het eerste experiment is een experiment dat gebuurd is om live beelden door gebruik te maken van de methodes `tempDifference` en `createMaskNew` van de klasse `Detect`. Een uitgebreidere beschrijving van dit experiment is terug te vinden in sectie 4.1. We hebben ook een experiment gedaan om na te gaan of het mogelijk is om de normalisatie achterwege te laten. Dit wordt beschreven in sectie 4.2. Het derde experiment dat besproken wordt, wordt er gebruik gemaakt van de detectie van het hoofd. Dit experiment vormt de inleiding naar hoofd gebaseerde detectie. Meer informatie hierover staat in sectie 4.3. Het laatste besproken experiment is de hoofd gebaseerde detectie. Deze wordt besproken in sectie 4.4

4.1 Experiment met live beelden

De totale bespreking van het experiment, wordt gedaan in drie verschillende delen. Als eerste verklaren we het doel in sectie 4.1.1. De uitwerking van het experiment wordt verduidelijkt in sectie 4.1.2. Nadat het experiment gedaan is, bekijken we de resultaten en trekken hieruit een aantal conclusies in sectie 4.1.3. Dit experiment wordt gedaan zonder hoofd detectie.

4.1.1 Doel van het experiment

Het doel van dit experiment is na te gaan of we de veranderingen in de beelden snel genoeg detecteren. We gaan dus testen of de detectie van langs welke zijde er aanstalten gemaakt wordt om uit bed te stappen, ook ogenblikkelijk gebeuren.

4.1.2 Uitwerking van het experiment

We hebben ons systeem aangesloten op de camera en live beelden verwerkt. Een schets van de proefopstelling ziet u in figuur 4.1. Tijdens dit experiment is er een persoon in het bed gaan liggen en heeft een paar keer gedaan als hij uit het bed wou stappen. Terwijl op het scherm in het oog werd gehouden of de computer de juiste detecties werden gemaakt. Om een duidelijk overzicht te hebben van welke methodes en klassen er gebruikt worden, is in figuur 4.2 te zien.

4.1.3 Besluiten na het experiment

Uit dit experiment kunnen we besluiten dat het systeem werkt. Vanaf er een arm of een been uit het bed wordt gestoken wordt er een melding gegenereerd met de juiste zijde. Het genereren van deze melding gebeurt vrijwel onmiddellijk. Kleding en deken vormen hier nog het grootste probleem. Als het lichaam bedekt is, wordt het door het masker niet gedetecteerd. Hierdoor zouden er evenementen gemist kunnen worden. Verder loopt het ook mis als er een persoon in het bed heeft gelegen en er uit stapt. De camera detecteert dan de restwarmte van het bed en gaat doen alsof er wel nog een persoon in ligt. Zoals reeds eerder is aangehaald, wordt hiervoor in een volgende experiment een oplossing gezocht.

4.2 Pixelwaarde zonder normalisatie

Het experiment over de pixelwaarde zonder normalisatie wordt in drie delen besproken. We bespreken het doel van het experiment in sectie 4.2.1. We bespreken de specifieke uitwerking van het experiment en de bekomen resultaten in sectie 4.2.2. We sluiten af met het bespreken van de resultaten en het nemen van besluiten uit deze resultaten in sectie 4.2.3, op het einde van het einde van het experiment.

4.2.1 Doel van het experiment

Het doel van het experiment is het na gaan of het mogelijk is om de normalisatie stap achterwege te laten. Voor we dit onderzoeken, moeten we natuurlijk ook weten wat juist het doel is van deze normalisatie stap. Tijdens de normalisatie worden de pixelwaarden verschaald. De laagste waarde, krijgt de laagste meegegeven waarde met de functie, in ons geval is dit 0. De grootste waarde van het frame krijgt de andere meegegeven waarde, bij ons 65535 omdat we in een 16 bit systeem werken. De tussenliggende waarden worden vervolgens hiertussen geschaald. Hierdoor krijgen we soms de indruk dat pixels warmer zijn geworden terwijl dit niet is. Dit geeft eveneens problemen met een leeg bed, omdat de temperatuur dan overal ongeveer gelijk is, worden kleine verschillen groot weergegeven worden.

4.2.2 Uitwerking van het experiment

We hebben dit experiment in drie delen gedaan. Dit met een paar kleine verschillen in de uitvoering. De werking van het eerste experiment is heel eenvoudig. We halen een frame binnen, we doen deze keer de normalisatie stap niet. We lopen één voor één door alle pixels. We zoeken de minimale, maximale pixelwaarde en drukken deze af. We berekenen eveneens de gemiddelde pixelwaarde en drukken deze eveneens op het scherm af.

We doen dit eerst een paar keer voor een leeg bed, zodat we de waarden voor de achtergrond weten. Vervolgens doen we dit een paar keer voor een frame terwijl een persoon in het bed ligt. De persoon gebruikt geen deken. Bijgevolg zou voor de camera het volledige lichaam zichtbaar moeten zijn. Vervolgens gaan we de waarden bekijken en hieruit besluiten trekken.

