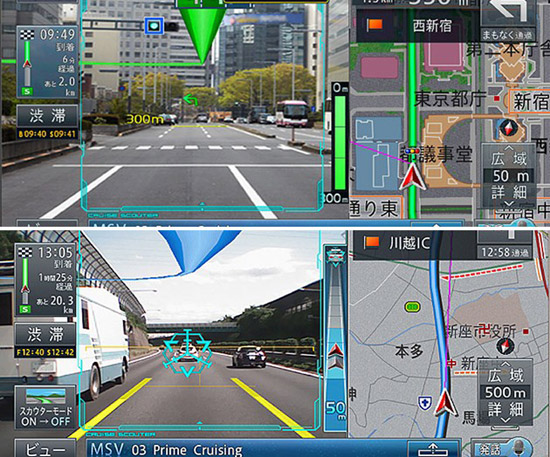
Imaging project

TI2710-D Computer Vision

B.A. van den Berg



Groep 5 – De Wijze Wieken

Raoul Harel – 4143205

Daniël Vermeulen – 4100808

Mirko Dunnewind – 4097343

Marnix de Graaf – 4172949

Tim Rensen – 4157443

Voorwoord

Onze groep, bestaande uit Technische Informatica studenten uit verschillende jaren, is begonnen aan het Imaging Project omdat we allemaal voor het Imaging variantblok gekozen hebben. Vorig jaar ging dit project over de herkenning van verschillende handposities, en daarmee een computer verschillende functies uit laten voeren. Dit jaar is het echter de bedoeling een camerasysteem te implementeren met een toepassing in het verkeer in het Mekelpark.

Iedereen uit het groepje wisselt van taken in overleg. Dit wordt besloten aan het begin van de projectochtend tijdens te vergadering. Ook wordt er af en toe thuis gewerkt.

We willen graag elkaar in het groepje, onze begeleiders en de gebruikers van de liften van EWI bedanken.

Inhoudsopgave

Voorwoord 3

Inhoudsopgave 4

Inleiding 5

Probleemstelling 5

Programma van eisen 6

Functieblokschema 7

Testplan 7

Prototype 1 8

Conclusie 9

Literatuurlijst 10

Bijlagen 11

Figuren en tabellen 12

Inleiding

Beeldherkenning wordt dagelijks en overal gebruikt. Denk aan streepjescodes, QR-codes, maar ook bijvoorbeeld aan kentekenherkenning in een flitspaal. Er wordt op dit moment veel onderzoek gedaan op beeldherkenning beter te maken. Hierbij kan je denken aan het identificeren van personen met behulp van enkel een foto. We kunnen in de toekomst misschien zelfs wel gaan denken aan een volledig automatische auto, die de complete omgeving (belijning, ander verkeer, gevaar) vaststelt en hier op reageert. Beeldherkenning gaat in de toekomst een hele belangrijke rol spelen.

Probleemstelling

Iedereen die wel eens gebruik maakt van de liften op EWI weet het wel, ze zijn ongelooflijk snel, maar het wachten op de lift kan ook ongelooflijk lang duren. Het is natuurlijk voor te stellen dat niet elke lift even vaak wordt gebruikt. Ook het aantal personen dat gebruik maakt van een lift kan erg verschillen. Voor het efficiënter maken van de liften is het handig om te weten hoe vaak een lift op een verdieping gaat en hoeveel personen er daar gebruik van maken. Door middel van het analyseren van camera beelden is het mogelijk deze data te verkrijgen. Door de camera met een live systeem in te zetten op verschillende verdiepingen en liften (niet tegelijkertijd) is het later mogelijk een totaalbeeld te krijgen van de data. Dat kan dan gebruikt worden voor de optimalisatie van het liftsysteem.

Hoe vaak en door hoeveel personen wordt één personenlift in EWI gebruikt tijdens een bepaald interval op één verdieping?

Programma van eisen

*De omgeving moet aan de volgende eisen voldoen:*

- De camera moet precies één hele persoonslift van EWI volledig in beeld hebben

- Het aantal personen in beeld moet minimaal nul en maximaal tien zijn, om het systeem goed te laten werken.

