

1. Week 1. 26-04-2015 (Intensity image en RGB image conversion)

1.1. Week1 26-04-2015

Gemaakt door Thijs van Tiem & Jos Roijackers.

1.2. Doel

Het doel van de implementatie is het veranderen van een RGB image naar grijswaarde in Intensity image doormiddel van een functie.

1.3. Methoden

Er zijn meerdere verschillende methodes om RGB naar grijswaarde te converteren. De meest voorkomende zijn:

$$\text{Luminance} = 0.3086 * \text{Red} + 0.6094 * \text{Green} + 0.0820 * \text{Blue}$$

$$\text{Luminance} = 0.299 * \text{Red} + 0.587 * \text{Green} + 0.114 * \text{Blue}$$

Deze formules zorgen ervoor dat de RGB kleuren bij elkaar grijs worden. Door het toepassen van 1 van deze formules is het mogelijk kleuren pixels te converteren naar grijspixels.

1.4. Keuze

De uiteindelijke formule die gebruikt is voor het omzetten is een afgeronde versie geworden van de Luminance = $0.299 * \text{Red} + 0.587 * \text{Green} + 0.114 * \text{Blue}$ formule. Namelijk: $\text{Intensity intensityPixel} = \text{Intensity}(0.2989 * \text{pixel.r} + 0.5870 * \text{pixel.g} + 0.1140 * \text{pixel.b})$;

Er is gekozen voor deze afronding na wat trial en error testing. En hier uit bleek dat deze versie met 4 cijfers achter de komma net iets beter werkte dan een afronding met 3 achter de komma.

1.5. Implementatie

Eerst wordt een dubbele array gevuld met pixels (voor de x en de y as). Hierna wordt bij elke pixel de formule uitgevoerd. ($\text{Intensity intensityPixel} = \text{Intensity}(0.2989 * \text{pixel.r} + 0.5870 * \text{pixel.g} + 0.1140 * \text{pixel.b})$) waarna de nieuwe grijze pixel returned worden in een nieuw plaatje om zo effectief het kleuren beeld te transformeren naar een grijs beeld. De uiteindelijke code ziet er zo uit:

```
IntensityImage* newObject = new IntensityImageStudent(image.getWidth(),
image.getHeight());
    for (int x = 0; x < image.getWidth(); x++){
        for (int y = 0; y < image.getHeight(); y++){
            RGB pixel = image.getPixel(x, y); // wat voor waarde moet dit zijn?
            Intensity intensityPixel = Intensity(0.2989*pixel.r + 0.5870*pixel.g
            + 0.1140*pixel.b);
            newObject->setPixel(x, y, intensityPixel);
        }
    }
    return newObject;
```

1.6. Evaluatie

Om te testen of het daadwerkelijk werkt zoals wij denken gaan we tijdens het implementeren een aantal personal tests doen om te kijken wat er precies gebeurt. Blijkt het uit deze tests dat het allemaal werkt, wordt hierna gekeken of er nog mogelijkheid is voor verbetering. Nadat we een werkende functie hebben zal er vergeleken worden tussen de nieuw gemaakte code en de default code van de leraar om te achterhalen met hoeveel verschil de image wordt geconverteerd.