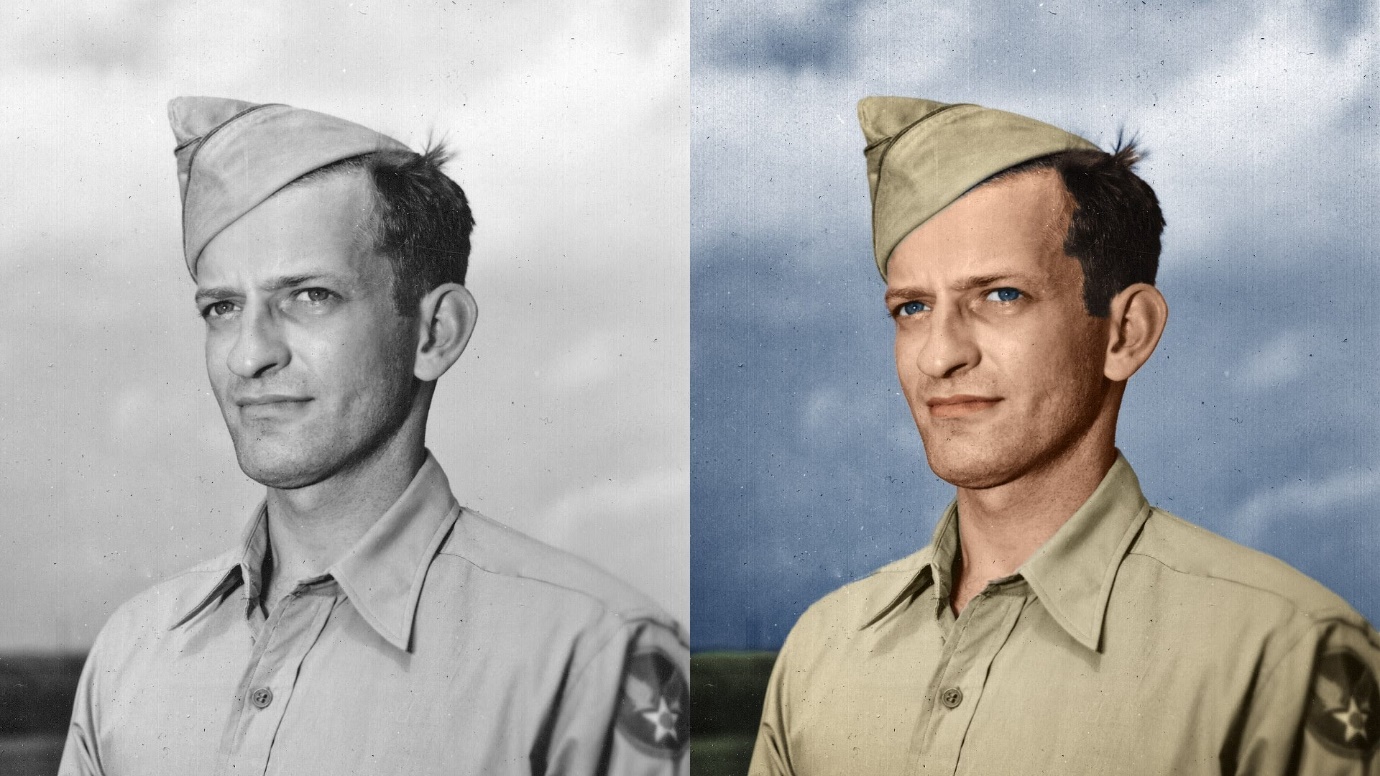
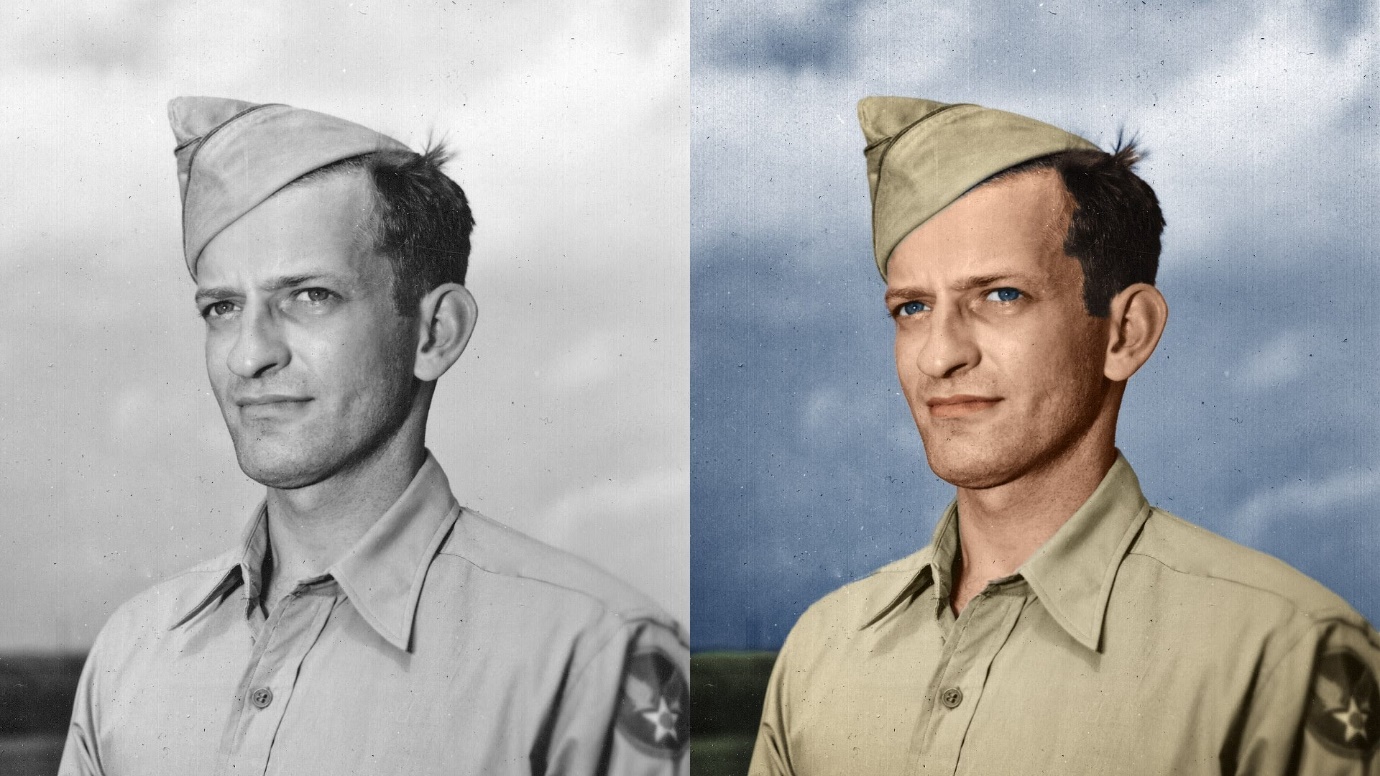
Implementatieplan

