

# Onderzoeksverslag grijsschaal algoritmes Vision



# Inhoudsopgave

Inhoudsopgave	2
Auteurs	3
Voorwoord	3
Korte samenvatting van het onderzoek	4
Gekozen algoritmes	4
Intensity methode	4
Luminance	4
Value	5
Nulmeting resultaten	5
<b>Intensity methode resultaten</b>	<b>6</b>
<b>Luminance methode resultaten</b>	<b>7</b>
<b>Value methode resultaten</b>	<b>8</b>
<b>Conclusie</b>	<b>9</b>
Bronnen	10

# Auteurs

Oscar Kromhout  
Gorinchem  
[oscarkromhout@gmail.com](mailto:oscarkromhout@gmail.com)

Justin van Ziel  
Amersfoort  
[justin.van.ziel@gmail.com](mailto:justin.van.ziel@gmail.com)

# Voorwoord

datum: 18-3-2020

Dit document is opgesteld om de onderzoeksresultaten van ons grijsschaal onderzoek weer te geven. Het onderzoek wordt uitgevoerd zoals omschreven in het [implementatieplan Vision](#). De eerste bedoeling was om 50 foto's te testen. Uiteindelijk hebben de auteurs van dit document gekozen om 100 foto's te testen. Het standaard algoritme om gezichten te herkennen dat ons is aangeleverd bleek zodanig slecht van kwaliteit dat we bij vijftig foto's te kleine verschillen kregen om een valide uitkomst uit het onderzoek te kunnen krijgen. In de Google spreadsheet [onderzoeksresultaten](#) zijn de rauwe uitkomsten van onze testen te vinden.

Verder is opvallend dat we tijdens het maken van ons onderzoek veel willekeurige foutmeldingen kregen in de standaard aangeleverde code. Dit kostte veel tijd om omheen te werken. Ook leek de standaard implementatie sommige foto's niet in te willen laden waardoor we foto's handmatig moesten kiezen. Wij hebben bij het handmatig kiezen zoveel mogelijk gekozen voor heldere foto's, waarbij de ogen goed zichtbaar zijn, en men zo recht mogelijk in de camera kijkt. Kortom, de standaard implementatie is van zeer lage kwaliteit.

De "main.cpp" hebben wij ook herschreven zodat we de aangeleverde GUI niet hoefden te gebruiken. Ook deze werkte matig en het was niet geschikt om grote testsets te testen.

Nadat we de bovenstaande tegenvallers hebben opgelost kwamen we uit op de onderstaande onderzoeksresultaten.

# Korte samenvatting van het onderzoek

Wij willen onderzoeken welk grijschaal algoritme het beste werkt om het naar ons aangeleverde standaard gezichtsherkenning algoritme beter te laten werken. Hiervoor hebben wij uit een [wetenschappelijke paper](#) drie algoritmes gekozen om te kijken hoe goed zij dit doen ten opzichte van elkaar. Om dit te realiseren hebben wij deze algoritmes geschreven in C++. Vervolgens gebruiken wij honderd foto's uit een [testset](#). Deze foto's hebben wij handmatig geselecteerd uit de grotere [testset](#) zoals hierboven staat omschreven.

Hierna passen wij in stappen van 5% drie keer de belichting aan op de foto, zowel omhoog als naar beneden. We maken de foto's dus 5, 10 en 15 procent lichter en donkerder. Dan laten we de drie gekozen algoritmes in samenwerking met de standaard aangeleverde algoritmes door de honderd foto's heen hun werk doen en bekijken we op hoeveel procent van de foto's gezichten herkend worden en op hoeveel procent dit niet herkend wordt.

Aan het einde van het onderzoek hopen we dus een uitspraak te kunnen doen welk algoritme het beste kan worden gebruikt om foto's om te zetten naar zwart wit beeld om de rest van de gezichtsherkennings algoritmes beter hun werk te laten doen.