Daarna hebben we hetzelfde experiment nog eens uitgevoerd. Omdat we een paar gegevens ontbreken om correcte besluiten te kunnen trekken. We gaan nu ook berekenen hoeveel keer de maximale en minimale waarde voorkomt. Deze twee waarden worden eveneens op het scherm weergegeven. Een ander verschil is dat we tijdens de tweede poging ook de waarden bepalen van

een frame waar een persoon onder een laken ligt. 's Nachts liggen de patiënten ook onder een laken en ons programma moet in dergelijke omgeving ook kunnen werken.

Tijdens het derde experiment, gaan we de waarden niet meer weergeven op het scherm, maar gaan we proberen om een masker te creëren. Dit willen we doen door gebruik te maken, van de functie `createMask` van de klasse `Detect`. Ook hier doen we dit voor drie verschillende toestanden, namelijk een leeg bed, een bed met een persoon en een bed met persoon met laken. Aan de hand van de verkregen maskers kunnen we dan besluiten of we met deze methode verder willen werken.

4.2.3 Bespreking resultaten en besluiten na het experiment

We hebben voor de duidelijkheid dit deel van de bespreking opgesplitst in drie delen. Dit om het verschil in resultaten van de verschillende experimenten duidelijk te maken.

Eerste poging

Tabel 4.1: Gemeten niet genormaliseerde waarden leeg bed, eerste poging

| maximale pixelwaarde | minimale pixelwaarde | gemiddelde pixelwaarde |
|----------------------|----------------------|------------------------|
| 204 | 30 | 79 |
| 189 | 17 | 75 |
| 197 | 10 | 71 |
| 255 | 0 | 73 |
| 255 | 0 | 74 |
| 254 | 6 | 69 |
| 204 | 37 | 82 |
| 255 | 0 | 64 |
| 190 | 5 | 71 |
| 255 | 0 | 59 |
| 198 | 48 | 84 |
| 206 | 40 | 82 |
| 173 | 3 | 64 |
| 190 | 1 | 66 |

Een overzicht van de maximale, minimale en gemiddelde waarde voor een leeg bed, zijn terug te vinden in tabel 4.1. Vervolgens hebben we de gemiddelde waarde berekend van de gemiddelde waarden, voor het leeg bed bedraagt deze 72,42. We hebben deze berekend omdat dit voor beiden gemakkelijk te berekenen en te vergelijken is.

Vervolgens staat een overzicht van de maximale, minimale en gemiddelde waarde voor een bed met daarin een persoon zonder deken in tabel 4.2. Hier bedraagt de gemiddelde waarde van de gemiddelde waarden 83,79.

Als we de waarden bekijken, zien we dat als een persoon aanwezig is, de maximale en minimale waarde altijd 255 en 0 is. Al zou normaal de aanwezigheid van een persoon de minimale waarde niet mogen beïnvloeden. Hieruit kunnen we afleiden dat er een vorm van normalisatie aanwezig is in de camera. Verder zien we dat de gemiddelde waarde ook hoger ligt. Maar omdat er bij het lege bed een aantal uitschieters naar boven zijn, en bij het bed met persoon een aantal uitschieters

naar beneden zijn, kunnen we hier niet alleen naar kijken. Daarom zijn wij in het tweede deel van dit experiment gaan tellen hoeveel keer de verschillende waarde voorkomt.

Tabel 4.2: Gemeten niet genormaliseerde waarden bed met persoon, eerste poging

| maximale pixelwaarde | minimale pixelwaarde | gemiddelde pixelwaarde |
|----------------------|----------------------|------------------------|
| 255 | 0 | 82 |
| 255 | 0 | 78 |
| 255 | 0 | 80 |
| 255 | 0 | 84 |
| 255 | 0 | 91 |
| 255 | 0 | 88 |
| 255 | 0 | 89 |
| 255 | 0 | 92 |
| 255 | 0 | 89 |
| 255 | 0 | 89 |
| 255 | 0 | 77 |
| 255 | 0 | 83 |
| 255 | 0 | 75 |
| 255 | 0 | 76 |

Tweede poging

Tabel 4.3: Gemeten niet genormaliseerde waarden leeg bed, tweede experiment

| maximale waarde | minimale waarde | gemiddelde waarde | aantal maximaal | aantal minimaal |
|-----------------|-----------------|-------------------|-----------------|-----------------|
| 202 | 21 | 78 | 1 | 1 |
| 188 | 18 | 74 | 1 | 1 |
| 196 | 39 | 83 | 1 | 1 |
| 208 | 39 | 83 | 1 | 1 |
| 193 | 25 | 77 | 1 | 1 |
| 207 | 36 | 84 | 1 | 1 |
| 255 | 0 | 61 | 6 | 8 |
| 211 | 39 | 82 | 1 | 3 |
| 182 | 9 | 73 | 1 | 1 |
| 202 | 28 | 78 | 1 | 1 |
| 185 | 10 | 72 | 1 | 1 |
| 197 | 15 | 75 | 1 | 1 |
| 174 | 3 | 67 | 1 | 1 |
| 200 | 30 | 81 | 1 | 1 |
| 255 | 0 | 60 | 2 | 11 |

De gemeten pixelwaarden voor het lege bed ziet staan in tabel 4.3. Ook hier hebben we de gemiddelde gemiddelde pixelwaarde berekend. Deze bedraagt nu 75,2. Hoe kan het dat deze ineens warmer is? Volgens ons komt dit door een aantal factoren. Het tweede experiment gebeurt later op de dag, daardoor is de kamer door de zon meer opgewarmd. Verder staan in dezelfde kamer ook

twee computers aan. Deze zorgen ook voor meer omgevingswarmte. We zien wel weinig verschil in de extremen.

