Lichtintensiteit, Resolutie en saturatie invloed op grayscale algoritmes gebruikt in facial recognition

# Namen en Datum

Mick Bos  
Ties Brouwer

12-03-2020

# Doel

Het doel van dit onderzoek is om te testen welk grayscale conversie algoritme beter werkt dan de standaard implementatie, met blik op verschil in lichtintensiteit, saturatie en resolutie

# Hypothese

Dit onderzoek werkt met verschillende grayscale conversie algoritmes die onderling verschillende kwaliteitseigenschappen hebben (zie: Werkwijze). Voor de verschillen in Lichtintensiteit, Saturatie en Resolutie waardes maken wij de volgende verwachtingen voor de algoritmes

* **Lichtintensiteit**Wij verwachten dat het *Luma algoritme* de beste resultaten zal leveren. Dit algoritme verkrijgt zijn resultaat, net zoals *Luminance,* door de RGB-waardes met vaste waardes te vermenigvuldigen. Het verschil ligt erin dat *Luma* haar resultaten duidelijkere verschillen tussen lichtere kleuren bevatten.
* **Saturatie**

Wij verwachten dat het *Value algoritme* de beste resultaten zal leveren. Value haalt zijn werking door de RGB-kanalen gelijk te trekken, wat overheen komt met het verhogen/verlagen van het *value* kanaal uit het HSV color space. De werking van dit algoritme zou moeten kunnen vermijden dat saturatie verschillen impact op het resultaat zullen leveren.

* **Resolutie**Aangezien resolutie verschil geen invloed heeft op de individuele RGB-kanalen verwachten wij dat de algoritmes in hun resultaat het meest zullen verschillen in hun mogelijkheid tot het creëren van geleidelijk kleurverschil. Hierdoor verwachten wij, net als bij lichtintensiteit, dat het *Luma algoritme* de hoogste Hit/miss ratio zal leveren. Er wordt bij de *Luma* resultaten extra nadruk overkleurverschil gezet, waardoor de facial recognition software gemakkelijker features zou moeten kunnen herkennen.

# Werkwijze

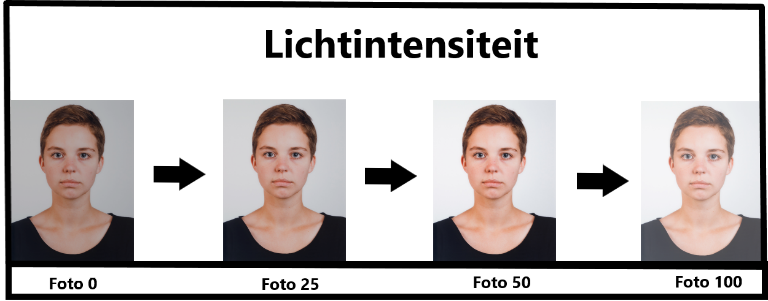
Voor dit onderzoek worden de volgende algoritmes1 getest:

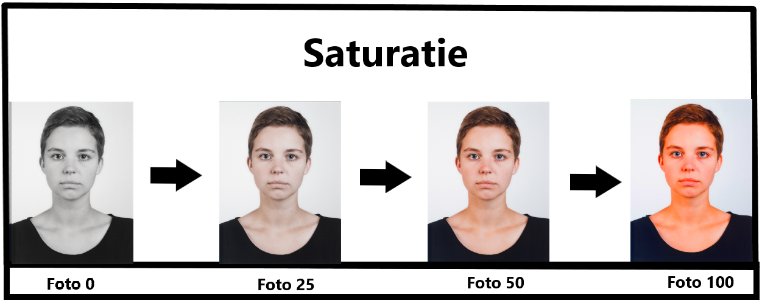
* **Intensity Algorithm**   
   Result = ((1/3) \* (R + G + B))
* **Value Algorithm**  
   Result = MAX(R, G, B)
* **Luminance Algorithm**   
   Result = (R \* 0.3) + (G \* 0.59) + (B \* 0.11)
* **Luma Algorithm**   
   Result = (R \* 0.2126) + (G \* 0.7152) + (B \* 0.0722)
* **Luster Algorithm**  
   Result = (1/2) \* (MAX(R, G, B) + MIN(R, G, B))