*ImageShell en Intensity*





|  |  |
| --- | --- |
| Namen: | Koen de Groot, René de Kluis |
| Klas: | TICT-V2B |
| Docent: | Arno Kamphuis |
| Versie: | V1.1 |

# Doel

Het doel van deze opdracht is dat er een correct werkende conversie van RGB naar Intensity (ook wel Grayscale genoemd) word gemaakt. De uitwerking moet een vergelijkbaar of beter resultaat geven dan de gegeven uitwerking van Arno Kamphuis.

Om dit te kunnen implementeren moet er een ImageShell klasse gemaakt moeten worden.

De ImageShell dient als de huls voor een afbeelding. Deze klasse bevat functies die het mogelijk maken om statische data uit de afbeelding te halen of te wijzigen. Zo kan bijvoorbeeld de kleur van een pixel gewijzigd worden.

Ook zal voor het implementeren de RGB-waarden omgezet moeten worden naar Grijs-waarden. Zodat de gescande foto, wat is kleur is, omgezet kan worden naar een zwart-wit foto.

Er zijn een aantal punten waar er in dit project rekening mee gehouden moet worden:

## Snelheid

De code moet efficiënt werken waardoor de snelheid van de conversie soepel verloopt. Een gebruiker wil namelijk niet al te lang wachten.

## Memory efficiency

Memory efficiency is belangrijk aangezien sommige systemen minder werkgeheugen hebben als anderen. Het is dan ook niet gewenst dat het programma hier veel van in beslag zal nemen tijdens het werken.

