

1. Verbetering Grayscale Algorithm

1.1. Namen en datum

Baartman, Michel
Bout, Nick
03 April 2018

1.2. Doel

Grayscale in het huidige systeem te verbeteren d.m.v. nieuwe grayscale algoritmes tijdens de pre-processing te gebruiken.

Hiermee proberen wij de snelheid en volledigheid te verbeteren.

Dit document zal een onderzoek bevatten naar de verschillende bestaande grayscale algoritmen en informatie over de onderbouwde keuzes die gemaakt worden.

1.3. Methoden

Het is aan ons om een alternatief te vinden op de grayscale algoritme die huidig in gebruik is, of het huidige algoritme te verbeteren. De algoritmes zijn afkomstig van de blog "Seven grayscale conversion algorithms (with pseudocode and VB6 source code), Tanner Helland(z.d.)[3].

De volgende methoden en algoritmen hebben wij in overweging genomen:

Averaging:

Bij averaging wordt er ge-itereert over elke pixel van een color image en wordt er gekeken naar het gemiddelde van de rood+green+blue waarde.

$$\text{Gray} = (\text{Red} + \text{Green} + \text{Blue}) / 3$$

Luma/Luminance:

Bij deze algoritme wordt de waarde per kleur ook gemiddeld, maar dan wordt er ook een gewicht achter elke kleur gehangen gebaseerd op de menselijke perceptie.

$$\text{Gray} = (\text{Red} * 0.299 + \text{Green} * 0.587 + \text{Blue} * 0.114)$$

Desaturation:

Bij desaturation zal er gekeken worden naar de hoogste en laagste RGB waarde van een selectie van pixels. Van deze waarden wordt het gemiddelde genomen.

$$\text{Gray} = (\text{Max}(\text{Red}, \text{Green}, \text{Blue}) + \text{Min}(\text{Red}, \text{Green}, \text{Blue})) / 2$$

Decomposition:

Bij decomposition wordt er gekeken naar het geheel en wordt er een max en min grayscale waarde aangemaakt. Vervolgens worden de waarden binnen dat kader aangewezen.

$$\text{Gray} = \text{Max}(\text{Red}, \text{Green}, \text{Blue}), \text{Gray} = \text{Min}(\text{Red}, \text{Green}, \text{Blue})$$

1.4. Keuze

Met onze flexibele werkomgeving kunnen wij alle methoden testen en tegenover elkaar zetten. Zo kunnen wij met een meetrapport het beste algoritme boven water halen voor facial recognition en gelijk ook testen wat de eindresultaten zijn in het gehele systeem waarin dit process zich bevind.

Volgens John D. Cook's "Three algorithms for converting color to grayscale"(2009)[2] zal de Luminance methode het beste resultaat overal geven. De wikipedia pagina van Grayscale(z.d.)[1] zegt ook dat de Luminance(het gebruik van de principes van kleur) een "common strategy" is.

1.5. Implementatie

Snelheid

Binnen het huidige meegegeven systeem zullen wij het subsysteem preprocessing aanpassen. Er zal een stuk code, die de algoritme bevat, vervangen worden door de nieuwe algoritmes genoteerd in 1.3.

Zodra de nieuwe algoritme aangeroepen wordt binnen de preprocessing fase zal deze meerdere keren herhaald worden om de snelheid te meten van de nieuw algoritmes.

Volledigheid

De eindproducten grayscale die geproduceerd zullen worden door de algoritmes zullen worden opgeslagen en in kaart gebracht worden voor verdere vergelijking om de volledigheid te meten door observatie en testen.

1.6. Evaluatie

Om te kijken of het nieuwe algoritme het gewenste doel heeft bereikt zullen er verschillende testen worden uitgevoerd:

- Voor de snelheid zal zowel het nieuwe als het huidige meerdere keren uitgevoerd worden. Voor ieder algoritme wordt apart een de gemiddelde duratie gemeten en aan de hand hiervan kan er gekeken worden welk algoritme sneller is; Dit kan gevisualiseerd worden binnen een grafiek en direct vergeleken worden met de resultaten van het vorige algoritme.
- De volledigheid zal worden gemeten door het observeren of de integriteit van de afbeelding behoudt wordt en of het systeem geen errors zal weergeven. De integriteit van een afbeelding is de aanwezigheid van ruis, enige fouten(glitches) gecreëerd door de algoritmes en het behouden van de details(samenkomende schaduwvlakken).

Deze tests worden vervolgens gevisualiseerd in grafieken en tabellen en toegelicht binnen het meetrapport.

1.7. Literatuurlijst

[1] Wikipedia: Grayscale. (z.d.). Geraadpleegd op 6 april 2018, van <https://en.wikipedia.org/wiki/Grayscale>

[2] D Cook, J. O. H. N. (z.d.). Three algorithms for converting color to grayscale. Geraadpleegd op 4 april 2018, van <https://www.johndcook.com/blog/2009/08/24/algorithms-convert-color-grayscale/>

[3] Helland, T. A. N. E. R. (z.d.). Seven grayscale conversion algorithms (with pseudocode and VB6 source code). Geraadpleegd op 4 april 2018, van <http://www.tannerhelland.com/3643/grayscale-image-algorithm-vb6/>