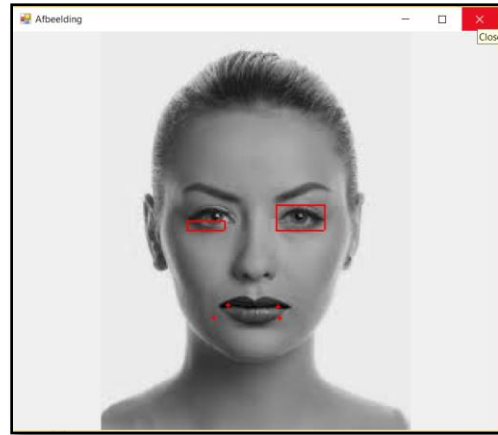


# Meetrapport Lokalisatie Correctheid

## 1. Doel

In dit document willen wij ons algoritme testen. We gaan het resultaat van onze implementatie vergelijken met de standaard implementatie. De test gaan kijken onze randdetectie implementatie de rest van de standaard implementatie helpt om beter te werken.

In de implementatie, wordt er in de post-processing lokalisatie gedaan, in de standaard implementatie kunt het algoritme soms geen lokalisatie doen. Of hij doet wel lokalisatie maar doet het niet correct. Wij nemen aan dat de randdetectie gedeelte van de implementatie hier invloed op heeft. Daarom willen we testen of onze implementatie het percentage van de correctheid van de lokalisatie verhoogt.



*Figuur 1 Onjuiste Lokalisatie*

Dat gezegd, willen wij een onderzoeksvraag opstellen:

“Hoeveel meer correctheidpercentage geeft de lokalisatie in deze geïmplementeerde randdetectie algoritme vergeleken met de standaard randdetectie implementatie?”.

Wij hopen daar door middel van deze meetrapport, deze onderzoeksvraag te kunnen beantwoorden.

## 2. Hypothese

De hypothese is dat er een kleine toename is in het slagingspercentage van de lokalisatie bij de geïmplementeerde algoritme vergeleken met standaard implementatie waarmee de correctheid van de lokalisatie ook toeneemt.

## 3. Werkwijze

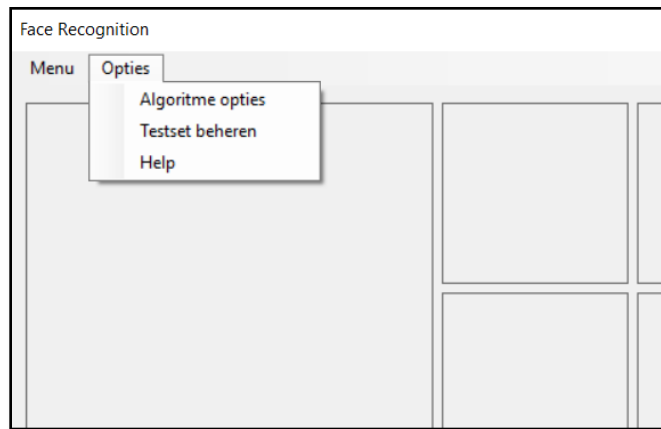
De werkwijze van deze meting is simpel. We maken gebruik van het bestaande GUI (Graphical User Interface). Hier kunnen we zelf kiezen welke gedeelte van ons eigen implementatie willen gebruiken en welke gedeelte we standaard implementatie gebruiken.

De meeting wordt getest op de beschikbare verzameling van afbeeldingen. Deze afbeeldingen zijn *child-1.png*, *female-1.png*, *female-2.png*, *female-3.png*, *male-1.png*, *male-2.png* en *male-3.png*. Deze verzameling van afbeeldingen zijn namelijk goede voorbeeld. Er is een goed diversiteit que geslacht, huidkleur en hoofd vorm.

