Prototype 4

Donderdag 13 november is prototype 4 ingeleverd. In dit hoofdstuk wordt het verschil met prototype 3 en de toevoegingen beschreven.

Nieuw in prototype 4 is de classify die kan herkennen of iemand binnen of buiten de lift staat. Hier zijn drie methoden voor ontwikkeld. De eerste, en tevens ook de simpelste methode gebruikt het zwaartepunt van een object dat herkend is als persoon om te bepalen of de persoon in de lift staat of er buiten. Hierbij wordt simpelweg gekeken of dit zwaartepunt overlap heeft met de liftdeuren of niet. Deze methode wordt echter niet gebruikt voor prototype 4. In prototype 4 wordt de tweede ontwikkelde methode gebruikt die niet alleen de locatie van het zwaartepunt ten opzichte van de lift gebruikt, maar ook de Cartesian box van een object dat herkend is als persoon om te bepalen wat de afstand en locatie is van een persoon ten opzichte van de lift. Deze box maakt het mogelijk voor het systeem om te bepalen of het object voor of in de lift staat in het geval dat het zwaartepunt van het object overlapt met het lift-object. De derde methode die ontwikkeld is bepaalt hoeveel mensen er in of uit de lift zijn gegaan door te kijken of het middelpunt van het persoon-object in het vlak van de lift gepositioneerd is en of de oppervlakte van dat object krimpt of als er een nieuw object bij komt of deze groeit. Dit is namelijk het effect van de liftdeuren; de objecten die in de lift staan worden kleiner of groter als de lift sluit of open gaat. Het idee is om bij deze methode ook nog de Cartesian box van de persoon-objecten te gebruiken om te kijken of ze binnen of buiten de lift staan.

Met de gegevens van de classify-methode is er een compute-methode opgesteld die aan de GUI de statistieken “Ingaand verkeer”, “Uitgaand verkeer”, “Totaal verkeer”, “Aantal mensen in beeld” en “Debug” (die laat zien hoeveel gegevens er nog in het geheugen geplaatst zijn) doorgeeft, waarna de GUI deze aan de rechterkant presenteert.

[Verhaal Marnix over auto-calibratie en skelet]

Voor prototype 4 is er een nieuwe methode voor de lift-detectie bedacht, echter is deze nog niet ingebouwd in prototype 4. Het idee achter deze nieuwe methode is om niet alleen de lift simpelweg zo goed mogelijk te segmenteren en te labelen, maar deze te voorzien van een kader om de lift heen. Hiervoor wordt eerst door een threshold bepaald welk deel van het beeld de lift is, waarna eventuele ruis met een opening weggewerkt wordt. Vervolgens worden de hoekpunten van het lift-object bepaald aan de hand van de minimum en maximum waarden voor het objecten en wordt er tussen deze hoekpunten een kader getekend op het scherm. Dit kader maakt duidelijk voor de gebruiker wat het systeem herkend heeft als lift en geeft een rechthoekig vlak met duidelijk gedefiniëerde hoekpunten waar eventueel andere delen van het systeem van kunnen profiteren. Bijvoorbeeld het onderdeel dat detecteert welke mensen er gebruikmaken van de lift is hierbij gebaat, aangezien er nu een vlak beschikbaar is in de systeemgegevens waarin de persoon-objecten moeten liggen, willen deze in aanmerking komen voor de bepaling of er gebruik gemaakt wordt van de lift.

Net als voorgaande prototypes, bevat ook prototype 4 weer een groot deel segmentatie-onderdelen. De segmentatie is een van de belangrijke delen van het gehele proces. Wanneer de segmentatie niet correct werkt, kan het al snel verkeerde data voor de vervolg processen opleveren. Het is daarom van belang dat altijd de juiste data voor de vervolg processen wordt geleverd. Hieronder volgen een aantal methoden die over de duur van het project tot aan prototype 4 uitgewerkt zijn:

Methode 1

Als eerste is er geprobeerd het inkomende frame van het gekalibreerde frame af te trekken en het resultaat absoluut te maken. Dit geeft de verandering van intensiteit op de verschillende kleur lagen. Wanneer er niets in beeld is levert dit een nagenoeg zwart beeld op omdat er geen verschil is tussen de frames. De kleine verschillen die er te zien zijn is dan ook ruis en moet eruit gefilterd worden. Hiervoor wordt er een threshold toegepast, die rond de 25 is vastgesteld. Dit filtert de ruis eruit, maar laat nog wel de grotere veranderingen zien, de personen die in beeld komen dus.

