

Implementatieplan **Intensity**

Jordy Alkema & Rick van Sambeek

5 April 2019



Inleiding

Inleiding	1
Doel	2
Methoden	3
Keuze	5
Implementatie	6
Evaluatie	7
Bronnen	8



Doel

Het doel van deze implementatie is het verbeteren van de RGB naar Greyscale transformatie op basis van resultaat en snelheid, wij hopen met onze implementatie een beter resultaat op te leveren en dit sneller te kunnen doen dan de standaard implementatie.

Methoden

Voor het omzetten van een afbeelding van standaard rgb waardes naar een afbeelding met greyscale waardes zijn meerdere opties, ieder met andere voordelen. Deze opties zijn:

Intensity

RGB converteren naar greyscale op basis van intensity is een van de simpelste algoritme om dit te doen. De waarde van de nieuwe pixel wordt berekend door alle RGB waardes bij elkaar op te tellen en daarna door 3 te delen.

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

Het voordeel van deze methode is dat dit heel simpel te implementeren is en dat dit een relatief snel algoritme is. Het nadeel is dat er geen rekening wordt gehouden met gamma waardoor de helderheid van de RGB waardes.

Gleam

Converteren op basis van gleam is vergelijkbaar met converteren op basis van Intensity. Het verschil is alleen dat bij gleam wel rekening wordt gehouden met gamma. Hierbij wordt de ' gebruikt om de gamma gecorrigeerde waardes aan te geven.

$$\mathcal{G}_{Gleam} = \frac{1}{3}(R' + G' + B').$$

De voordelen van gleam is dat de nieuwe pixels berekend zijn met de juiste helderheid waardoor er dus een realistischer beeld ontstaat. Deze methode is iets langzamer dan de intensity methode omdat eerst alle kleuren een gamma correctie moeten krijgen.

Luminance

Het Luminance algoritme is ontworpen om bij de conversie van RGB naar greyscale rekening te houden met hoe mensen licht waarnemen. De kanalen worden vermenigvuldigd met een factor die overeenkomt met hoe onze ogen reageren op de verschillende kleurkanalen.

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

Het voordeel van dit algoritme is dat kleuren die op elkaar lijken in de greyscale ook op elkaar lijken. Het nadeel is dat bij deze methode de factoren niet rekening houden met gamma, maar in plaats daarvan worden factoren gebruikt die in de buurt komen van de daadwerkelijke perceptie van licht.

Luma

Luma werkt vergelijkbaar met Luminance. Luma maakt ook gebruik van factoren die corrigeren op de perceptie van mensen, maar Luma maakt wel eerst een gamma correctie. Hierdoor zijn de factoren iets anders dan de factoren van Luminance.

$$\mathcal{G}_{Luma} \leftarrow 0.2126R' + 0.7152G' + 0.0722B'.$$

Het voordeel van Luma is dat het beter rekening houdt met de helderheid van de originele kleuren, het nadeel is hierdoor wel dat dit een stuk langzamer wordt dan de andere methoden.