## Stappenplan

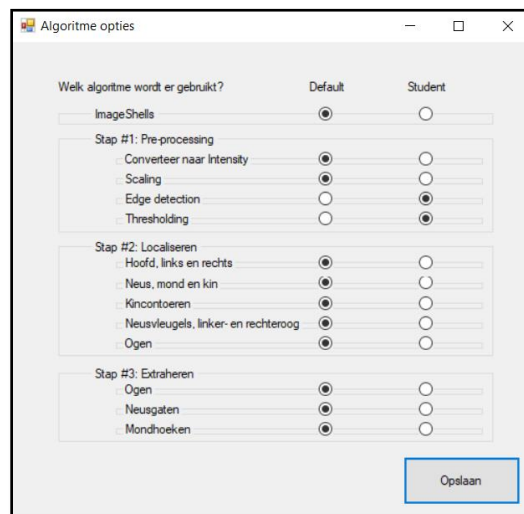
In deze eerste meeting willen we gaan bekijken of de applicatie überhaupt werkt. We gaan kijken of de lokalisatie niet mislukt (geen error krijgt) bij het gebruik. We kijken ook naar de standaard implementatie om een vergelijking te kunnen maken.

Hoe doen we deze meeting? Ten allereerste moet je de FaceRecognitionGUI.exe applicatie openen. Daarna moet je bepalen wat je eerst wilt meten, de standaard implementatie of de student/eigen implementatie. Als je student implementatie wil gaan meten dan moet je naar **Opties > Algoritme Opties**, dit staat aan de linker boven kant van de applicatie.



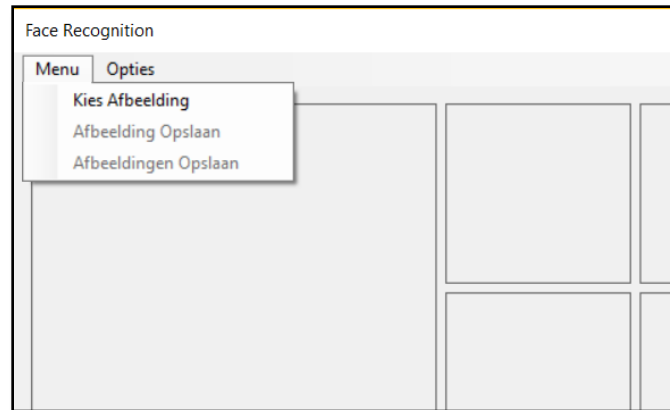
Figuur 2 Opties Selecteren

Daarna selecteer je de gedeelte van je implementatie die je wil testen, bijvoorbeeld *Edge Detection* (Randdetectie) en *Thresholding*. Druk daarna op **Opslaan**.



Figuur 3 Student algoritme selecteren

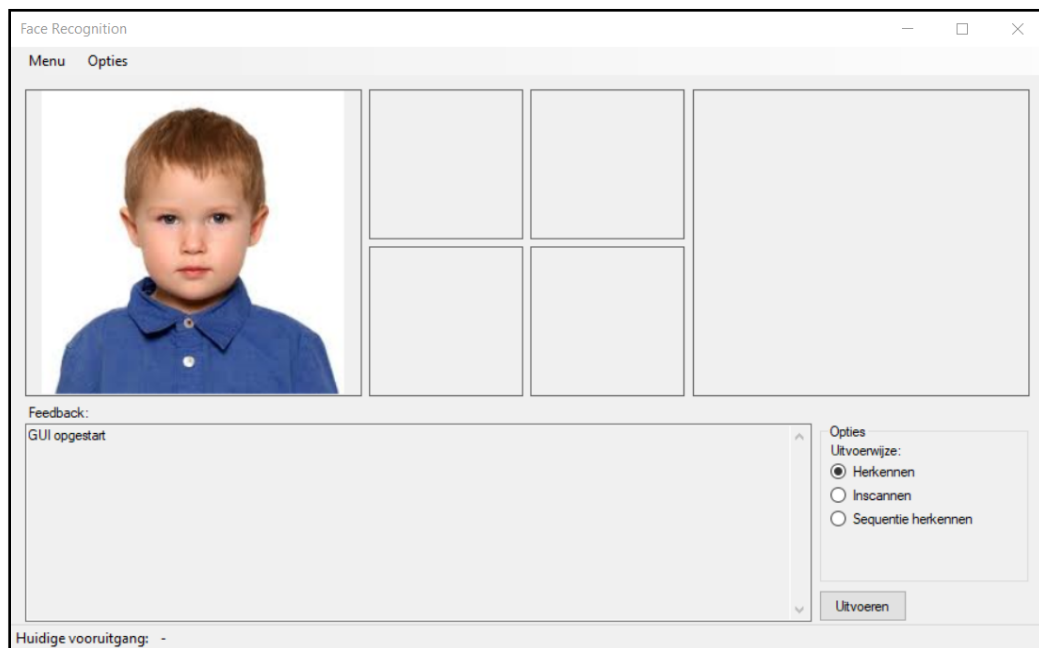
De volgende stap is de afbeelding selecteren. Om dit te doen moet je linksboven op **Menu > Kies Afbeelding**. Of je kan ook op de linksboven vierkante vakje dubbelklikken.