- De lichtintensiteit van de omgeving moet tussen 80 lux (vergelijkbaar met een kantoor of gang) en 10.000 (vergelijkbaar met vol daglicht) lux liggen

- Er mag geen direct licht in de webcam vallen

- De camera moet zich op 2.00 meter hoogte bevinden buiten de lift

- De camera moet zich centraal, loodrecht op de liftdeur bevinden buiten de lift

- De camera moet op minimaal 3 en maximaal 6 meter afstand bevinden buiten de lift

- De camera moet op de lift gericht zijn buiten de lift

- Camera mag niet bewegen

- Er mag niemand langer zijn dan 2.00 meter

*De eisen aan het systeem:*

- Het systeem moet mensen die de lift in- en uitlopen kunnen tellen

- Het systeem moet real-time statistieken tonen van het aantal mensen dat in en uit de lift zijn gelopen

- Het systeem moet real-time werken

- Het systeem moet kunnen draaien op een TU-computer in zaal 0.010 van Drebbelweg, of op een computer met betere specificaties

- Het beeldscherm van het systeem moet een minimale resolutie van 800x600 pixels hebben (GUI)

- Het systeem moet het openen en sluiten van de liftdeur kunnen detecteren

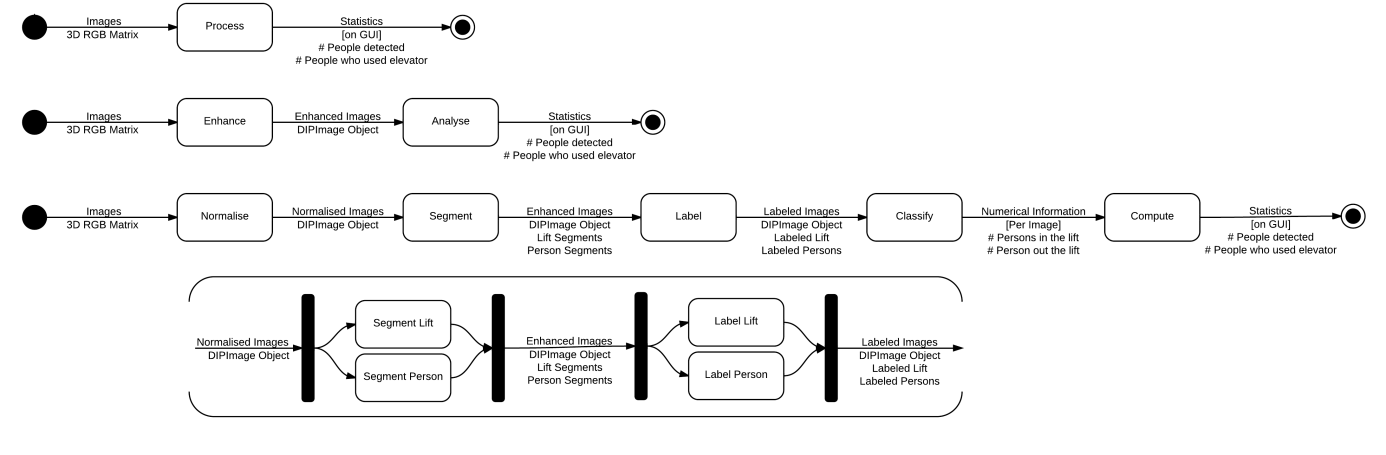
- Er moet gebruikt worden gemaakt van Matlab R2011b of R2012b

- De webcam moet op de aangesloten computer (een computer met de hierboven gespecificeerde eisen) werken

- De cameraresolutie moet 320x240 bedragen op 5 fps

- De herkenning mag maximaal 20% van het werkelijke aantal personen afwijken

Functieblokschema



In de figuren (Figuur 1) staat een grote versie van het functie blokschema.

Testplan

Voor het testen van het uiteindelijke systeem (en de eventuele eerdere prototypes) moet er aan bepaalde eisen worden gehouden. Zo zullen logischerwijs alle tests uitgevoerd worden onder de omstandigheden die in het plan van eisen staan beschreven. Buiten deze eisen is werking van het systeem niet ondersteund en dus niet gegarandeerd. Om die reden is het ook niet van belang om buiten deze eisen te testen. Het net buiten de grenzen testen van de eisen heeft wel als voordeel dat theoretisch het systeem binnen de grenzen ook werkt als de tests erbuiten slagen.

