

# 1. Implementatieplan titel

## 1.1. Namen en datum

Anas Shehata, 13/11/2018

## 1.2. Doel

Het schrijven van een ImageShell voor rgb en voor Intensity images. Het doel van het ImageShell is om het mogelijk te maken om pixelwaardes uit RGB images en Intensity images te kunnen halen en die vervolgens te kunnen opslaan. Dit kan in het vervolg worden gebruikt om wijzigingen te kunnen maken aan het RGB image of Intensity Image.

## 1.3. Methoden

C++ container templates voor de ImageShell:

- Array: "Arrays zijn containers met een vaste grootte: ze bevatten een specifiek aantal elementen dat in een strikte lineaire volgorde is geordend." (CPLUSPLUS, n.d.)
- Vector: "Vectoren zijn reekscontainers die reeksen vertegenwoordigen die in grootte kunnen veranderen." (CPLUSPLUS, n.d.)
- List: "Lijsten zijn sequentiecontainers die constante tijd invoegen en wissen bewerkingen overal in de reeks mogelijk maken, en iteratie in beide richtingen." (CPLUSPLUS, n.d.)

## 1.4. Keuze

Volgens (Joseph, 2019) zijn array zeer efficiënt op basis van snelheid en performance ten opzichte van Vectors. Ook worden er meerdere dimensies ondersteund onder arrays. Bij toegang van waardes hebben Lists langzamer toegang dan arrays vanwege het "lineaire-traversal" van Lists (CPLUSPLUS, n.d.).

Op basis van performance is de keuze gemaakt om een array te gebruiken. Er zal een vergelijking worden gemaakt tussen 1-dimensionale arrays en 2-dimensionale arrays. En vervolgens wordt er een dimensie gekozen die sneller is qua performance.

## 1.5. Implementatie

Voor de implementatie van de RGB ImageShell wordt er in de RGBImageStudent.cpp een array gemaakt ten groote van de breedte \* Lengte van het Image. Vervolgens worden de pixelwaardes in de array toegewezen. De implementatie voor de Intensity Image shell wordt op dezelfde manier gedaan in het IntensityImageStudent.cpp

## 1.6. Evaluatie

De 1-dimensionale array zal vergeleken worden met de 2-dimensionale array op snelheid. Dit wordt gedaan door de RGB to intensity algoritme 2000 maal te gaan laten uitvoeren; 1000maal met een 1-dimensiole array en 1000 maal met een 2-dimensionale array. Tijdens elke uitvoering wordt er een tijd bijgehouden met behulp van (std::chrono::high\_resolution\_clock.). Van de dimensie arrays wordt er een gemiddelde genomen. Deze worden met elkaar vergeleken. De Image die gebruik gaat worden tijdens het testen is Male-3 van het standaard testsets folder.

De reden waarom er voor deze image is gekozen is omdat deze image verschillende kleurwaardes bevat. Er wordt ingeschat dat er 24 uur nodig is voor de implementatie.