*Figuur 4 Afbeelding kiezen*

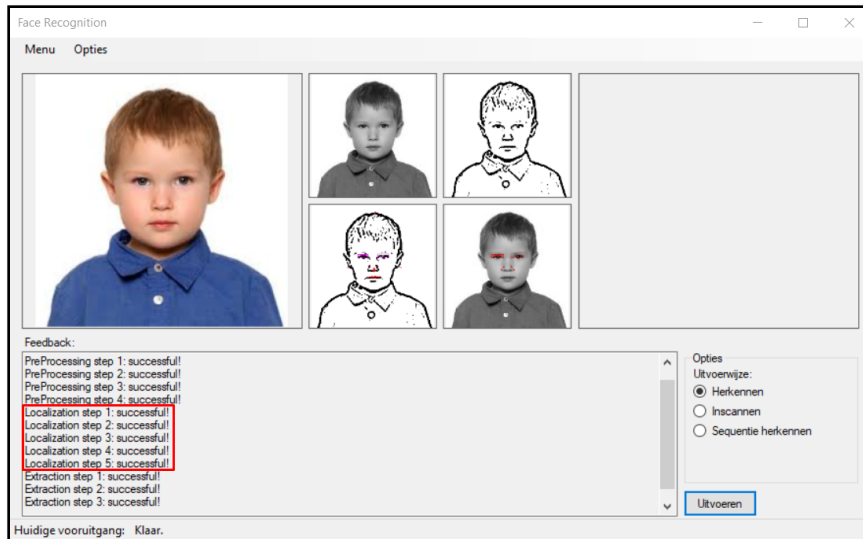
Zoek de afbeelding die je wilt testen. De standaard testset folder staat in lokale **repo > testsets > Set A > Testset Images**. Kies de afbeelding die je wilt testen en druk op **Ok**.

Daarna kies je de uitvoeringwijze. Deze is te vinden aan de rechts onder kant van de applicatie. Kies hier **Herkennen**. Als je **Herkennen** hebt geselecteerd druk je op **Uitvoeren**.



*Figuur 5 Uitvoerwijze selecteren*

Dat was de laatste stap van deze test methode. Bij het vakje **Feedback**, kan je alle resultaten zien van de gezichtsherkenning, van Preprocessing stap 1 tot en met Extraction stap 3. Hier kijken wij naar de **Localization**, van stap 1 tot en met 5.



*Figuur 6 Lokalisatie feedback bekijken*

Deze test doen wij voor alle afbeelding die er eerder benoemd zijn in deze document. We documenteren alle stappen van de lokalisatie of ze succesvol waren of niet. Hiermee kunnen we makkelijk een slagingspercentage maken.

## Stappenplan 2

Daarnaast gaan we tegelijken tijd kijken of de lokalisatie inderdaad goed is of niet. Dit moet manual bekeken worden door een mens. De applicatie kan namelijk wel zeggen dan de lokalisatie succesvol was, maar dat betekent alleen dat het gelukt is, of het daadwerkelijk juist moet kan alleen een mens bevestigen.

Om dit te doen moet je dubbelklikken op de rechtsonderste vierkante blok. Hiermee krijg je een afbeelding met rode lijnen en punten waar de lokalisatie denkt dat de ogen, neus en mondhoeken zijn. De rode vierkante zijn voor de ogen, de eerste parallelle punten zijn voor de neusgaten en de onderste twee parallelle punten zijn voor de mondhoeken.