Om het systeem goed te testen zullen dus de uiterste mogelijkheden en combinaties die nog net toegestaan zijn door het plan van eisen, moeten worden getest. Hierbij wordt bij elke test simpelweg bijgehouden hoeveel procent van de ingaande en uitgaande personen er correct wordt herkend. Daarnaast moet worden opgelet of het systeem niet de mensen gaat tellen die simpelweg voorbij lopen en dus geen aandeel hebben in het op te lossen probleem.

De gegevens van het systeem en die handmatig zijn bijgehouden moeten real time worden gecontroleerd. De tests zijn geslaagd als het systeem het percentage correcte herkenningen, dat is vastgesteld in het plan van eisen, haalt.

Wanneer er niet wordt geslaagd moet worden gekeken bij welke gevallen de meeste foute of geen herkenningen waren. Door dan handmatig de beelden te analyseren kan een beter algoritme worden bedacht die het probleem oplost en tegelijkertijd niet voor nieuwe problemen zorgt.

Het testen zal worden gedaan met zowel voorgedefinieerde situaties als willekeurige situaties.

Voor de test situaties die zullen worden gegeven definiëren we het volgende:  
Herkenen/herkenning: dit houd in dat het systeem denkt dat er 1 persoon in of uit de loopt. Dit zal de respectievelijke teller ophogen met 1. Het tegenovergestelde hiervan is dus dat de tellers niet veranderen en dat het systeem dus denkt dat er niemand in of uit de lift loopt.

De voorgedefinieerde situaties spelen simpelweg in op de enkelvoudige basisfuncties van het systeem. Deze tests zullen dan ook worden uitgevoerd door ons zelf als "figurant". De volgende tests zullen minimaal worden uitgevoerd:  
- Lift is dicht en er zijn geen personen in beeld (er vind geen herkenning plaats).  
- Lift gaat open en dicht zonder dat er een personen in beeld zijn (er vind geen herkenning plaats).  
- Lift is dicht en er loopt 1 enkel persoon parallel met de muur langs (er vind geen herkenning plaats).  
- Lift is open en er loopt 1 enkel persoon parallel met de muur langs (er vind geen herkenning plaats).  
- Lift gaat open en dicht terwijl er geen personen in beeld zijn behalve 1 enkel persoon in de lift (er vindt geen herkenning plaats).  
- De lift gaat open en er stapt 1 enkel persoon uit de lift, verder zijn er geen andere personen in beeld (het systeem herkend 1 persoon die uit de lift loopt).  
- Er loopt 1 enkel persoon de lift in vanaf buiten het beeld, verder zijn er geen andere personen in beeld (het systeem herkend 1 enkel persoon dat de lift in loopt).  
- Er staat 1 enkel persoon stil (niet lopend) voor de lift terwijl die open en dicht gaat, verder zijn er geen andere personen in beeld (er vindt geen herkenning plaats).

Wanneer aan deze minimalistische tests is voldaan en het systeem voor al deze situaties werkt, is het van belang een grootschaligere test te doen. Deze test maakt gebruik van live situaties met random personen die gebruik maken van de lift(en). De enige voorwaarde die hier natuurlijk wordt gesteld is dat de random situaties binnen de eisen in het PvE vallen. Alles situaties daarbuiten worden niet meegenomen in de beoordeling van het systeem, alleen in eventuele referentie bij het verbeteren van het herkenningssysteem.

Deze laatste tests houden in dat er met meerdere mensen tegelijk in beeld wordt getest. Dit is over het algemeen een combinatie van 2 of meer van de enkelvoudige situaties die hierboven zijn gegeven. Het totaal aantal personen in beeld mag dan niet het maximaal aantal personen uit het PvE overschrijden. Uiteindelijk wanneer het systeem in minstens 5 van deze meervoudige situaties (aan de rand van de eisen) heeft voldoen kan worden gezegd dat het systeem werkt.

