

# Meetrappport Edge Detection

Zehna van den Berg 1662506

Remco Nijkamp 1657833

Datum 13-05-2016

# 1. Doel

Doel is te kijken welke kernel het beste geschikt is voor het doen van een edge detection. Er is voor gekozen de Prewitt kernel met de Sobel kernel te vergelijken. De onderzoeksvraag is dan ook:

*Welke kernel geeft een beter resultaat voor edge detection, Prewitt of Sobel?*

Verder zal worden gekeken of er nog verschil zit tussen de verschillende formaten van de kernels.

# 2. Hypothese

Naar verwachting zal de Sobel over het algemeen beter zijn voor edge detection dan de Prewitt. Dit heeft te maken met de opbouw van de Sobel en de extra gewichten die hij bepaalde pixels meegeeft (*voor meer informatie hierover: zie implementatieplan*). Binnen de Sobel verwachten we dan weer dat kleinere kernels beter resultaat op zullen leveren, of in elk geval dat de kwaliteit van de opgeleverde afbeeldingen te weinig verschillen voor het verschil in de rekentijd die tussen de formaten optreden.

# 3. Werkwijze

Voor het uitvoeren van de metingen was het nodig het edge detection programma een aantal keer te laten draaien met verschillende afbeeldingen en verschillende kernels. In totaal zijn 3 verschillende afbeeldingen opgenomen in voor de meting, waarop verschillende hoofden te zien waren. Per afbeelding zijn 6 nieuwe afbeeldingen gemaakt, namelijk een Prewitt en een Sobel kernel van 3x3, 5x5 en 7x7 waarden.

Vervolgens werden de opgeleverde afbeeldingen per verschillende input afbeelding met elkaar vergeleken op zoveel mogelijk verschillende aspecten. Zo is gekeken naar de bruikbaarheid van de uitkomstafbeelding voor het vinden van daadwerkelijke edges nadat er ook nog een threshold overheen is gegaan.

Tot slot is er nog een timer opgenomen in de code, waardoor precies te meten was welke kernel het snelt een nieuwe afbeelding opleverde. Hierdoor kan in de beslissing welke kernel het beste geschikt is voor edge detection, ook de variabele tijd mee worden genomen. Stel dat de Prewitt net iets “duidelijker” resultaat oplevert dan de Sobel, maar wel aanzienlijk langer doet over het produceren van de afbeeldingen, kan de beslissing worden genomen dat Sobel toch beter geschikt is.

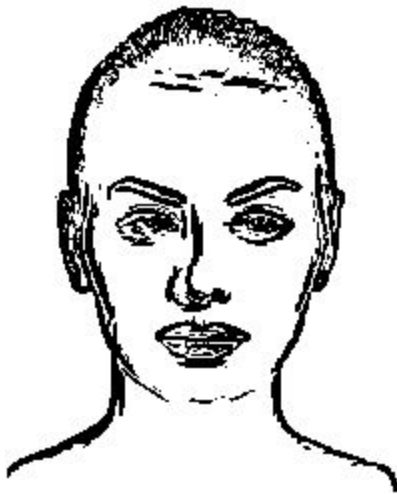
## 4. Resultaten

Hier zijn de foto's te zien die gerendered zijn met de erbij staande instellingen. Naast de foto staat ook aangegeven hoeveel tijd deze kernel ongeveer gekost heeft.

Boven de foto's staan de variabelen die gebruikt zijn en of het plaatje uiteindelijk de features eruit heeft kunnen halen.

### Prewitt:

3x3 met strength 2 (Herkend, rechteroog te laag)



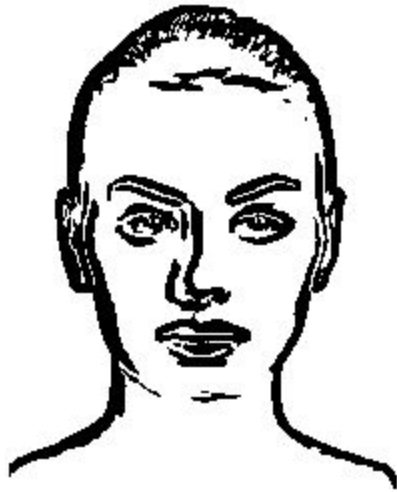
Millisecondes: 203

Factor 1.5. Alles herkend



Millisecondes: 202

5x5 met strength 0.5 (Herkend, linkeroog bij oor)



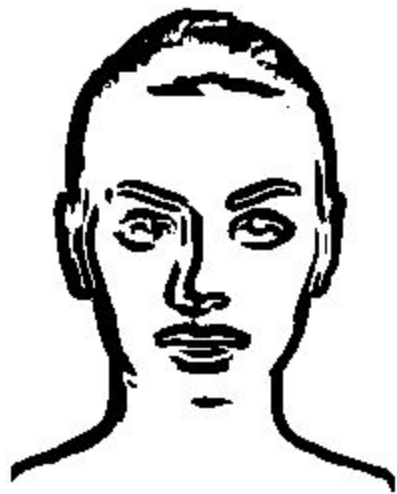
Millisecondes: 519

factor 0.5, Alles herkend



Millisecondes: 528

7x7 met strenght 0.2 (Herkend, beide ogen te laag)



Millisecondes: 1022

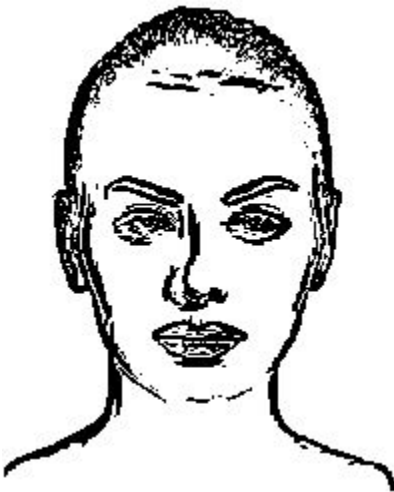
Factor 0.25, alles herkend, rechteroog wat groot.