Figuur 7 Lokalisatie is onjuist

Zoals te zien in de afbeelding hierboven (Afbeelding 5) is te zien dat de punten niet juist zijn. De applicatie denkt dat het middelstuk van de neus de neus gaten zijn en dat neushoeken de mondhoeken zijn. Terwijl de applicatie zij dat de lokalisatie en extractie succesvol was, kunnen we zien dat het niet helemaal succesvol is. Daarom moeten we het zelf bekijken of het inderdaad klopt.

#### 4. Resultaten

De eerste test die gedaan is, is het bekijken of de lokalisatie stappen succesvol zijn. Uit de test blijkt dan beide implementaties 100% slaggingspercentage hebben.

Afbeelding	Standaard Stap					Student Stap				
	Stap 1	Stap 2	Stap 3	Stap 4	Stap 5	Stap 1	Stap 2	Stap 3	Stap 4	Stap 5
Child-1.png	S	S	S	S	S	S	S	S	S	S
Female-1.png	S	S	S	S	S	S	S	S	S	S
Female-2.png	S	S	S	S	S	S	S	S	S	S
Female-3.png	S	S	S	S	S	S	S	S	S	S
Male-1.png	S	S	S	S	S	S	S	S	S	S
Male-2.png	S	S	S	S	S	S	S	S	S	S
Male-3.png	S	S	S	S	S	S	S	S	S	S
Total Succes	100% Succes					100% Succes				

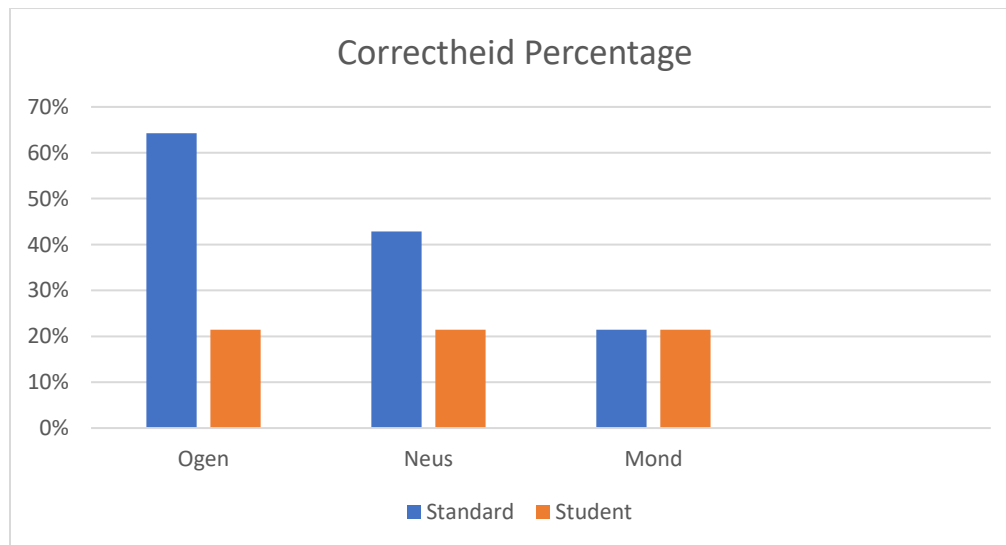
Figuur 8 Lokalisatie stappen test

Daarna hebben we tuurlijk de tweede methode van testen gebruikt, hier bekijken we zelf of de lokalisatie inderdaad goed is. We bekijken de twee ogen, twee neus gaten en de twee mondhoeken. In deze tabel staat het getaal nul tot en met 2 voor het aantal juiste lokalisatie.