Prototype 1

Donderdag 22 november is prototype 1 afgeleverd. In dit hoofdstuk is te lezen wat er in dit prototype verwerkt is.

Prototype 1 bevat onder andere een werkende GUI, die vier beelden toont: Originele beeld, gefilterd beeld, verwerkt beeld en het uiteindelijke beeld. Dit is te zien in *Figuur 1*. De ‘originele input’ laat de beelden zien die rechtstreeks van de webcam naar de computer worden gestuurd. Het ‘Gefilterd’ laat het beeld na de verwerking *normalise* zien. ‘Na verwerking’ bevat het beeld na *segmentation*. Wat deze functies precies doen, wordt hieronder beschreven.

Prototype 1 bevat vier dummy-functies: *normalise*, *property*, *classification* en *count*. Voor deze operaties geldt: . Met het beeld dat binnen komt, wordt dus niets gedaan en wordt gewoon teruggestuurd door deze functies. Later sturen deze functies verwerkte beelden door naar de volgende functie, die er ook een bewerking mee doen.

*Figuur 1 - De werkende GUI van Prototype 1*

In dit prototype zitten verder nog twee functies die wel iets met het beeld doen: *labeling* en *segmentation*. Zoals te zien in *Figuur 1* maakt de functie *segmentation* het beeld zo goed als blauw. *Segmentation* zou de lift moeten herkennen qua kleur. Op dit moment worden alleen de kleuren van de lift soms met iets lichter blauw getoond. Segmentation werkt door eerst het beeld op te splitsen in 3 nieuwe beelden die de losse RGB kanalen zijn van het invoer beeld. Hier wordt vervolgens afzonderlijk een threshold op toegepast en de beelden worden weer samengevoegd. Hier zou een gesegmenteerd beeld uit moeten komen. Doordat *segmentation* niet de juiste informatie retourneert, kan de functie *labeling* (te zien bij ‘Uiteindelijk beeld’) ook haar werk (vaststellen wat de lift is) niet uitvoeren. De functie labeling werkt door een label functie uit te voeren over het beeld dat binnenkomt

**Evaluatie**

Momenteel werkt er nog niet erg veel in het prototype. We hebben een werkende GUI die in ieder geval het orginele beeld en enkele verwerkte beelden kan tonen. Ook zijn er in de code al dummy functies gemaakt om later code in te plaatsen, waardoor de structuur van het programma vanaf nu al duidelijk is. De segmentatie en labelings functies zullen flink verbeterd moeten worden willen andere functies hiervan gebruik kunnen maken, die wel nodig zijn voor het uiteindelijke doel van dit programma. De basis is nu gelegd, en nu kunnen er nieuwe functies gemaakt en verbeterd worden.

Prototype 2

Donderdag 29 november is prototype 2 ingeleverd. In dit hoofdstuk wordt het verschil met prototype 1 en de toevoegingen beschreven.

De user interface is in dit prototype iets aangepast. Er is een knop bijgekomen waarin ‘Capture Background’ staat. Bij het drukken op deze knop wordt één nieuw beeld opgeslagen dat van de webcam komt. Het is idee achter deze functie is: het opslaan van de achtergrond zonder mensen erop. Deze knop is tijdelijk, aangezien het opslaan later geautomatiseerd zal gaan worden.

Het beeld bij ‘Gefilterd’ is een binaire repre-sentatie van het verschil tussen een eerder opgeslagen beeld en de huidige invoer. Het filtert dus wezenlijk de achtergrond weg waardoor interessante objecten(mensen) overblijven. Dit wordt gedaan door van de huidige genormaliseerde invoer twee nieuwe beelden te maken: één beeld is de opgeslagen achtergrond minus het huidige beeld, en de ander het huidige beeld minus de opgeslagen achtergrond. Door op de losse RGB kanalen van deze beelden een threshold toe te passen hou je alleen nieuwe objecten in het beeld over. Het idee hiervoor komt uit ‘Tracking Groups of People’ door Stephan J. McKenna in het jaar 2000. Ideeen uit dit artikel zullen later meer gebruikt worden, bijvoorbeeld bij het daadwerkelijk kunnen tellen van mensen in groepen, omdat het doel van dit artikel redelijk overeenkomt met wat wij proberen te doen.