De verschillende algoritmes worden voor Lichtintensiteit, Saturatie en Resolutie ieder getest op een bijpassende testset. Elke testset bestaat uit een verzameling van 5 verschillende portretten (figuur 1), waarvoor voor elk portret 100 versies worden getest. Deze versies lopen van een factor (0) voor foto 0 naar een factor (100) voor foto 100, waarbij de factor gelijkstaat aan de waarde levels van hun corresponderende invloed. ( figuur2)

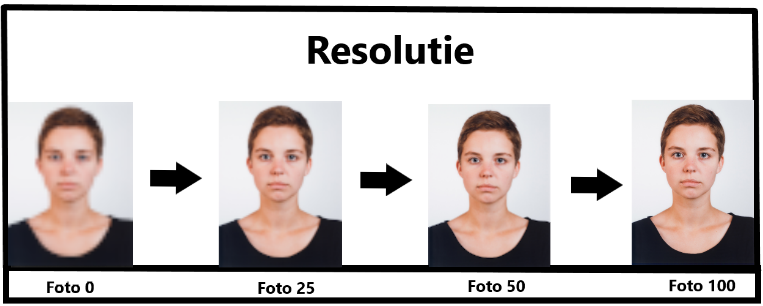


*Figuur 1*





*Figuur 2*



De verschillende software steps waar de Hitratio van de metingen op wordt gebaseerd loopt van stap 1 tot 12 als volgt:

1. PreLocalization
2. Localization (1)
3. Localization (2)
4. Localization (3)
5. Localization (4)
6. Localization (5)
7. Prepare extraction
8. Extraction (1)
9. Extraction (2)
10. Extraction (3)
11. Extraction (4)
12. PostProcessing

De algoritmes worden op de volgende wijze individueel getest:

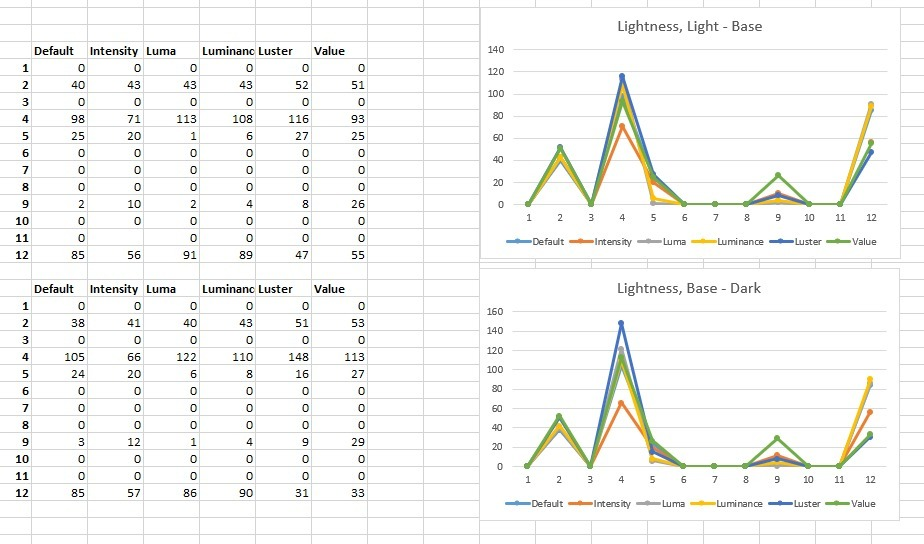
* Voer voor lichtintensiteit, saturatie en resolutie de volgende stappen uit:

1. Verwerk het eerste algoritme in de facial recognition software als grayscale conversie wijze.
2. Laad de eerste testset in en gebruik het huidige algoritme om dit portret te verwerken tot grayscale. Het algoritme slaat nu op welke software steps er mogelijk waren voor de resulterende grayscale.
3. Herhaal stap 2 tot alle testsets gemeten zijn.
4. Herhaal stap 1 tot en met 3 voor de overige algoritmes.