## Gekozen algoritmes

### Intensity methode

Intensity methode is een simpel 'color-to-grayscale' algoritme. Het idee van dit algoritme is van iedere pixel het gemiddelde nemen van de RGB waarde. Om zo een hoger contrast te krijgen tussen donker en licht. Dan zou je dus ook beter de edges in de foto moeten kunnen zien.

$$\mathcal{G}_{Intensity} \leftarrow \frac{1}{3}(R + G + B).$$

Intensity leek ons goed om te onderzoeken omdat dit veruit de simpelste methode van grijschalen is. Gewoon het gemiddelde van de drie waarden. Die kunnen we dan dus gebruiken in de testresultaten om te zien of een simpel algoritme slechter is dan een meer ingewikkelde variant. Zoals ons gevoel zegt

### Luminance

Deze methode leek ons interessant om te proberen omdat hier niet meer naar de menselijke perceptie van helderheid wordt gekeken maar dit wel probeert te bereiken. Luminance wordt volgens ons literatuuronderzoek veel gebruikt bij "standaard" image

processing software en het leek ons daarom interessant om ook deze te bekijken. Het nadeel is dat deze kennelijk veel gebruikt wordt, dus misschien gaan we nu iets onderzoeken wat al meermalen onderzocht is.

$$\mathcal{G}_{Luminance} \leftarrow 0.3R + 0.59G + 0.11B.$$

Luminance hebben we gekozen omdat hij veel gebruikt wordt bij “standaard” fotobewerkingssoftware. Die is dus zinnig om te bekijken. Want dat betekent dat het makkelijk is om foto’s beter zichtbaar te maken voor recognition systemen

## Value

Deze methode bestaat uit twee stappen en gebruikt ook gammacorrectie om te schalen. Het rekent eerst het maximum van de R-G-B waarden uit om daarna een gammacorrectie daarvan te doen (zie formules). Deze is interessant om te bekijken omdat hij uit meer stappen bestaat en dus wat ingewikkelder is. We kunnen dan gelijk kijken of het gebruiken van gamma op deze manier om te schalen ervoor zorgt dat gezichten beter herkenbaar worden voor computers.

