

1. Meetrapport ImageShell : snelheid

1.1. Namen en datum

Bas van Eck en Lennart Jensen. 26-3-2019

1.2. Doel

In dit meetrapport wordt onderzocht welke datastructuur het snelste werkt voor het opslaan van afbeeldingen. Er worden 2 methoden vergeleken: gebruik van `std::vector` en het direct alloceren van data op de heap, in plaats van `std::vector` dit te laten afhandelen. Verder wordt de implementatie vergeleken met de default implementatie.

1.3. Hypothese

Er wordt verwacht dat de data direct op de heap te alloceren sneller zal zijn dan het gebruik van `std::vector`, omdat met `std::vector` een extra laag wordt toegevoegd, die ook tijd nodig heeft.

1.4. Werkwijze

Het programma voor computer vision zal eerst licht aangepast worden, om de tijd te kunnen meten die het programma nodig heeft om te runnen. Voor elke implementatie zal het programma 10 keer gerund worden, de tijden zullen worden bijgehouden en hier zal een gemiddelde van gemaakt worden. Deze metingen worden voor alle drie de implementatie uitgevoerd.

1.5. Resultaten

ID	Default	vector	direct heap
1	414	793	564
2	363	783	518
3	371	766	505
4	422	780	510
5	352	788	507
6	423	792	485
7	362	784	495
8	366	785	530
9	355	786	515
10	349	775	506
average	377,7	783,2	513,5

1.6. Verwerking

Als we kijken naar de resultaten, zien we dat de default implementatie het snelst is en de implementatie met `std::vector` het langzaamst. de implementatie met `std::vector` is significant langzamer dan de heap-implementatie. Helaas is de default implementatie nog steeds sneller dan beide implementaties.

1.7. Conclusie

Uit deze gegevens kan worden geconcludeerd dat een datastructuur die direct van de heap gebruik maakt het meest geschikt is voor dit project. De datastructuur met `std::vector` blijkt significant langzamer te zijn dan de heap-implementatie. Om deze reden zullen we besluiten om de heap-implementatie te gebruiken.

1.8. Evaluatie

De conclusie komt overeen met de hypothese. Verder is het onderzoek goed verlopen. De resultaten zijn duidelijk en de verschillen zijn significant.