Tabel 4.4: Gemeten niet genormaliseerde waarden bij een bed met persoon zonder laken, tweede experiment

| maximale waarde | minimale waarde | gemiddelde waarde | aantal maximaal | aantal minimaal |
|-----------------|-----------------|-------------------|-----------------|-----------------|
| 255 | 0 | 100 | 12 | 19 |
| 255 | 0 | 106 | 11 | 9 |
| 255 | 0 | 105 | 26 | 29 |
| 255 | 0 | 96 | 7 | 6 |
| 255 | 0 | 107 | 73 | 51 |
| 255 | 0 | 104 | 40 | 46 |
| 255 | 0 | 103 | 42 | 46 |
| 255 | 0 | 102 | 45 | 44 |
| 255 | 0 | 89 | 22 | 21 |
| 255 | 0 | 104 | 11 | 20 |
| 255 | 0 | 96 | 9 | 7 |
| 255 | 0 | 96 | 8 | 15 |
| 255 | 0 | 86 | 7 | 13 |
| 255 | 0 | 94 | 11 | 13 |
| 255 | 0 | 94 | 4 | 6 |

De waarden voor het bed met een persoon zonder laken ziet u in tabel 4.4, hier bedraagt de gemiddelde gemiddelde pixelwaarde 86,33. Deze waarde is ongeveer evenveel gestegen als de gemiddelde waarde bij een leeg bed. Dit komt natuurlijk door dezelfde factoren. Hier valt natuurlijk direct op dat de maxima en minima weer elke keer gelijk zijn. De extremen zijn extremer geworden. Verder zien we ook dat de maximale en minimale waarden beduidend meer voorkomen. Dit duidt om normalisatie.

Als laatste hebben we ook de resultaten van de frames van een persoon met laken samengevoegd in tabel 4.5. Uit deze waarden bepaalden we de gemiddelde gemiddelde pixelwaarde, deze bedraagt 98,8. Deze is beduidend hoger dan de pixelwaarde voor een persoon zonder laken. Een mogelijke verklaring hiervoor is dat het laken de lichaamswarmte meer verspreid. Daardoor dat ook het aantal pixels met een maximale waarde stijgt.

Uit dit experiment kunnen we natuurlijk geen definitief besluit trekken, hiervoor zijn 15 steekproeven niet voldoende. Het geeft echter aan dat we in deze richting mogelijk wel verder kunnen. Maar hiervoor moeten eerst andere onderzoeken en experimenten uitgevoerd worden. Dit hebben we dan ook gedaan in het derde experiment.

Derde poging

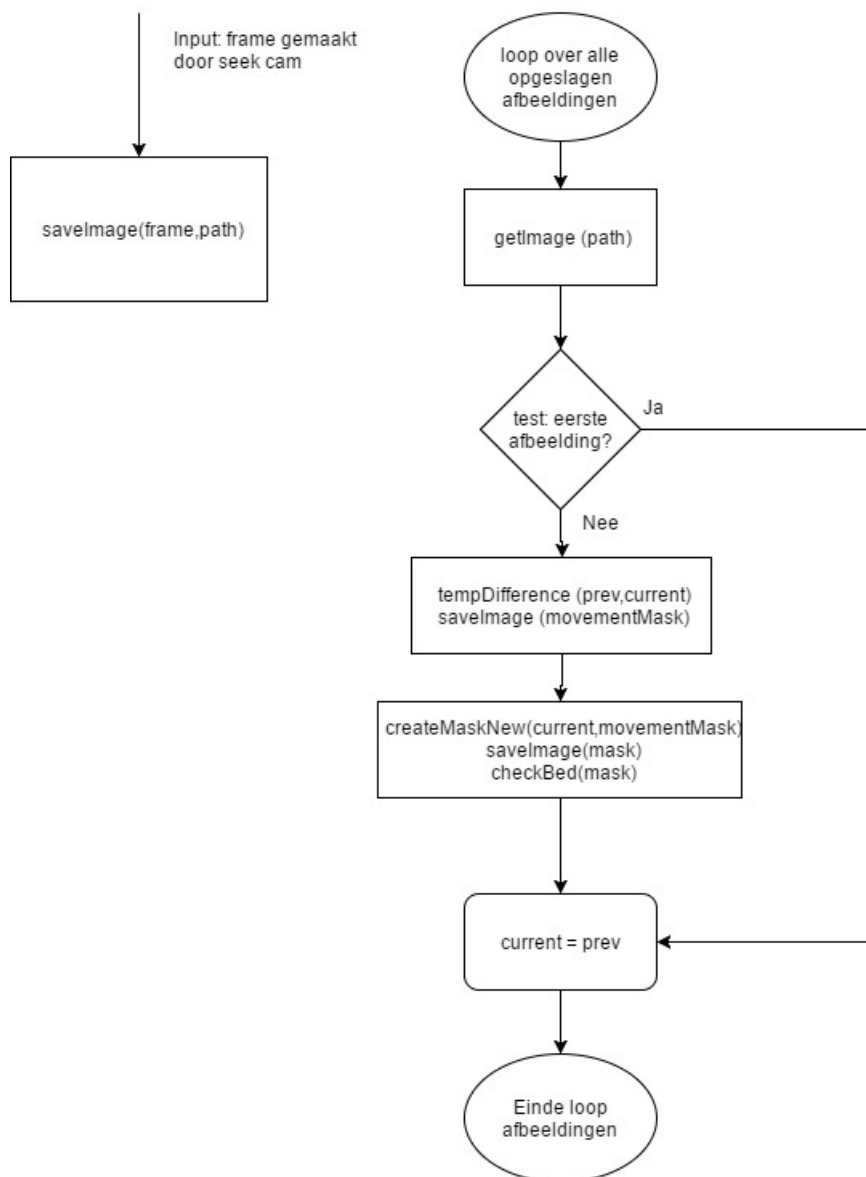
We hebben geprobeerd om zo maskers te maken om dit verder te onderzoeken. Een voorbeeld van zo een masker is getoond in figuur 4.3. Links ziet u het frame, om dit op te slaan zijn de waarde wel genormaliseerd, anders krijg je een volledig grijs frame te zien. Links ziet u het berekende masker van het frame. Zoals te zien is in de figuur, geeft dit niet de gewenste resultaten. Omdat er een veel groter deel dan enkel de persoon wit wordt. Dit werd niet beter door het aan-