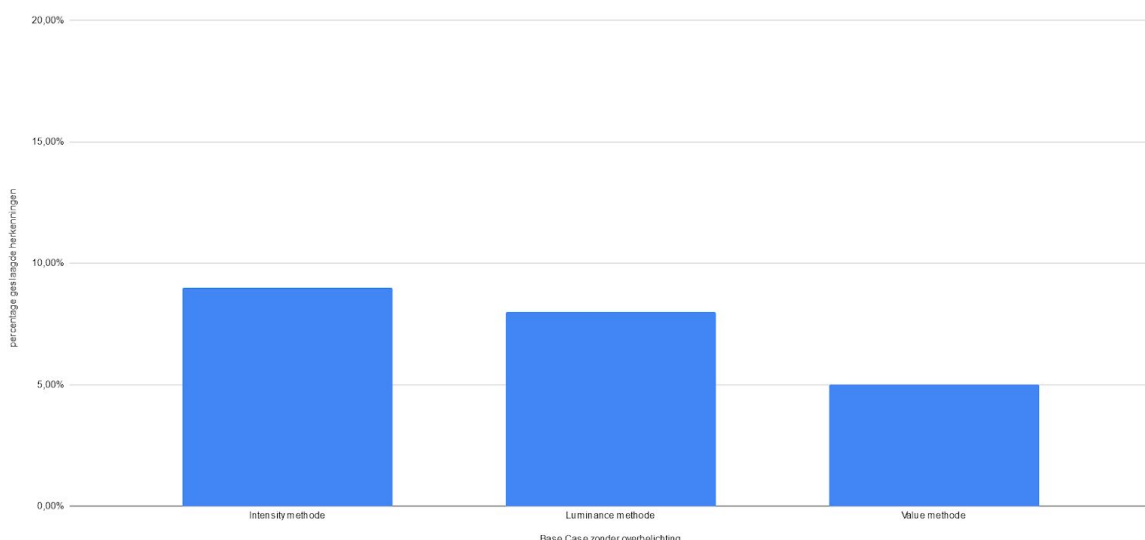
$$G_{Value} = \max(R, G, B).$$

$$\Gamma(\max(R, G, B)) = \max(R', G', B').$$

## Nulmeting resultaten

Hieronder staat een staafdiagram waarin te zien is hoe goed ieder afzonderlijk grijschaal algoritme ervoor zorgt dat de standaard algoritmes de gezichten kunnen herkennen. Hierbij zijn de foto’s niet overbelicht. Hieruit kunnen we opmaken dat de herkenningsoftware niet al te best werkt.

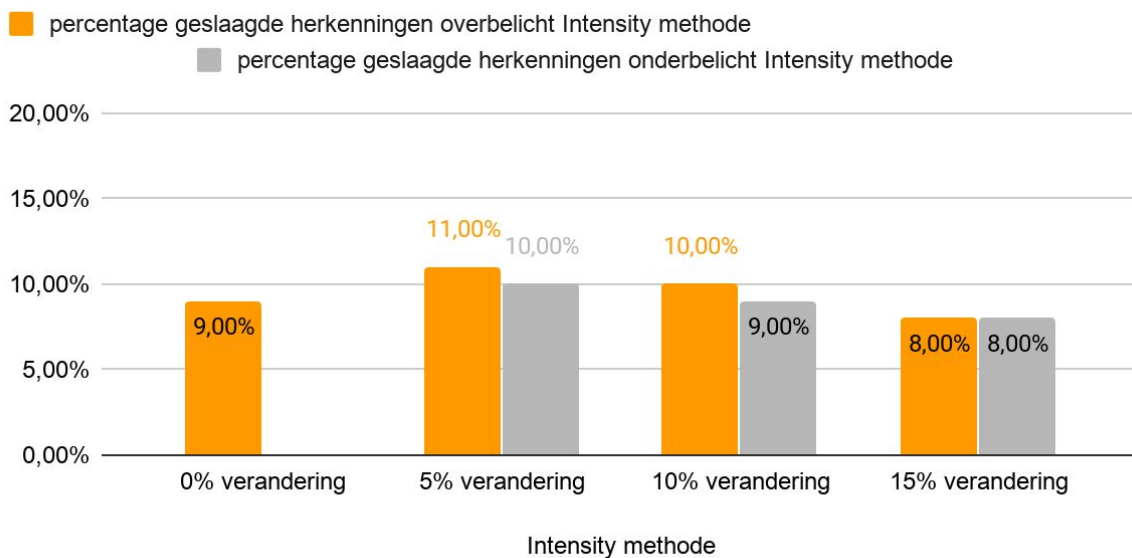
percentage geslaagde herkenningen versus Base Case zonder overbelichting



# Intensity methode resultaten

Hieronder wordt een staafdiagram weergegeven met resultaten volgens de Intensity methode. Het wordt duidelijk wat voor effect overbelichting en onderbelichting heeft op de resultaten van de aangeleverde gezichtsherkenning. Opvallend is dat de 5% verandering naar beide kanten van het spectrum een positief effect lijkt te hebben.