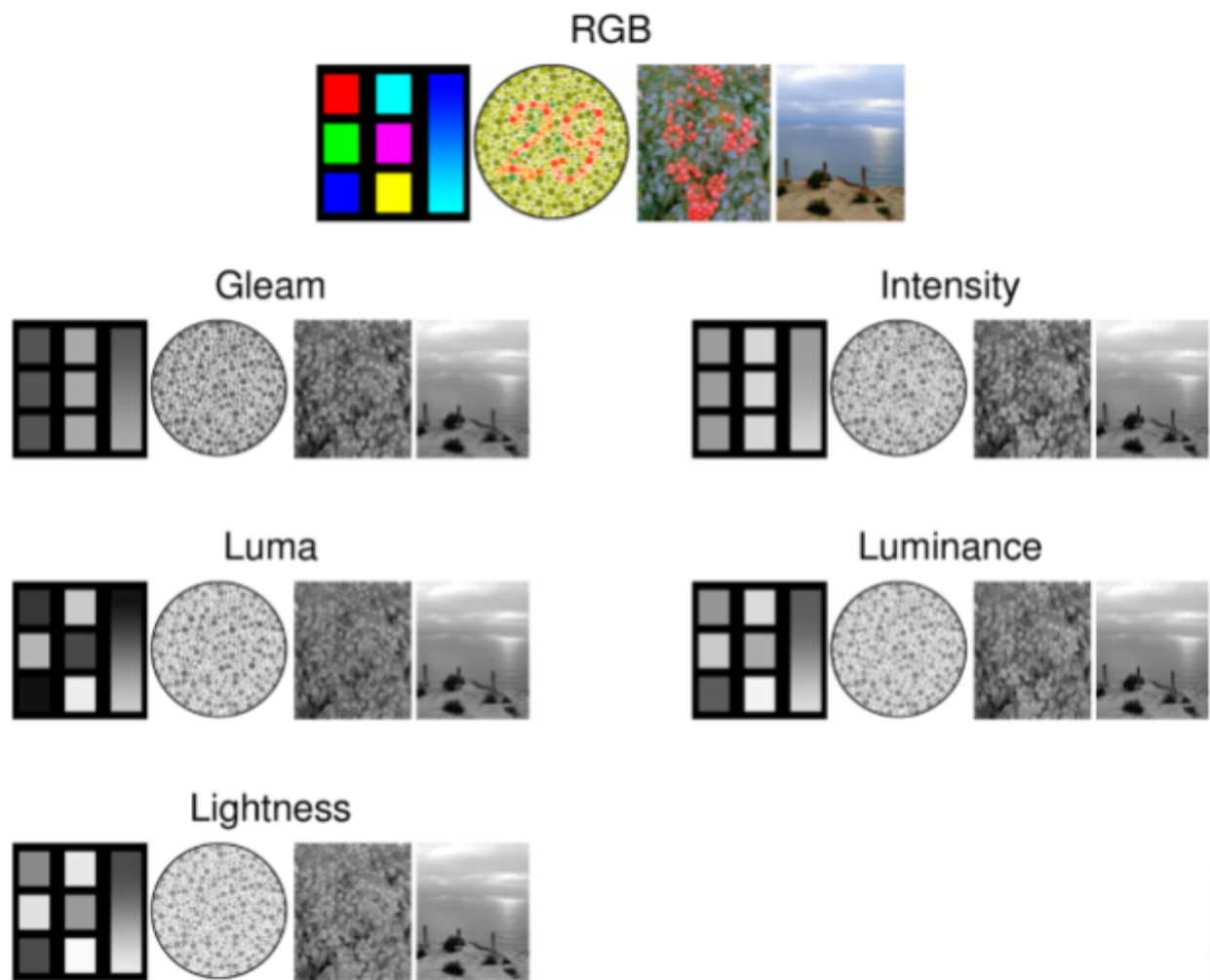
Lightness

Lightness is een uitbreiding op Luma. Bij lightness wordt naast de perceptie van kleuren door mensen en de correctie van gamma ook nog rekening gehouden met helderheid van de kleuren. Dit wordt bereikt door een non lineaire transformatie op het RGB spectrum uit te voeren en daar de nieuwe gevonden grijswaarde doorheen te halen.

$$\mathcal{G}_{Lightness} \leftarrow \frac{1}{100} (116f(Y) - 16),$$

Hierbij geldt dat $Y = 0.2126R + 0.7152G + 0.0722B$ en

$$f(t) = \begin{cases} t^{1/3} & \text{if } t > (6/29)^3 \\ \frac{1}{3} \left(\frac{29}{6}\right)^2 t + \frac{4}{29} & \text{otherwise.} \end{cases}$$



Keuze

Criteria	Weging	Intensity	Gleam	Luminance	Luma	Lightness
Snelheid	3	5	4	4	3	2
Geschatte effectiviteit	5	2	3	4	5	5
Implementatie	4	5	4	4	3	2
Totaal		45	43	48	46	39

Wij hebben gekozen voor Luminance omdat deze methode de beste geschatte effectiviteit en hoogste snelheid heeft in verhouding met hoe moeilijk het is om te implementeren. Verder wordt dit algoritme gebruikt door Gimp (een fotobewerkingsprogramma) en is dit het algoritme wat in MATLAB's wordt gebruikt voor de "rgb2gray" functie.



Implementatie

Voor de implementatie van de onze conversie van RGB naar greyscale moeten wij de classe `PreProcessorStudent` aanpassen. Hierin passen wij de functie `stepToIntensity` aan om gebruik te maken van het luminance algoritme. Dit doen wij voor iedere pixel in de afbeelding.



Evaluatie

Wij gaan voor het testen of wij de goede keuzes hebben gemaakt 2 experimenten doen, wij gaan een poll doen om te kijken wat de mening is van mensen over welke het beste resultaat oplevert, de standaard implementatie of onze implementatie.

Ook gaan wij testen op snelheid van het omzetten van een afbeelding van RGB naar greyscale, dit gaan wij vergelijken met de standaard implementatie.

Wij hopen hieruit een conclusie te krijgen of wij het goed hebben aangepakt.



Bronnen

Color-to-Grayscale: Does the Method Matter in Image Recognition?:

<https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0029740> 5-4-2019