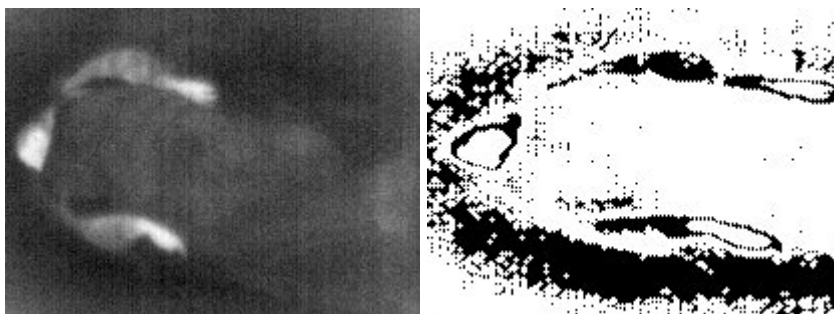
Figuur 4.1: Schets experiment met live beelden

Tabel 4.5: Gemeten niet genormaliseerde waarden bij een bed met persoon met laken, tweede experiment

| maximale waarde | minimale waarde | gemiddelde waarde | aantal maximaal | aantal minimaal |
|-----------------|-----------------|-------------------|-----------------|-----------------|
| 255 | 0 | 99 | 19 | 28 |
| 255 | 0 | 89 | 14 | 20 |
| 255 | 0 | 93 | 18 | 19 |
| 255 | 0 | 95 | 20 | 23 |
| 255 | 0 | 88 | 17 | 22 |
| 255 | 0 | 88 | 22 | 22 |
| 255 | 0 | 91 | 21 | 22 |
| 255 | 0 | 86 | 19 | 17 |
| 255 | 0 | 85 | 19 | 23 |
| 255 | 0 | 83 | 6 | 10 |
| 255 | 0 | 72 | 35 | 37 |
| 255 | 0 | 78 | 15 | 23 |
| 255 | 0 | 77 | 16 | 9 |
| 255 | 0 | 86 | 27 | 16 |
| 255 | 0 | 85 | 8 | 6 |



Figuur 4.2: Flowchart detectie uit bed stappen met live beelden



Figuur 4.3: frame (links) en bijhorend masker (rechts) bepaald door niet genormaliseerde waarden

passen van de threshold. Daarom hebben we moeten besluiten dat het niet mogelijk is om dit te gebruiken. Doordat de camera 3.2 een warmtebereik heeft van -40 tot 330 graden, is het verschil tussen kamertemperatuur en de lichaamstemperatuur niet groot genoeg. Hierdoor gebeurden valse detecties. De normalisatie stap is dus nodig om dit verschil in temperatuur te vergroten.

4.3 Hoofd gebaseerde detectie inleidend experiment

Het idee achter dit experiment komt voort uit de veronderstelling dat het hoofd altijd zichtbaar is. Verder zagen we uit de voorgaande resultaten dat we het interessant kan zijn om een keer opnieuw te beginnen en deze keer vanuit een ander oogpunt te werken. Zodat we zo weinig mogelijk last ondervinden van de restwarmte. Dit experiment wordt ook in drie verschillende delen besproken. Het doel van het experiment wordt verduidelijkt in sectie 4.3.1. De uitwerking van het experiment wordt besproken in sectie 4.3.2. Om het experiment af te sluiten bespreken we de resultaten en vormen we een besluit in sectie 4.3.3.

4.3.1 Doel van het experiment

Tijdens dit experiment willen we nagaan of het mogelijk is door om aan de hand van de locatie van het hoofd te bepalen of een persoon al dan niet uit bed probeert te stappen.