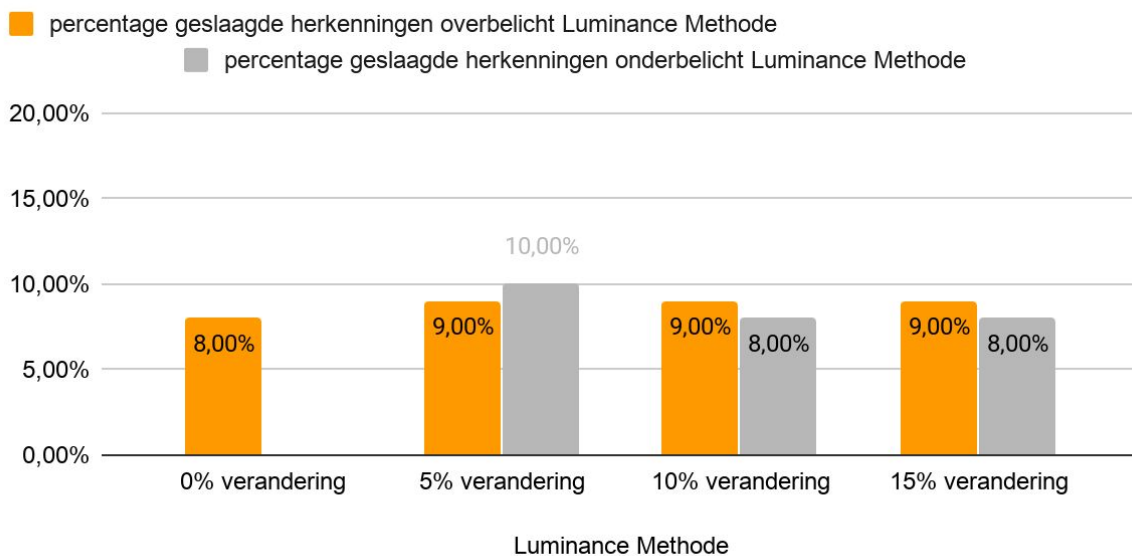
## percentage geslaagde herkenningen overbelicht en percentage geslaagde herkenningen onderbelicht



# Luminance methode resultaten

Hieronder wordt een staafdiagram weergegeven met resultaten volgens de Luminance methode. Het wordt duidelijk wat voor effect overbelichting en onderbelichting heeft op de resultaten van de aangeleverde gezichtsherkenning. Opvallend is dat het verschil in overbelichting nauwelijks uitmaakt voor de uitkomsten.

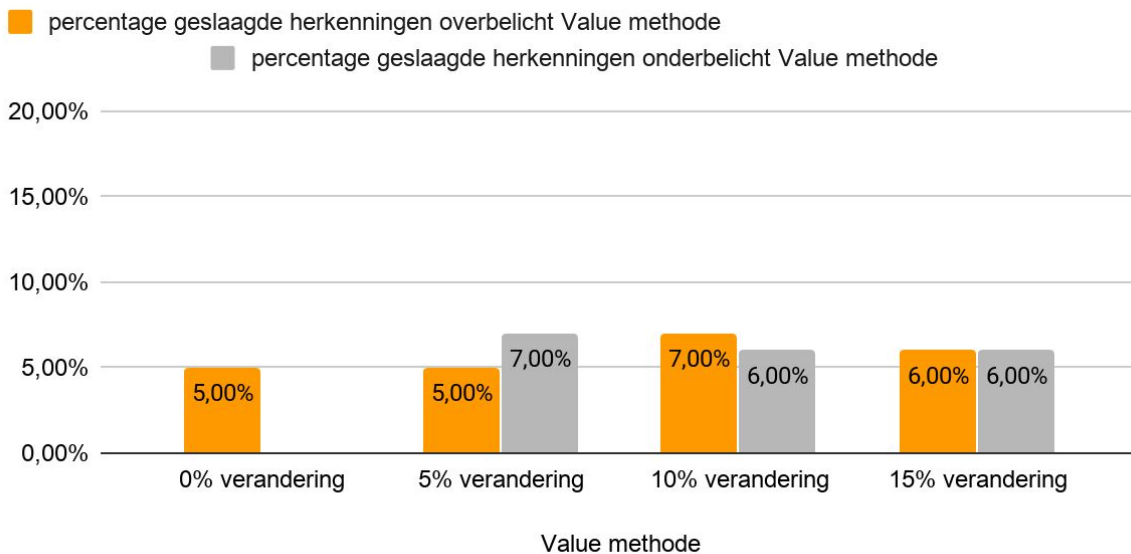
## percentage geslaagde herkenningen overbelicht en percentage geslaagde herkenningen onderbelicht



