

1. Meetrapport titel

1.1. Namen en datum

Robbin van den Berg & Danny Horvath

1 juni 2015

1.2. Doel

Het doel van dit experiment is om de meest efficiënte functie voor het schalen van een Intensity image te vinden. Er worden 2 functies met elkaar vergeleken.

1.3. Hypothese

Wij verwachten dat het verschil in snelheid minimaal is maar dat de gemiddelde snelheid van de originele functie sneller is dan de eigen gemaakte functie.

1.4. Werkwijze

Er zal in totaal 20x een programma uitgevoerd worden om een Intensity Image te schalen. Eerst 10x met de eigen gemaakte functie, een timer houdt bij wat de snelheid van het gedeelte is van de functie. Vervolgens hetzelfde maar dan met de originele functie. Tot slot wordt van beide functies de gemiddelde uitvoersnelheid berekent en zal blijken welke functie gemiddeld het snelst is.

Eigen functie: `IntensityImage * StudentPreProcessing::stepScaleImage(const IntensityImage &image);`

Originele functie: `IntensityImage * DefaultPreProcessing::stepScaleImage(const IntensityImage &src);`

1.5. Resultaten

#	Snelheid Eigen functie (micro sec)	Snelheid Originele functie (micro sec)
1	9047	13085
2	9142	17271
3	6427	11388
4	6896	15779
5	6769	10720
6	9603	11638
7	9041	10085
8	7017	10487
9	7841	10174
10	8507	12801

1.6. Verwerking

#	Snelheid Eigen functie (micro sec)	Snelheid Originele functie (micro sec)
1	9047	13085
2	9142	17271
3	6427	11388
4	6896	15779
5	6769	10720
6	9603	11638
7	9041	10085
8	7017	10487
9	7841	10174
10	8507	12801
Totaal	80290	123428
Gemiddeld	8029	12342.8

1.7. Conclusie

Uit de verwerking van de meetresultaten kan worden geconcludeerd dat de eigen functie (8029 microseconden) gemiddeld sneller is dan de originele functie (12342.8 microseconden), precies gezegd 4313.8 microseconden sneller.

1.8. Evaluatie

Het doel was om de onderzoeken wat de meest efficiënte functie was voor het schalen van een Intensity Image. Uit de meetresultaten is gebleken dat onze hypothese niet klopte, de originele functie bleek uiteindelijk niet sneller dan onze eigen functie. Doordat er tijdens het meten ook nog andere processen draaide dan alleen het geteste programma, kan het zo zijn dat de omstandigheden waarin getest is niet gelijk zijn geweest, en er als gevolg dus meet onzekerheden kunnen zijn.