4.3.2 Uitwerking van het experiment

Dit experiment bestaat uit twee delen, eerst hebben we een sequentie van beelden opgenomen waarop een persoon te zien is die een paar keer in en uit een bed stapt. Dit een paar keer zonder een deken en een paar keer met een deken. Nadat deze beelden gemaakt en opgeslagen zijn, gaan we ze één voor één terug ophalen en met behulp van de `mouseCallBack`, die eerst verklaard werd in sectie 3.3.2, gaan we ongeveer op het midden van het zichtbare deel van het hoofd klikken. Hierdoor vinden we de pixelwaarden voor het midden van het hoofd. Deze waarden worden genoteerd en samengevoegd in een grafiek, later worden hier besluiten uit genomen.

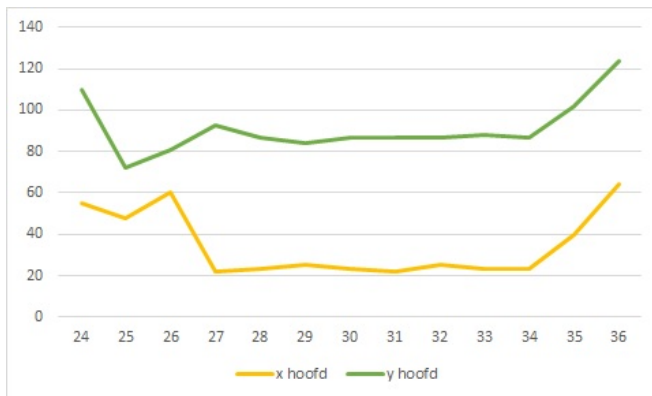
4.3.3 Resultaten en besluiten van het experiment

Zoals eerder vermeld hebben we de waarden van de coördinaten in een grafiek weergegeven, de grafiek voor de eerste keer in en uit een bed stappen, staat in figuur 4.4. Hier valt direct op dat tussen frame 24 en 27 de persoon bezig is met gaan liggen. Hier zijn de waarden voor de coördinaten groot, tussen frame 28 en 34 ligt de persoon in bed en blijven de waarden ongeveer gelijk. Later tussen frame 34 en 36 staat de persoon op en worden de waarden weer geleidelijk aan weer groter. We zullen moeten zien of deze resultaten ook bij de andere drie metingen terug komen.

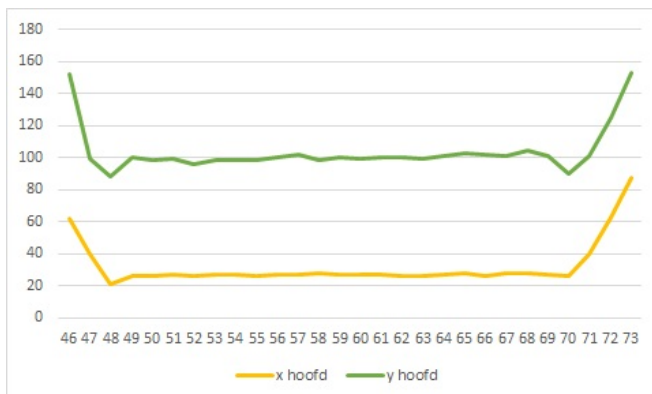
De grafiek van de tweede reeks frames wordt weergegeven in figuur 4.5. Uit deze waarden kunnen we dezelfde besluiten trekken.

De derde en vierde set zijn frames van een persoon met laken, al heeft dat voor deze test geen invloed. De resultaten van de derde set zijn samengebracht in figuur 4.6.

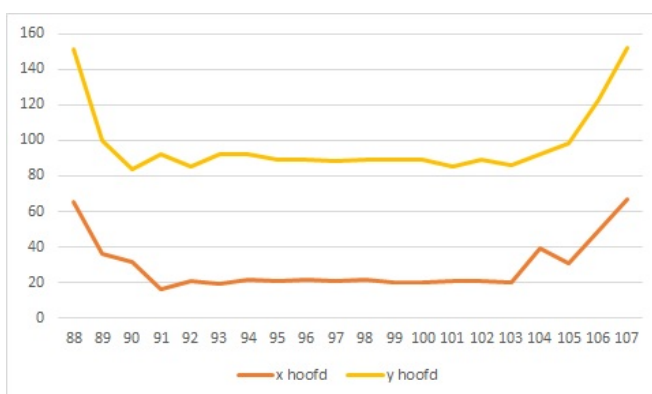
De grafiek van de vierde sequentie van frames zijn terug te vinden in figuur 4.7. Hierin zijn in het begin een paar schommelingen in de waarden. Dit komt doordat de persoon een paar keer terug recht is gekomen om het deken goed te leggen. Verder zien we dat de besluiten die we namen na



Figuur 4.4: Grafiek met de coördinaten van het hoofd (eerste set frames), met op de horizontale as de framenummers, de verticale as de pixelwaarden



Figuur 4.5: Grafiek met de coördinaten van het hoofd (tweede set frames), met op de horizontale as de framenummers, de verticale as de pixelwaarden



Figuur 4.6: Grafiek met de coördinaten van het hoofd (derde set frames), met op de horizontale as de framenummers, de verticale as de pixelwaarden

de eerste set, hier ook nog geldig zijn.