# Value methode resultaten

Hieronder wordt een staafdiagram weergegeven met resultaten volgens de Value methode. Het wordt duidelijk wat voor effect overbelichting en onderbelichting heeft op de resultaten van de aangeleverde gezichtsherkenning. Verder is opvallend dat deze methode over het algemeen slechter werkt dan de andere twee. De verschillen zijn groter.

## percentage geslaagde herkenningen overbelicht en percentage geslaagde herkenningen onderbelicht

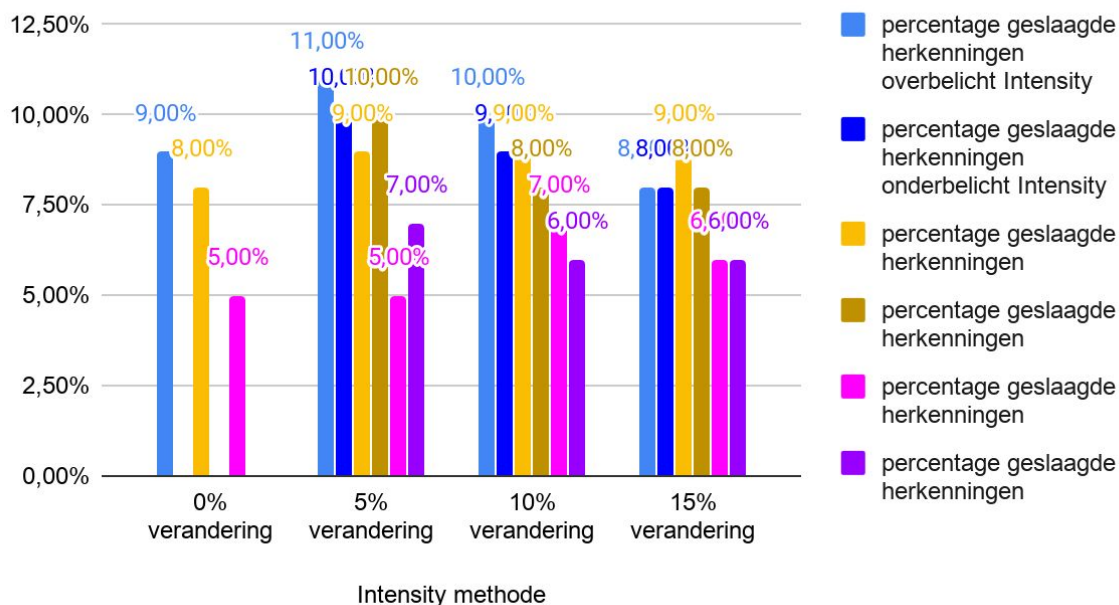




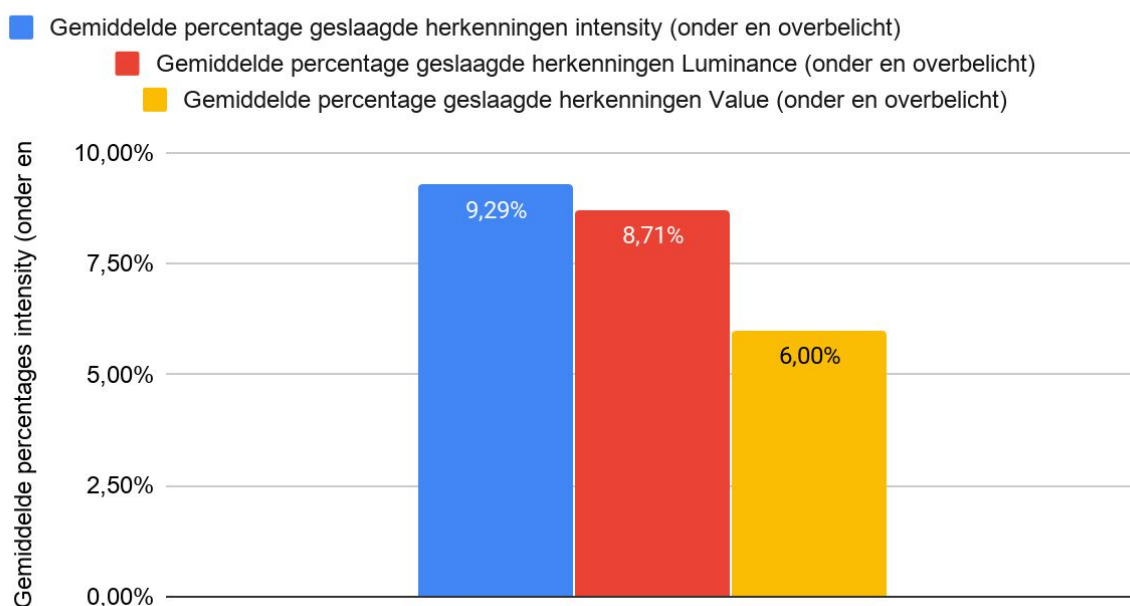
# Conclusie

Hieronder staan nog twee diagrammen. De eerste laat alle afzonderlijke resultaten naast elkaar zien. De tweede laat het gemiddelde aantal herkende foto's zien van alle algoritmen.

## Alle percentages van de verschillende methodes naast elkaar



## Gemiddelde percentages intensity (onder en overbelicht)



Met name het laatste staafdiagram laat ons een wat betere verdeling zien. We kunnen uit dit diagram opmaken dat het Value algoritme, getest op honderd foto's in drie stappen van 5% onder en overbelichting gemiddeld genomen 3,29% en 2,71 % slechter presteert dan respectievelijk de Intensity en luminance methode.

