



IMPLEMENTATIEPLAN RGB TO INTENSITY

jasper oosterbroek & Jens Bouman
18-3-2019

Inhoud

Doel	2
Methoden	2
Methode 1.....	2
Methode 2.....	2
Methode 3.....	2
Methode 4.....	2
Methode 5.....	2
Samenvattend	3
Keuze.....	3
Implementatie.....	3
Evaluatie.....	4
Verwijzingen.....	4

Doel

In dit implementatie plan worden keuzes en methoden toegelicht voor het implementeren van een functie voor het converteren van RGB waarden naar intensity waarden. Voor object herkenning zijn kleuren plaatjes overbodig. Data uit 1 kanaal halen is veel sneller dan data uit 3 kanalen halen. Hiervoor is echter een functie nodig om de conversie te doen waarna het plaatje kan worden opgeslagen.

Methoden

Er zijn verschillende manieren om RGB om te zetten naar intensiteit. Hieronder zullen 3 verschillende methoden besproken worden met de voor- en nadelen.

Methode 1

Bij deze methode wordt per pixel alle rgb waarden bij elkaar opgeteld en daarna door 3 gedeeld. Er komt dan een grijswaarde uit die dan voor intensity image kan worden gebruikt. Deze manier is erg snel te schrijven en zal erg efficiënt te werk gaan aangezien er weinig berekeningen worden gedaan. De afbeelding kan door deze manier er minder goed uit zien.

Methode 2

Bij deze methode wordt per pixel alle rgb waarden met een bepaalde weging bij elkaar opgeteld. Deze weging kan worden aangepast waar nodig, maar moet altijd uitkomen op 1. Deze methode is net als methode 1 heel snel en makkelijk te schrijven. Deze methode zorgt voor betere flexibiliteit dan methode 1 en er zijn onderzoeken gedaan naar de wegingen zodat de afbeelding er goed uit zal zien.

Methode 3

Door het gebruik van methode 1 en 2 kan het voorkomen dat voor het oog hele andere kleuren voor de computer het niet heel anders lijkt. Dit kan er voor zorgen dat het moeilijker is om edges te detecteren. Bala & Eschbach nemen ook andere factoren mee in een berekening voor de grijswaarde, dit zorgt ervoor dat de afbeeldingen er beter uitkomen. Door het mee berekenen van deze factoren gebruiken we veel meer geheugen omdat deze waarden bewaart moeten blijven, ook de berekening zelf kost meer tijd om deze uit te voeren.

Methode 4

Een andere methode zou zijn om alleen een enkel kleur kanaal te nemen. Dit kan hele slechte resultaten geven maar hier wordt niks berekent. Dit maakt het wel de snelste methode die er is.

Methode 5

De laatste methode wordt vaak vernoemd als de decomposition methode. Wat er bij deze methode gebeurt is best simpel, we pakken de grootste of kleinste waarden van de drie kanalen als grijs waarden. Dit zorgt voor een heel felle of heel donkere afbeelding wat vaak voor artistieke redenen wordt gebruikt maar niet het beste is voor image recognition.

Samenvattend

Er zijn nu vijf methoden genoemd in het volgende diagram vergelijken wij de methoden.

Bij iedere kolom wordt er een ranking gemaakt waarbij 1 het beste is en 5 het slechtste.

	Snelheid	Implementatie gemak	Verwachte resultaat	Scoren (hoger is slechter)
Methode 1	2	2	3	7
Methode 2	3	3	2	8
Methode 3	5	5	1	11
Methode 4	1	1	5	7
Methode 5	4	4	4	12

Keuze

Als keuze gaan wij voor of methode 1 of zijn gewogen variant methode 2. Er gaan dus 2 methode worden geïmplementeerd om deze met elkaar te vergelijken. De keuze voor deze 2 methode is gedaan omdat beide haalbaar zijn om te implementeren, beide weinig operaties nodig hebben om zijn werk te doen en omdat wij erg nieuwsgierig zijn of het gewicht van methode 2 veel verschil maakt bij de volgende stappen in het programma. Methode 4 had in het samenvattende model weinig punten gescoord, wat iets goeds is, maar wij kiezen ervoor om dit niet te implementeren wegens de slechte resultaten dat het zal geven.

Implementatie

Beide methoden worden in de StudentPreProcessing.cpp gemaakt als de functie `stepToIntensityImage`. Dit wordt dus binnen de bestaande software gedaan er komen verder ook geen nieuwe hulpklassen. Bij de input van de functie nemen wij aan dat dit een correcte `RGBImage` is.

Bij het implementeren van methode worden de volgende waarden al gewichten gebruikt:

Rood : 0.2126

Groen : 0.7152

Blauw : 0.0722

Deze wegingen zijn volgens Poynton een accurate conversie van RGB naar intensiteit.

Bij beide methoden wordt er door de bijgeleverde kleuren afbeelding heen gelopen en wordt iedere pixel opgehaald. Wat er met de pixel wordt gedaan verschilt per methoden. Bij methode 1 worden alle RGB waarden met elkaar opgeteld en daarna door drie gedeeld. Bij methode 2 worden de rgb waarden individueel met de corresponderende waarden vermenigvuldigd. Allebei de methoden resulteren in een intensiteit waarden die dan als nieuwe waarden voor die pixel kan worden gebruikt. Als dit allemaal goed verloopt komen er uit methode 1 en 2 een grijze afbeelding die in de verdere stappen kan worden gebruikt.

Evaluatie

We willen twee methoden gaan vergelijken en kijken of beter kunnen worden gebruikt ten opzichten van de manier die op het moment is geïmplementeerd. Tijdens de vergelijking wordt er op snelheid en kwaliteit ten opzichten van de al geïmplementeerde code gekeken. De snelheid gaat worden getest door een tijdsmeting van de functie te doen, de kwaliteit wordt een vergelijking van debug informatie. Wij verwachten dat de test resultaten van methode 2 erg gaat lijken op de standaard implementatie.

Verwijzingen

- Bala, R., & Eschbach, R. (2004). Spatial Color-to-Grayscale Transform Preserving Chrominance Edge Information. *The Twelfth Color Imaging Conference: Color Science and Engineering Systems, Technologies, Applications* (pp. 82--86). Scottsdale, Arizona: Society for Imaging Science and Technology. Opgehaald van Spatial Color-to-Grayscale Transform: <https://pdfs.semanticscholar.org/ee58/051719d42925562dbbb17d17eaadff19f428.pdf>
- Čadík, M. (2008). Perceptual Evaluation of Color-to-Grayscale Image Conversions. *Computer Graphics Forum*, 1745 - 1754. doi:10.1111/j.1467-8659.2008.01319.x.
- Poynton, C. (2006, 11 28). *Color FAQ - Frequently Asked Questions Color*. Opgehaald van poynton: http://poynton.ca/notes/colour_and_gamma/ColorFAQ.html#RTFToc9