In het algemeen kunnen we dit experiment afsluiten met het besluit dat het zinvol is dit verder te bekijken. Zoals bijvoorbeeld te bepalen hoe de coördinaten zich verhouden met de hoeken van het bed. Verder moeten we ook nog bekijken of het mogelijk is om met een soort van blobdetectie het midden van het hoofd te berekenen en of we dan geen valse detecties krijgen door de restwarmte.

4.4 Hoofd gebaseerde detectie

Net zoals de voorgaande experimenten wordt dit experiment in drie delen besproken. Het doel van het experiment wordt verduidelijkt in sectie ???. Om te verduidelijken wat er in het experiment gebeurt, wordt de uitwerking verklaard in sectie 4.4.2. De resultaten van dit experiment en de bijhorende besluiten worden gevonden in sectie 4.4.3.

4.4.1 Doel van het experiment

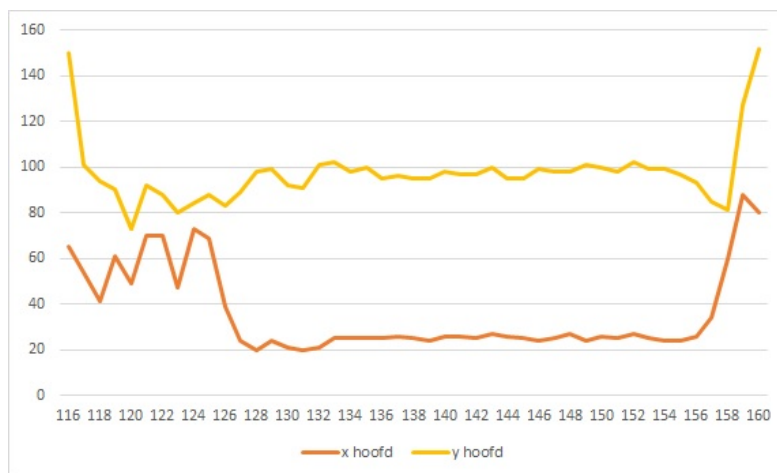
Tijdens dit experiment willen we op basis van de locatie van het hoofd detecteren of er een persoon in het bed ligt. Indien de persoon niet meer in bed ligt, willen we detecteren langs welke zijde deze persoon uit het bed stapt. Vanaf de persoon niet meer in bed ligt, moet er bijhouden worden of het alarm moet afgaan.

4.4.2 Uitwerking van het experiment

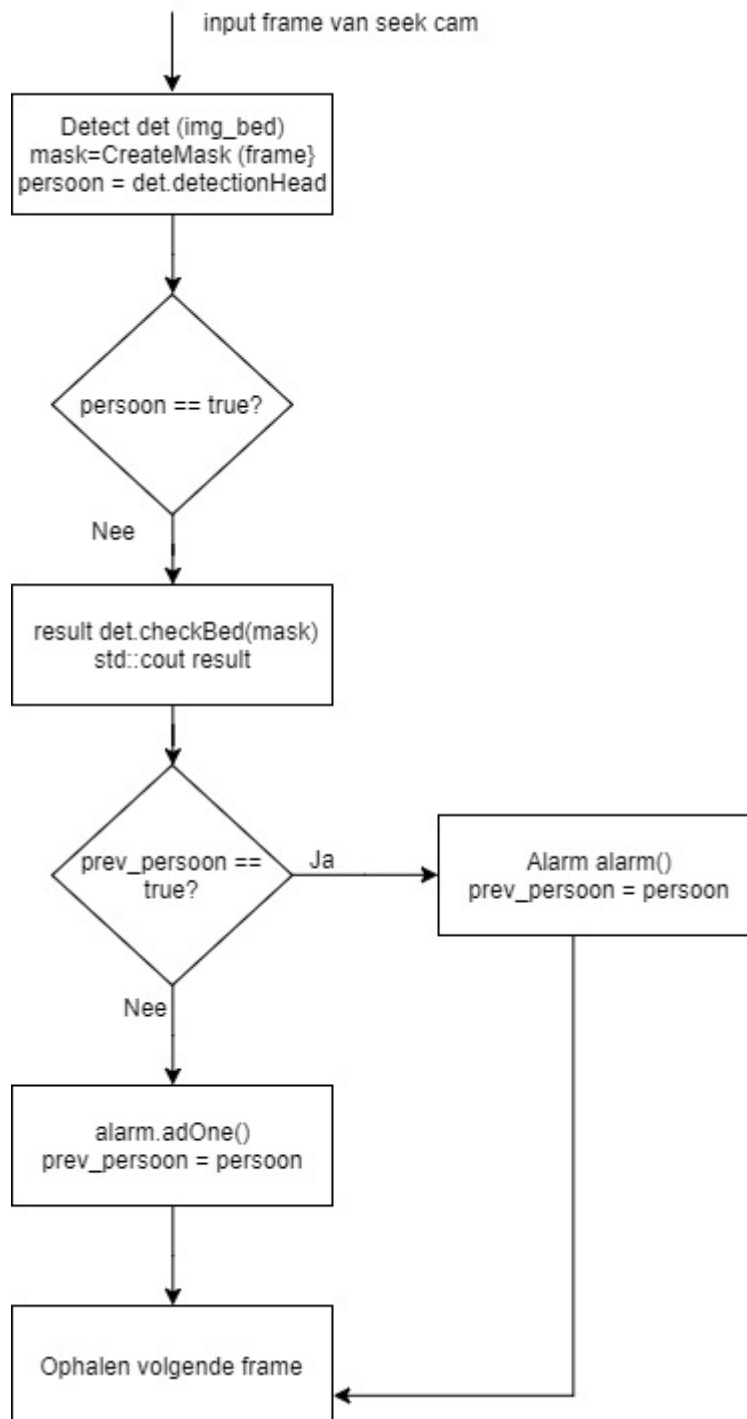
We verklaren de werking van dit experiment aan de hand van een flowchart, deze is te zien in figuur 4.8. We beginnen met het binnenhalen van een frame. We bepalen het masker van het frame. Vervolgens roepen we `detectionHead` op. Als er een persoon in het bed ligt, gaan we de huidige persoonswaarde toekennen aan de vorige. We gaan vervolgens het volgende frame ophalen. Indien er geen frame gevonden wordt, gaan we `checkBed` oproepen om na te kijken langs welke zijde de persoon uit bed stapt. Indien die vorige keer er wel een persoon aanwezig was, wordt er een teller voor het alarm gestart door een nieuw object van het type `Alarm` te creëren. Indien er geen persoon was in het vorige frame, gaan we bij de teller van het alarm een bij tellen door `adOne` op te roepen. Daarna gaan we de huidige persoonswaarde toekennen aan de vorige.