Bij ‘Na verwerking’ wordt het beeld weergegeven dat gemaakt is door het beeld bij ‘Gefilterd’ als mask op de huidige genormaliseerde invoer toe te passen. Hierdoor hou je dus een beeld over wat alleen nieuwe objecten laat zien, maar dan ook in kleur.

Door te normaliseren hou je een beeld over wat altijd dezelfde lichtintensiteit heeft. Dit verbetert het herkenningsproces aanzienlijk. Als er bijvoorbeeld een liftdeur opengaat is ineens veel meer licht. De camera past zich hier automatisch op aan door de intensiteit over het gehele beeld te verlagen. Hierdoor is echter de rest van de achtergrond ook donkerder, wat door de segmentatiecode wordt opgepakt als een nieuw object. Dit is uiteraard niet de bedoeling.

Het normaliseren gebeurt door op dit moment door de gemiddelde intensiteit van alle pixels en alle kleurlagen te nemen en dit dan door een lineaire verhogen of verlagen bij te stellen naar de vastgestelde standaard intensiteit van 120. Dit werkt niet helemaal correct met de zojuist geïmplementeerde methode om het beeld van de achtergrond te ontzien. Wat er dus voor het volgende prototype gedaan moet worden is het live beeld normaliseren aan de hand het gemaakte achtergrondbeeld.

Om goed te kunnen tellen hoeveel mensen er nu precies de lift in stappen is het handig om te weten of de liftdeuren open zijn. Hiervoor is het functie LiftDetect geschreven. LiftDetect gebruikt thresholding op een bepaalde kleur (namelijk de kleur van de liftdeuren) om de lift-segments uit de frame te kunnen brengen. Deze segments worden dan geanalyseerd door het functie liftVisible. LiftVisible meet een aantal eigenschappen (bijv. oppervlakte en omtrek) van deze segments om daaruit te kunnen concluderen of de liftdeuren open of dicht zijn. Deze functionaliteit is echter nog niet in het prototype gebracht omdat dit momenteel fouten oplevert.

Hoewel het wegfilteren van de achtergrond nu goed werkt moet het statistieken verzamelen nog geïmplementeerd worden. Ook moet er nog een closing toegepast worden op de verwerkte beelden om dit goed te kunnen verwerken. Verder mist er ook nog een markering wat het systeem als mens herkent, wat handig is om de telling te kunnen controleren. Kortom, nog niet alles is al geïmplementeerd, maar wat er in zit werkt al wel goed.

Conclusie

Het systeem is in staat om nuttige objecten in het beeld te onderscheiden van niet nuttige objecten. Ook is er al code ingebouwd om liftdeuren en hun status te herkennen, en om het inkomende beeld altijd constant dezelfde eigenschappen te geven. Hierdoor is de basis gelegd voor een systeem dat daadwerkelijk het aantal mensen kan tellen dat in en uit een lift in EWI lopen. Er moet nog veel gebeuren om dit daadwerkelijk goed te laten werken, en daar zal de komende weken tijd aan besteed worden.

Literatuurlijst

Tracking groups of people – Stephan J. McKenna – 2000 : <http://www.computing.dundee.ac.uk/staff/stephen/cviu2000_www.pdf>

Get wavelength from image:

<http://www.mathworks.nl/matlabcentral/answers/14587>

“Toward Face Detection, Pose Estimation and Human Recognition from Hyperspectral Imagery”:

<http://isda.ncsa.illinois.edu/peter/publications/techreports/2004/TR-20041108-1.pdf>

Background removal and human detection methods, based on video changes, color combination and selection and more:

<http://www.montefiore.ulg.ac.be/~barnich/files/barnich10thesis.pdf>

Body part recognition:

<http://www.cs.dartmouth.edu/~cs104/BodyPartRecognition.pdf>

Model based recognition:

<http://www.cs.dartmouth.edu/~cs104/BodyPartRecognition.pdf>

Bijlagen

Figuren en tabellen

# Figuur 1

Dit figuur is gesplitst in twee delen aangezien het anders niet leesbaar op de pagina kon.