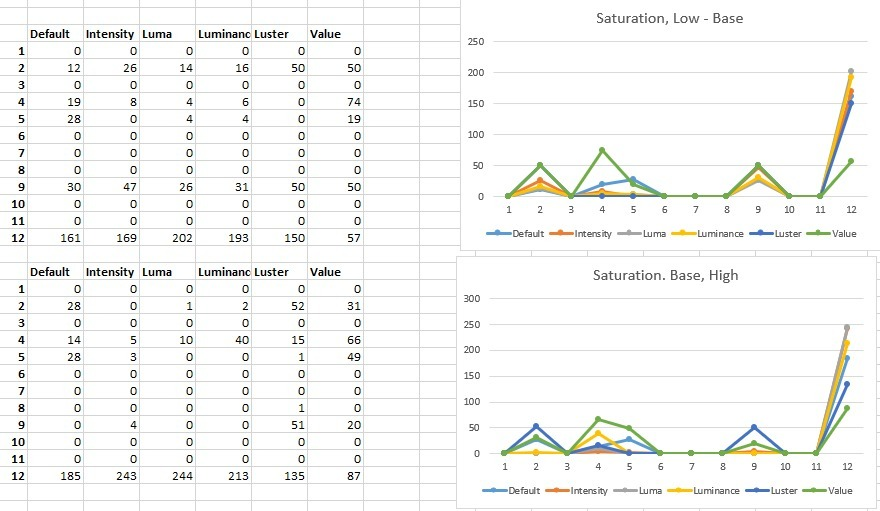
# Resultaten

De volgende 3 meetresultaten zijn afleidingen van de verkregen data. De tabellen aan de linkerkant staan voor hoeveel steps een algoritmes over al zijn testsets heeft verkregen. Rechts staan de grafieken die erop gebaseerd zijn.

# Resultaten Lichtintensiteit



# Resultaten Saturatie



# Resultaten Resolutie

# 

# Verwerking

Uit de meetresultaten is een conclusie te trekken op basis vanwaar de pieken in de grafiek te vinden zijn. Hierbij is het belangrijk hoe de verschillende algoritmes met elkaar vergelijken op de grafiek pieken.

# Verwerking Lichtintensiteit

**Light – Base:**

* **Default:** Default haalt bij het localizeren de meeste images correct door in het lichte spectrum. Daarnaast staat het gelijk met meerdere algoritmes bij het einde
* **Intensity:**
* **Luma:**
* **Luminance:**
* **Luster:**
* **Value:**

**Base – Dark:**

* **Default:**
* **Intensity:**
* **Luma:**
* **Luminance:**
* **Luster:**
* **Value:**

# Verwerking Saturatie

**Low – Base:**

* **Default:**
* **Intensity:**
* **Luma:**
* **Luminance:**
* **Luster:**
* **Value:**

**Base – High:**

* **Default:**
* **Intensity:**
* **Luma:**
* **Luminance:**
* **Luster:**
* **Value:**

# Verwerking Resolutie

**Light – Base:**

* **Default:**
* **Intensity:**
* **Luma:**
* **Luminance:**
* **Luster:**
* **Value:**

**Base – High:**

* **Default:**
* **Intensity:**
* **Luma:**
* **Luminance:**
* **Luster:**
* **Value:**

# Conclusie Lichtintensiteit

# Conclusie Saturatie

Zoals je duidelijk kan zien, is er in de strijd der algoritmes als het gaat om Saturatie een duidelijke winnaar. Het onderste algoritme genaamd Luma is duidelijk de beste van de gekozen algoritmes. Luma heeft de beste hit ratio van alle algoritmes.

# Conclusie Resolutie

Als we kijken naar de diagrammen van Resolutie, is goed te zien dat Luma de beste optie is van alle vijf gekozen algoritmes. De hit ratio van het Luma algoritme is ook gemiddeld het hoogste van alle vijf gekozen algoritmes.

# Evaluatie

Om de hypothese van dit meetrapport te beantwoorden: Het beste algoritme om de default implementatie te verbeteren is Luma. Luma heeft de hoogste hit ratio van alle geteste algoritmes.

Als je kijkt naar de hypothese die wij gedaan hebben, valt te zien dat we een goede hypothese hebben gedaan. Van de drie hypotheses, zijn er twee waar. Dit komt omdat het Luma algoritme goed zijn in het creëren van geleidelijk kleurverschil. Wij hadden niet verwacht dat Luma ook het beste zal zijn in het saturatiegedeelte van het onderzoek.

Om het verschil in hit ratio te verduidelijken, moet er natuurlijk wel gezegd worden dan de rest van de facial recognition software niet is afgestemd op het grayscale algoritme. Dit kan komen omdat de software ontwikkeld is met een bepaalde implementatie in gedachten, waardoor deze beter werkt of minder goed werkt met bepaalde algoritmen. De resultaten die wij hebben gekregen zijn daardoor ook heel erg implementatie gevoelig.