4.4.3 Resultaten van het experiment en besluit

De resultaten van dit programma zijn zoals we ze verwachten. er wordt gedetecteerd dat of er een persoon in bed is. Vervolgens wordt de juiste zijde weergegeven en gaat er indien nodig een alarm af. We kunnen hieruit besluiten dat dit deel van de masterproef werkt.



Figuur 4.7: Grafiek met de coördinaten van het hoofd (vierde set frames), met op de horizontale as de framenummers, de verticale de waarden

**Figuur 4.8:** Flowchart hoofdgebaseerde detectie

Hoofdstuk 5

Algemeen besluit

De onderzoeksvraag van deze masterproef was: Is het mogelijk om door middel van een visiesysteem, te detecteren langs welke zijde patiënten uit bed stappen en hoelang ze wegblijven? Een mogelijke uitbreiding is het toevoegen van valdetectie.

Ons systeem maakt gebruik van een infrarood camera, dit onderzoek zou nog eens gedaan kunnen worden maar dan voor een ander type camera. De detectie van het uit bed stappen en langs welke zijde dit gebeurt werkt. Doordat we gebruiken van een infrarood camera en vervolgens een masker creëren, zijn de patiënten onherkenbaar in de beelden. Dit was één van de systeem vereisten. Er zijn een paar delen van de code die nog geoptimaliseerd kunnen worden.

De voorbereiding naar de valdetectie is niet gedaan. Hiernaar moet nog extra onderzoek gedaan worden. Zo zal men bijvoorbeeld ook moeten bekijken of men hiervoor een tweede camera nodig heeft.

De implementatie van de gedragsanalyse in een werkend systeem in de zorgsector moet nog gebeuren. Hier is eerst nog verder onderzoek naar de communicatiemogelijkheden en dergelijk nodig. Over het algemeen kunnen we besluiten dat we deze camera en gedragsanalyse samen kunnen gebruiken om tot een goede oplossing te komen van ons probleem. Ideaal zou zijn als we een soortgelijke camera vinden die geen klik geluidje maakt bij het nemen van een frame. Dit verstoort de nachtrust van de patiënten niet.

Bibliografie

- [1] m. J. N. Ahmed Yahya Dawod and J. Abdullah. Static hand gestures: Fingertips detection based on segmented images. *Journal of Computer Sciences*, 11.
- [2] A.Manzenera and J.C.Richefeu. A robust and computationally efficient motion detection algorithm based on σ - δ background estimation. Technical report, ENSTA/LEI, 32 Bd Victor, F-75739 PARIS CEDEX 15, 2015.
- [3] D. A. Asushi Shimade and R. ichiro Taniguchi. Dynamic control of adaptive mixture-of-gaussians background model. Technical report, Kyushu University, Departement of Intelligent Systems, Fukuoka, Japan, 2006.
- [4] A. G. Badri Narayan Subudhi, Susmita Ghosh. Change detection for moving object segmentation with robust background construction under wronskian framework. *Machine Vision and Applications*, 2013.
- [5] O. Barnich and M. V. Droogenbroeck. Vibe: A universal background subtraction algorithm for video sequences. *IEEE Transactions on Image Processing*, 20(6):1709–1724, 2011.
- [6] S. Benton. Background subtraction, part 1: Matlab models. *EE Times*, 2008.
- [7] M. Brezovan and C. Badica. A review on vision surveillance techniques in smart home environments. Technical report, Dept. of Computers and Information Technology, University of Craiova, Craiova Romania, 2014.
- [8] J. W. Davis and M. A. Keck. A two-stage template approach to person detection in thermal imagery. Technical report, Dept. of Computers Science and engineering, Ohio State University, Columbus OH 43210 USA, 2005.
- [9] J. W. Davis and V. Sharma. Robust background-subtraction for person detection in thermal imagery. Technical report, Dept. of Computers Science and engineering, Ohio State University, Columbus OH 43210 USA, 2004.
- [10] J. W. Davis and V. Sharma. Background-subtraction in thermal imagery using contour saliency. Technical report, Dept. of Computers Science and engineering, Ohio State University, Columbus OH 43210 USA, 2006.
- [11] J. Han and B. Bhanu. Fusion of color and infrared video for moving human detection. *Pattern Recognition Society*, 40, 2006.
- [12] i. K. K. Hendrickx W., dr.ir. Goedemé T. and ir. Debard G. Monitoren van de activiteitsgraad van bijen aan de hand van camerabeelden. Master's thesis, Lessius, Campus De Nayer, 2011-2012.

