IMPLEMENTATIE PLAN

*ImageShell & Intensity*



**Christiaan van den Berg & Mike Hilhorst**

1660475 & 1676029

20-02-2019

INLEIDING

In dit verslag zal er besproken worden wat ons implementatieplan is voor de practicumopdracht: Image Shell & Intensity. Deze opdracht bestaat uit twee delen, namelijk:

**Het maken van een Image Shell voor RGB en voor Intensity images**

* Een ImageShell dient als de huls voor een afbeelding, het bevat functies die het mogelijk maken om statische data uit de afbeelding te halen of te wijzigen (denk hierbij over het veranderen van de kleur van een pixel). Voor deze opdracht zal de student een invulling geven aan de bestaande klassen “RGBImageStudent” en “IntensityImageStudent”. De functies die sowieso ‘gemaakt’ moeten worden zijn te vinden in de header-files “IntensityImage.h” en “RGBImage.h”, het is altijd een optie om hier extra functies bij te verzinnen.

**Het maken van een code voor de conversie van RGB naar Intensity images**

* Bij object recognition is het vrijwel altijd overbodig om in het kleurmodel RGB te werken. Het kost meer tijd om data uit drie kanalen te halen (R, G, B) dan uit één kanaal (Grayscale of Intensity).

Om deze rede worden afbeeldingen vaak naar intensity images geconverteerd.

Voor deze opdracht zal de student een functie moeten maken die er voor zorgt dat een “RGBImageStudent” omgezet kan worden naar een “IntensityImageStudent”. Er zijn op het internet meerdere verschillende manier te vinden voor deze omzetting. Beschrijf waarom je welke keuze gemaakt hebt en wat de resultaten zijn in hetmeetrapport!

DOEL

Wij willen de huidige implementatie Image Shell en Intensity verbeteren, dit willen wij doen zonder het gebruik van de bibliotheek genaamd openCV. Wij willen Image Shell en Intensity verbeteren op snelheid en volledigheid.

RANDVOORWAARDEN

Om dit practicum een succes te maken moeten wij twee meet rapporten maken. In deze meet rapporten moeten wij gaan meten en vastleggen wat het verschil is tussen de originele implementatie en onze implementatie. Deze implementaties mogen getest worden op onder andere:

* Snelheid
* Memory efficiency
* Robuustheid
* Volledigheid
* Extra functionaliteiten

Veder moet er elke week moet er een git TAG aangemaakt worden en ingeleverd worden op het Canvas platform. Dit moet tot de laatste week of totdat het project met een voldoende is afgerond.

WERKWIJZE

Wij zullen de huidige implementatie bestuderen, om een beeld te krijgen van hoe de huidige werking functioneert en welke data nodig is om onze eigen implementatie te laten werken. Wij zullen online onderzoek doen naar wat de beste methodes zijn en voor ook waarom deze methodes de beste zijn (voor ons probleem). Verder zullen wij ook vast leggen wat bevindingen zijn tijdens het implementatie proces.

Als wij een werkbare versie hebben dan gaan wij deze testen met veel data. Dit gaan wij doen om een duidelijke en betrouwbare informatie te kunnen verzamelen over onze implantatie. De data die wij willen gaan gebruiken is:

Honderd afbeeldingen met de resolutie:

* 128 bij 128
* 256 bij 256
* 512 bij 512
* 1024 bij 1024

Tien afbeeldingen met veel gebruikte computerscherm resoluties:

* 640 bij 480
* 1280 bij 720
* 1920 bij 1080
* 2160 bij 1440