

# Meetrapport

Imageshell en grayscaling over tijd .

Bram van Bergeijk, Leo Jenneskens

2-4-2017

## Doel

Het doel van de metingen is om te onderzoeken wat het gebruik van een ander algoritme voor RGB naar grayscale voor een effect heeft op de snelheid van het programma. Voor de algoritme zullen we single color (enkele color), average(gemiddelde van alle kleuren) en luminosity(verschillende verdelingen van rood, groen en blauw).gebruiken.

## Hypothese

In het onderzoek is de verwachting dat single color het snelst zou zijn aangezien single color de minste berekeningen vereist vergeleken met de andere algoritme.

## Werkwijze.

Voor het maken van de meeting is het als eerst belangrijk om zo min mogelijk achtergrond processen te hebben runnen voor een zo correct mogelijk resultaat. Voor de testen moet als eerst in de main.cpp in de bool executeSteps(DLLExecution \* executor) function executePreProcessingStep1 op true voor de student implementatie en voor de default op false. Voor de verschillende algoritme moet er in de StudentPreProccesing.cpp worden geswitched doormiddel van het commit en uncommit van de vershilldende algoritme. Voor het testen van de tijd hoeft alleen de external dll te runnen als .exe het resultaat wordt weergegeven in de terminal. Na elke meting is het belangrijk om de external dll solution te cleanen voor meest accurate resultaten

De schatting voor het doen van de testen is gemiddeld 2 uur qua testen.

## Resultaat

IN SECONDE	DEFAULT	COLOR.R	COLOR.B	COLOR.G	(COLOR.R+COLOR.G+COLOR.B)/3	0.21 R + 0.72 G + 0.07 B. luminosity
	n.v.t.	single color red	single color blue	single color green	gemiddeld	

VALUE	0.737133	0.855676	0.733936	0.777998	0.730799	0.742846
VALUE	0.76982	0.742455	0.73495	0.759512	0.73727	0.730903
VALUE	0.725824	0.73663	0.714354	0.734647	0.728498	0.736198
VALUE	0.73886	0.72819	0.71343	0.756406	0.726957	0.728552
VALUE	0.75759	0.729563	0.725698	0.748325	0.741794	0.725038
VALUE	0.736136	0.736148	0.656247	0.67267	0.747423	0.731799
VALUE	0.750985	0.745464	0.715656	0.737287	0.679755	0.726906
VALUE	0.769295	0.727707	0.723159	0.746112	0.730594	0.740876
VALUE	0.762521	0.732298	0.725786	0.746626	0.751619	0.750911
VALUE	0.76867	0.717619	0.727259	0.740251	0.739324	0.745208
GEMIDDELDE	0.7516834	0.745175	0.7170475	0.7419834	0.7314033	0.7359237
TEGENOVER DEFAULT( SNELLER)	n.v.t.	0.0065084	0.0346359	0.0097	0.0202801	0.0157597

## Verwerking.

Voor de verwerking van de gemeten gegevens hebben we het gemiddelde genomen van alle 10 de metingen. Dit geeft een relatief correct antwoord op hoe snel het programma runt met bepaalt algoritme. Na de gemiddelde hebben we elke gemiddelde van de default implementatie gemiddelde getrokken om te zien hoeveel sneller een bepaald algoritme is vergeleken met de default implementatie.

## Conclusie.

Uit de verwerking van de gegevens is te halen dat het single color algoritme zoals verwacht het snelst is in het geval van het gebruik van de blauw waarde van RGB.

## Evaluatie.

Zoals uit het onderzoek blijkt is single color het snelst en dat is zoals de hypothese zegt te verwachten door de minimale berekeningen die er voor nodig zijn. Er moet wel rekening gehouden worden dat bij de metingen er achtergrond processen hebben kunnen innervieren met correcte metingen. dit kan effect hebben gehad op de resultaten.