- [13] S. T. inc. Compact seek thermal, 2017.
- [14] K. Jüngling and M. Arens. Feature based person detection beyond the visible spectrum. *IEEE Computers Society Conference on Computer Vision and Pattern Recognition*, 2009.
- [15] M. Lewellen and J. Rimington. Night vision camera, 2004. US Patent D489,386.
- [16] A. Manzanera. σ - δ background subtraction ant the zipf law. Technical report, ENSTA - Elec. and Comp. Sc. lab, 32 Bd Victor, 75015 Paris, France, 2007.
- [17] D. B. Marco Cristani, Michela Farenzena and V. Murino. Background subtraction for automated multisensor surveillance: A comprehensive review. *EURASIP Journal on Advances in Signal Processing*, 2004.
- [18] A. F. T. G. Massimo Pertizzo, Alberto Broggi and M.-M. Meinecke. Pedestrian detection for driver assistance using multiresolution infrared vision. *IEEE transaction on vehicular technology*, 53:1666 – 1678, 2004.
- [19] d. G. T. d. S. S. Mathieu J. Detectie en tellen van personen in infraroodbeelden via objectcategorisatie. Master's thesis, KULeuven, Campus De Nayer, 2013-2014.
- [20] A. M. Michael Bramberger, Andreas Doblander and B. Rinner. Distributed embedded smart cameras for surveillance applications. *IEEE Computer Society*, 2006.
- [21] P. K. A. Mukesh Kumar Saini and A. E. Saddik. From smart camera to smarthub: embracing cloud for video surveillance. *International Journal of Distriubted Sensor Networks*, 2014, 2014.
- [22] M.Valera and S. Velastin. Intelligent ditribured surveillance systems: a review. *IEE Poc.-Vis. Image Singal Process.*, 152, 2005.
- [23] H. Nanda and L. Davis. Probalistic template based pedestrian detection int infrared videos. *IEEE Intelligent Vehicle Symposium*, 2002.
- [24] J. S. V. S. Nurulfajar Abd Manap, Gaetano Di Caterina and H. Yao. Smart surveillance system based on sterao matching algorithms with ip and ptz cameras. *IEEE Computer Society*, 2010.
- [25] D. Parks. Background subtraction - donovan parks, 2014.
- [26] D. S. H. B. Peter M. Roth, Helmut Grabner and A. Leonardis. On-line conservative learning for person detection. *IEEE Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, 2006.
- [27] F. B. S.Toral, M. Vargas and M. Ortega. Improved sigma-delta background estimation for vehicle detection. *Electronics Letters*, 45 no 1., 2009.
- [28] R. Vandersmissen. Night-vision camera combines thermal and low-light-level images. *Photonik international*, 2.
- [29] M. Vandersteegen. Github repository, 2017.
- [30] P. D. Z. Varcheie and G.-A. Bilodeau. Adaptive fuzzy particle filter tracker for a ptz camera in an ip surveillance system. Technical report, Dept. of Computer engineering and software engineering, Ecole Polytechnique de Montréal, Box 6079, Station Centre-Ville, Montréal, Canada, 2010.

- [31] S. Vinck and ir. S. De Beugher. Ontwikkeling van een algoritme om eigen handen te detecteren in eye-tracker beelden. Master's thesis, KULeuven, Campus De Nayer, 2015-2016.
- [32] W.-S. L. Wen-Tsuen Chen, Po-Yu Chen and C.-F. Huang. Design and implementation of a real time vide surveillance system with wireless sensor networks. *IEEE Computer Society*, 2008.
- [33] Z. Yu and Y. Chen. A real-time motion detection algorithm for traffic monitoring systems based on consecutive temporal distance. *IEEE Asian Control Conference*, 9:1594 – 1599, 2009.

FACULTEIT INDUSTRIELE INGENIEURSWETENSCHAPPEN
TECHNOLOGIECAMPUS DE NAYER
Jan De Nayerlaan 5
2860 SINT-KATELIJNE-WAVER, België
tel. + 32 15 31 69 44
iiw.denayer@kuleuven.be
www.iw.kuleuven.be



LID VAN **ASSOCIATIE
KU LEUVEN**