Deze methode heeft twee grote nadelen. Ten eerste worden (delen van) kledingstukken met een laag contrast met de achtergrond (hier vooral grijs en rood) niet gezien als verandering omdat het contrast te laag is. Deze delen worden dan gezien als geen verandering of ruis. Als tweede geeft deze methode een grote verandering weer als de lift opengaat. Hierdoor is het niet meer mogelijk om veranderingen, die wordt veroorzaakt door personen die voor de lift staan of erin/uit lopen, waar te nemen.

Methode 2

Door de nadelen van methode 1 moest er iets toegevoegd worden. De methode maakt gebruik van het resultaat van methode 1 en combineert dit met de verandering tussen het huidige en vorige frame. Dit combineert dus de verandering met de achtergrond met de beweging in de video stream. Door beide binaire beelden met elkaar te vermenigvuldigen, komt er een beeld uit dat sowieso alleen maar veranderingen bevat.

Uiteraard worden er nog gepaste dilaties en erosies uitgevoerd voor en na het combineren van beide beelden.

Nadeel van de methode is dat wanneer een persoon in beeld niet beweegt, hij ook niet wordt herkend. Aangezien het systeem personen die de lift in of uit lopen moet herkennen en niet personen die stilstaan, is het dus nog een onderwerp van discussie of dit nadeel een probleem is. Wat er al wel gesteld kan worden is dat wanneer een deel van de persoon niet beweegt, er ook maar een deel van de persoon wordt herkend. Het herkennen van halve personen doet de vervolg stappen in het proces geen goed, daarom is deze methode niet geheel geschikt.

Methode 3

Een vlugge methode die tussendoor een mogelijkheid zou kunnen zijn was kijken naar het verschil tussen 2 beelden, waarvan de een verticaal verschoven was. Het huidige beeld wordt dan een aantal pixels naar beneden geschoven, waarna wordt gekeken naar kleine verschillen tussen het verschoven en niet verschoven beeld. Dit zou dan de beweging van een object kunnen detecteren. Maar met deze methode was niet rekenen gehouden met egale oppervlakten als de muur en de liftdeur die dan ook als bewegend zouden zien omdat een groot deel van dit oppervlakte weinig verschild als het wordt bewogen. Deze methode is dus niet nuttig.

Methode 4

Methode 3 is uitgebreid met een dilatie met een lang verticaal structurerend object. We willen personen herkennen, ook die naast elkaar staan. Teveel een dilatie in de horizontale richting zal personen met elkaar combineren wat natuurlijk niet de bedoeling is, vandaar dat het voornamelijk verticaal wordt gedileerd. Dit geeft al een stuk beter resultaat dan eerst, al zitten er nog wel gaten in de objecten, voornamelijk wanneer er sprake is van een egaal oppervlakte.

Probleem - schaduw

Schaduw wordt bij de segmentatie ook mee genomen. Dit omdat een schaduw ook een verschil met de achtergrond is en net zoals een persoon beweegt. De overweging is om dit weg te filteren. Methodes op het internet laten zien dat zogenaamde “shadow removal” best lastig is en niet vlekkeloos werkt. Het lijkt dus niet rendabel en nuttig om schaduwen te detecteren/verwijderen aangezien dit hoogstwaarschijnlijk alsnog een frame oplevert met een groot verschil op de plek van de schaduw in vergelijking met het kalibratie plaatje.

<http://ijest.info/docs/IJEST10-02-09-93.pdf>

In prototype 4 is de vierde methode geïmplementeerd, aangezien deze de beste resultaten geeft tot nu toe.