Millisecondes: 1046

**Sobel:**

3x3 met factor 1.5 (herkend met rechteroog te laag)



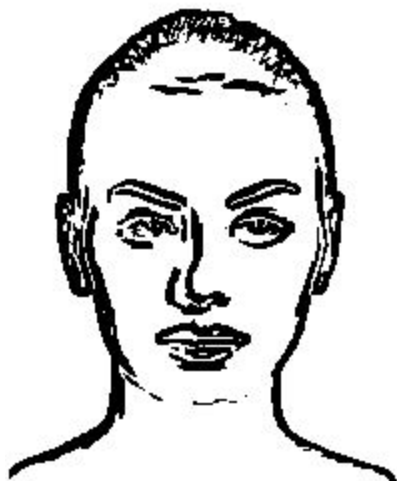
Millisecondes: 202

Factor 1, Alles herkend maar geen kin.



Millisecondes: 204

5x5 met factor 0.35 (herkend met beide ogen te laag)



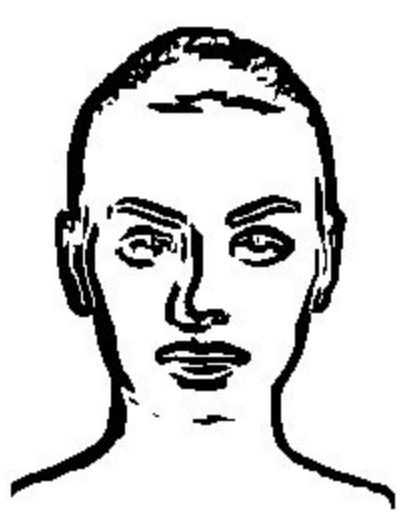
Millisecondes: 527

Factor 0.35, Herkend zonder kin



Millisecondes: 535

7x7 met factor 0.15 (herkend met beide ogen te laag)



Millisecons: 1019

Factor 0.15. Alles herkend.



Milliseconds: 1025

## 5. Verwerking

// Laat zien hoe je de meetresultaten verwerkt om een conclusie te kunnen trekken. Het is niet nodig om alle berekeningen op te schrijven, als je bijvoorbeeld maar laat zien welke formule(s) je gebruikt voor het verwerken van de meetresultaten en daar zo nodig één voorbeeldberekening aan toevoegt.



## 6. Conclusie

Zoals te zien aan de tijdsmeting is er geen verschil in de tijd tussen de Prewitt en de Sobel kernel als de afmetingen van de kernels overeenkomen met elkaar.

Bij beide soorten kernels is duidelijk te zien dat bij een kleine kernel, hier dus 3x3, nog veel kleine details overblijven die eigenlijk onnodig zijn. Bij de kernels van 7x7 blijven wel enkel de lijnen over die je zou willen hebben. Echter bij de vervolgstap worden bij zowel Prewitt als Sobel de beide ogen iets te laag gezien. Bij het gebruik van een kleinere kernel gaat dit bij Prewitt meestal wat beter, hooguit 1 oog wordt op een verkeerde positie gezien. Bij Sobel worden al snel beide ogen te laag gezien.

Er groot zijn de verschillen dus niet. Dit komt omdat de kernels eigenlijk best veel op elkaar lijken. Toch hadden we van tevoren verwacht dat er wat meer verschil te zien zou zijn als het ging om de scheiding tussen onnodige details in een plaatje en de dikte van lijnen. We hadden verwacht dat het extra beetje gewicht rond de pixel die je wilt aanpassen een grotere rol zou spelen en gunstiger uit zou pakken voor de Sobel, waardoor deze beter uit de bus zou komen.

Op basis van dit onderzoek valt niet een duidelijke conclusie te trekken over welke kernel nou beter is voor edge detection. Conclusie zou dan ook zijn dat beide kernels even geschikt zijn, uitgaande van onze experimenten.

Aangezien de verwachting was dat een grotere kernel beter resultaat op zou leveren, hadden we bedacht ergens een grens te moeten trekken tussen de tijd die de bewerking duurt en de echte bijdrage aan de verbetering van het resultaat. Dus wanneer het resultaat niet meer dusdanig beter wordt naarmate de bewerking meer tijd kost, zal een grens getrokken moeten worden. Echter bleek dat hoewel er meer details overbleven in de afbeeldingen, de overige software beter uit de voeten kon met afbeeldingen na een 3x3 matrix. Dit was vanzelfsprekend ook de snelste kernel, dus we raden 3x3 als formaat aan.

## 7. Evaluatie

Het doel van het experiment was het bekijken welke kernel meer geschikt zou zijn voor het uitvoeren van een edge detection. Daarbij werd de Prewitt tegen de Sobel uitgezet, omdat deze vrijwel dezelfde opbouw hebben en slechts in kleine details van elkaar verschillen. Gezien het feit dat deze toch beiden voorkomen gingen we ervan uit dat er ondanks de kleine verschillen tussen de kernels, toch een duidelijk verschil zou zijn in de uitkomst. Bij het uitvoeren van onze testen bleek dit echter niet naar voren te komen.

Echter hebben we wel gewerkt met een enkel plaatje van een vast formaat. Het zou kunnen zijn dat het toevallig zo uitkwam dat voor dit plaatje het verschil tussen de kernels nauwelijks naar voren kwam. Het zou dus goed mogelijk zijn dat dit bij andere plaatjes wel duidelijker zichtbaar is. Voor de volgende keer zouden dus meer plaatjes meegenomen moeten worden voor het testen.