
Implementatieplan voor de conversie van RGB image naar grayscale image.



Florian Humblot - 1720570
Vera Schoonderwoerd - 1721202

Datum: 11 februari 2019

Inhoudsopgave

| | |
|---------------------|---|
| Doel | 3 |
| Methoden | 3 |
| Keuze | 4 |
| Implementatie | 4 |
| Evaluatie | 5 |
| Bronnen | 5 |

Doel

Het doel van de implementatie is om een goede conversie van RGB image naar Grayscale image te bieden.

Methoden

Je geeft hier aan welke methoden er zijn, wat de verschillen tussen de methodes zijn.

- Pixel-based operation
 - o We nemen de gemiddelde intensiteit van de 3 kleuren (RGB) per pixel en tonen deze.
 - Berekening: $Gray = 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B$ (OpenCV, 2015)
 - o Pro:
 - Met elke kleur wordt rekening gehouden.
 - Weight based, per kleur wordt er een gewicht aan gehangen, gezien het feit dat elke kleur een andere invloed heeft op de intensiteit.
 - Bewezen effectief, openCV gebruikt deze ook. (OpenCV, 2015)
- Desaturation
 - o Berekening: $Gray = (Max(Red, Green, Blue) + Min(Red, Green, Blue)) / 2$ (Helland, 2011)
 - o Pro:
 - Oneffenheden worden hierdoor verwijderd.
 - o Con:
 - Weinig contrast blijft over, plaatje wordt relatief donker
- Histogram-based operations
 - o We geven de nieuwe pixelwaarde aan de hand van oude pixelwaarde en histogram.
 - o Con:
 - Onnodig veel werk, gezien andere algoritmes al goed werken
- Neighborhood operation
 - o We geven de pixel een nieuwe waarde, aan de hand van zijn eigen en van zijn omliggende intensiteit.
 - o Pro:
 - Oneffenheden worden hierdoor verwijderd.
 - o Con:
 - Onnodig veel werk, gezien dit niet per se nodig is
- Single color channel
 - o De grayscale wordt bepaald aan de hand van de intensiteit van 1 kleur. Dit kan rood, groen of blauw zijn. De beste kleur is groen, uit onderzoek is gebleken dat mensen hier het sterkst op reageren. (Helland, 2011)
 - o Pro:
 - Eenvoudig te implementeren
 - o Con:

- Kan verschillende uitkomsten geven, aan de hand van hoeveel rood, groen of blauw er is gebruikt in de foto. Als je bijvoorbeeld een foto van de zee converteert via de rood channel, zal er geen mooie conversie optreden.

Keuze

Je geeft een onderbouwing over waarom een bepaalde methode is gekozen, en/of waarom bepaalde settings zijn gebruikt.

Wij gaan de Pixel-based operation gebruiken, gezien dit op verschillende sites aangeprezen wordt als de beste methode. Met deze methode wordt namelijk rekening gehouden met de intensiteit per 3 kleuren, waardoor de conversie per kleur gelijkmatig is.

"Histogram-based operations" heeft een histogram nodig, wat ons onnodig veel werk lijkt. De optie met "desaturation" geeft een relatief donker resultaat, wat de edge detection in de volgende stap kan bemoeilijken. Bij de "single color channel" conversie is het per foto onhandig om één vaste kleur te hebben voor de conversie. Zo kan een foto vrij weinig groen bevatten, maar wordt er wel een conversie met die kleur gedaan. Dit lijkt ons erg onhandig.

Implementatie

Je geeft aan hoe deze keuze is geïmplementeerd in de code

Wij gaan gebruik maken van de functie:

```
IntensityImage * stepToIntensityImage(const RGBImage &image) const;
```

```
(In class StudentPreProcessing)
```

In deze **stepToIntensityImage(const RGBImage &image)** functie wordt er eerst een nieuwe IntensityImageStudent* aangemaakt. Hierna wordt er van de RGBImageStudent de waarden van de RGBpixel bekeken via **getPixel(int i)**. Per pixel wordt de groenwaarde van de RGB de intensity waarde. Deze Graywaarde wordt via **setPixel(int i, Intensity pixel)** gegeven aan de gemaakte IntensityImageStudent. Deze slaat de Intensitywaarde op. Nadat elke pixel is berekend en opgeslagen in IntensityImageStudent, wordt de IntensityImageStudent* gereturned.

Evaluatie

Je geeft aan welke experimenten er gedaan zullen worden om de implementatie te testen en te 'bewijzen' dat de implementatie daadwerkelijk correct werkt. Dit geeft direct informatie over de meetrapporten die er zullen worden gemaakt.

Om te bewijzen dat onze grayscale daadwerkelijk goed gelukt is, willen wij via een enquête mensen ondervragen welke zwart-wit foto beter is, die van ons, of de standaard foto. Wij hopen zo een goed beeld te krijgen van hoe goed onze zwart-wit foto is.

Tenslotte gaan wij een tijdsanalyse doen, om te vergelijken of onze conversie sneller is dan de default conversie.

Bronnen

OpenCV (2015, 18 december). Color conversions. Geraadpleegd op 21 februari 2019, van https://docs.opencv.org/3.1.0/de/d25/imgproc_color_conversions.html

Helland, T. (2011, 1 oktober) Seven grayscale conversion algorithms (with pseudocode and VB6 source code). Geraadpleegd op 24 februari 2019, van <http://www.tannerhelland.com/3643/grayscale-image-algorithm-vb6/>