## Robuustheid

Het programma moet voornamelijk soft-coded zijn, wat betekend dat er veel structuur in zit en het modulair is. Het is niet gewenst als er bepaalde eigenschappen in de code veranderd moeten kunnen worden door de gebruiker en deze dan in de code bepaalde waarden hiervoor moet aanpassen.

Maar het is ook belangrijk dat de code een goede structuur heeft en er niet te veel code duplicatie is.

## Volledigheid

Het programma moet “volledig” zijn. Dit betekend dat er geen stappen overgeslagen worden of dat er gewenste eigenschappen niet in het programma zitten.

# Methoden

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

Er zijn verscheidene methoden voor het omzetten van een RGB afbeelding naar grijswaarden.

In deze paragraaf zullen deze besproken worden samen met de voor en nadelen.

Maar om deze methoden te kunnen bepalen is het eerst belangrijk om te weten wat kleur is.

Kleur is een eigenschap van licht. Dit is Electro-magnetische straling die uit golven bestaat.

Deze samengestelde golflengtes worden door ons oog omgezet naar kleuren.

Het oog kan drie verschillende kleur intensiteiten opvangen (blauw, groen en rood). Dit word daarna door de hersenen omgezet tot de kleur die te zien is.

De intensiteit van deze kleuren is belangrijk, aangezien hiermee de grijswaarden intensiteiten uitgerekend kunnen worden.

## Omzettingsmethoden

Zoals gezegd zijn er verscheidene methoden beschikbaar om een Grayscale afbeelding te maken uit een RGB afbeelding.

De methoden die in dit document besproken worden zijn de volgende:

* Colorimetrische conversie
* Luminance
* Averaging
* Desaturation
* Decompositie
* Signle color channel

## Colorimetrische conversie

Een colorimetrische conversie werkt met de intensiteit per pixel. De conversie zorgt dat de intensiteit van de omgezette foto hetzelfde is als de intensiteit van de originele.

## Luminance

Deze methode gaat uit van het feit dat mensen niet alle kleuren even goed zien. Mensen zien bijvoorbeeld groen beter dan rood en rood beter dan blauw. Dit wetende als je dus een voor het menselijk oog een beteren grijs waarde wil hebben moet je dus de RGB waardes niet gelijk behandelen.

## Avaraging

Deze methoden telt de RGB waarde van een pixel bij elkaar en deelt dit door 3. De waarde die hier uit komt word dan de waarde van de grijze pixel. Hoewel deze methode snel en makkelijk is heeft het wel problemen met de helderheid van de pixels.

## Desaturation

Deze conversie gebeurd als volgt:

Een RGB (Rood Groen Blauw) waarde word omzet naar een HSL (Hue, Saturation, Lightness) waarde.

Daarna word de verzadiging (saturation) naar de waarde nul getrokken.

Eigenlijk pakt deze manier een kleur en maakt hier de minst verzadigde variant van.

Een berekening die hiervoor gebruikt kan worden is:

Grijswaarde = (max(R,G,B)+min(R,G,B))/2

Na onderzoek blijkt deze manier niet het meest handig te zijn aangezien het resulteert in een foto die eruitziet alsof het met een goedkope zwart-wit camera gemaakt is.

## Decompositie

Decompositie kan op twee manieren gebruikt worden. je kan uitgaan van de max RGB waarden van een pixel of de min RGB waarden. Het maakt bij decompositie niet uit welke kleur de max of min is het kijkt namelijk alleen naar de waarde en zet deze als grijswaarde.

## Single color channel

Met deze conversie kunnen er drie foto’s verkregen worden.

Deze conversie zet namelijk de RGB waarden allemaal op de waarde van rood, groen of blauw.

Deze manier word het meeste gebruikt in digitale camera’s

Het voordeel van deze manier is dat deze het snelst gaat van alle methoden.

Het nadeel is wel dat het moeilijk te voorspellen is hoe het resultaat van de foto eruit komt te zien.

# Keuze

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

# Implementatie

Je geeft aan hoe deze keuze is geïmplementeerd in de code

# 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.

# Bibliografie

N.B. (2017, Februari 2). *Grayscale*. Opgehaald van Wikipedia: https://en.wikipedia.org/wiki/Grayscale

N.B. (2017, Januari 26). *Kleur*. Opgehaald van Wikipedia: https://nl.wikipedia.org/wiki/Kleur