Afbeelding	Standaard Lokalisatie			Student Lokalisatie		
	Ogen	Neus	Mond	Ogen	Neus	Mond
Child-1.png	2	0	0	0	0	2
Female-1.png	2	2	1	2	1	1
Female-2.png	2	2	2	0	0	0
Female-3.png	1	0	0	1	0	0
Male-1.png	2	0	0	0	0	0
Male-2.png	0	0	0	0	0	0
Male-3.png	0	2	0	0	2	0
Goede Lokalisatie	9	6	3	3	3	3
Total	18			9		

Figuur 9 Correctheid van lokalisatie

Aan deze tabel vallen een paar dingen op. Ten eerste is dat bij de standaard implementatie *female-2.png* de enige is die 100% correct is, terwijl ze in de studenten implementatie 0% correct is. *Male-3.png* is de enige afbeelding die voor beide implementaties hetzelfde resultaat heeft. Hieronder hebben we nog een tabel gemaakt om duidelijk te maken waar het meeste mis gaat.



Hier is te zien dat de standaard implementatie beter resultaat heeft over het algemeen. Hij herkent ogen het meeste, met 64%, daarna de neusgaten met 43% en op het laast de mondhoeken met 21%. Aan de andere kant heeft de student implementatie voor alle onderdelen een correctheid van 21%.

## 5. Verwerking

Alle resultaten die verwerkt zijn tweekeer gemeten. Het zou accurater kunnen door deze meerdere keren te doen maar aangezien er manual naar gekeken moest worden is het niet gedaan.

Daarnaast hebben de resultaten minimale/geen verschil tussen meerdere testen, dus de resultaten zijn vrij statisch.

Door de resultaten met de hypothese te vergelijken kunnen we niet zeggen of het slagingspercentage is verbeterd. Dit is omdat het slagingspercentage van de standaard implementatie 100%, en onze implementatie ook.

Verder kunnen we wel zien dat de correctheid van de lokalisatie is gedaald. Dit is het omgekeerde dan onze hypothese, en heeft ook ons verbaasd. We dachten dat door een betere randdetectie, de lokalisatie zou helpen om een accuratere inschatting te kunnen maken. Maar dit is blijkbaar dus niet waar.

## 6. Conclusie

Door de resultaten te bekijken kunnen we zien dat ons eigen implementatie zowel de standaard implementatie 100% succes lokalisatie kunnen doen. Hierdoor kunnen we helaas dus geen vergelijking voor maken over welke beter is.

Maar we kunnen wel de daadwerkelijke correctheid vergelijken. De standaard implementatie heeft 18 correcte lokalisaties kunnen maken, terwijl onze implementatie heeft er maar 9 correct. Dit maakt de standaard implementatie 2 keer zo goed als onze implementatie. Hiermee kunnen we ook een conclusie maken dan een betere randdetectie voor de oog van een mens niet betekent dat het ook beter is voor een computer (in computer vision).

Het antwoord op onze hoofdvraag “Hoeveel correctheidpercentage geeft de lokalisatie in deze geïmplementeerde randdetectie algoritme vergeleken met de standaard randdetectie implementatie?” is dus 50% minder.

## 7. Evaluatie

Bij het maken van het implementatieplan hadden we een duidelijk doel voor ogen. Dit doel was om een werkende vorm van randdetectie te implementeren door middel van een eigen implementatie. Dit is uiteindelijk zeker gelukt en sommige onderdelen van onze implementatie zijn zo te zien ook is het slagingspercentage van de lokalisatie.

Alleen jammer dat dit niet de correctheid van de lokalisatie heeft geholpen. Wij vinden dat onze randdetectie algoritme wel een beter resultaat geeft, en daarom zou het de lokalisatie correctheid moeten verbeteren. Maar dit blijkt dus niet waar te zijn, wij denken dus dat onze implementatie niet goed synchroniseert met de rest van de algoritme waardoor we dus geen goed resultaat tonen of dat het alleen beter is voor ons oog maar niet voor die van de computer (computer vision).

Misschien hadden we ook van tevoren onderzoek moeten doen hoe de lokalisatie werkt, aangezien ons doel was om deze te steunen om beter te helpen. Als we beter wisten hoe hij precies werkte hadden we misschien betere doelgerichte implementatie kunnen maken voor de lokalisatie gedeelte. We hadden dus meer onderzoek van tevoren moeten doen en niet te snel gaan implementeren.