Oftewel, van het Value algoritme durven we de stelling te maken dat het slechter presteert dan de andere twee. Vooral in het eerste diagram zien we dat het bij alle belichtingen minder foto's laat herkennen dan de andere twee algoritmes. Het intensity en Luminance algoritme zitten echter zo dicht bij elkaar dat we niet durven zeggen of de oorzaak hiervan aan het grijsschaal algoritme ligt, of aan de aangeleverde standaard algoritmes voor gezichtsherkenning. Wat wel opviel aan de Intensity methode is dat bij een 5% overbelichting, de software 2 extra foto's ging herkennen ten opzichte van de nulmeting. Zo'n groot verschil werd bij de andere algoritmes niet gehaald.

Toch is naar onze mening ook het value algoritme eigenlijk nog niet goed getest. Volgens het gezegde moet men "roeien met de riemen die je hebt." En dat hebben we in dit onderzoek ook gedaan. De verschillen zijn erg klein en wellicht hadden we met betere standaard software ook een betere conclusie kunnen trekken.

## Bronnen

- Kanan, C., & Cottrel, G. W. (2012, 10 januari). *Color-to-Grayscale: Does the Method Matter in Image Recognition?* Geraadpleegd op 21 februari 2020, van [Color-to-Grayscale: Does the Method Matter in Image Recognition?](#)
- [Large-scale CelebFaces Attributes \(CelebA\) Dataset](#) (open dataset met foto's)
- Oscar Kromhout & Justin van zien (2020, 18 march). *Onderzoeksresultaten ruwe data* Geraadpleegd op 18 march 2020, van <https://docs.google.com/spreadsheets/d/1v-DxDxYtIUpD-T4IKvVIPyWnxk3RTo7bLt-uPCyl8o/edit?usp=sharing>