# 3 Tijdverschil tussen zero en first-order

## 3.1 Namen en datum

Jeroen Huisen & Hendrik Cornelisse

06-04-2014

Week 6

## 3.2 Doel

Het doel van dit experiment is te kijken hoeveel procent tijds verschil er zit tussen zero en first-order op een 1080p afbeelding en een 4K afbeelding. Dus de onderzoeksvraag is: hoeveel procent doet first-order er langer over dan zero-order bij 1080 en 4K afbeeldingen?

## 3.3 Hypothese

We verwachten dat bij een afbeelding van 1080p de tijd van first-order ongeveer 75% meer tijd nodig heeft dan zero-order. Dit komt vanwege dat er 4 pixels opgevraagd moeten worden in plaats van 1 en dan nog een berekening gedaan moet worden welke kleur de pixel moet worden. Bij de 4K afbeelding verwachten we ongeveer de zelfde uitslag als bij de 1080 afbeelding, dus dat first-order ongeveer 75% meer tijd nodig heeft. We gaan er van uit dat hier ongeveer 5 % vanaf geweken mag worden vanwege dat het systeem tussen door ook andere dingen kan doen.

## 3.4 Werkwijze

Hoe we het experiment uitgevoerd hebben is de 4K afbeelding 5 keer met first-order laten draaien en 5 keer met zero-order laten draaien. Dit geld ook voor de 1080p. Deze hebben we ook 5 keer met first-order en 5 keer met zero-order laten draaien. Na de 5 keer laten draaien bereken we het gemiddelde van de tijd dat de first en zero-order. Van de 2 gemiddelde bereken we het percentage dat first-order langer duurt dan zero-order.

We meten de tijd met de Basetimer library. De timer wordt gestart wanneer first of zero-order gestart worden. De timer wordt stop gezet als de afbeelding opgeslagen is en daarna wordt de tijd op het console afgedrukt.

Het systeem waar we dit op getest hebben bevat de volgende systeemeigenschappen:

* Windows 7 Professional 64-bit
* CPU Intel Core I7-2670QM op 2,2 GHz met turbo naar 3,1 Ghz
* 6 GB ram op 1333 MHz

Verder waren op het systeem alle onnodige programma’s tijdens het testen afgesloten.

## 3.5 Resultaten

In deze tabel staan de tijden van een 4K (4096 \* 2304) afbeelding

|  |  |
| --- | --- |
| First-order | Zero-order |
| 12.16 | 7.29 |
| 12.53 | 7.26 |
| 12.29 | 7.29 |
| 12.18 | 7.38 |
| 12.52 | 7.27 |
| **Gem = 12.336** | **Gem = 7.298** |

In de volgende tabel staan de waarde van een 1080P (1920 \* 1080) afbeelding

|  |  |
| --- | --- |
| First-order | Zero-order |
| 3.01 | 1.76 |
| 3.17 | 1.72 |
| 2.92 | 1.74 |
| 2.89 | 1.68 |
| 2.91 | 1.79 |
| **Gem = 2.98** | **Gem = 1.738** |

## 3.6 Verwerking

Nu de resultaten hebben moeten we ze nog omrekenen naar percentages.

Het percentage dat first-order er langer over doet is bij een 4K afbeelding is:

(12.336 / 7.298 – 1) \* 100% = 69.0 %

Dus met een 4K afbeelding duurt First-order 69.0 % langer

Het percentage dat first-order er langer over doet bij een 1080P afbeelding is:

(2.89 / 1.738 – 1) \* 100% = 66.3%

Dus met een 1080P afbeelding duurt First-order 66.3% langer

Wat we verder nog van de resultaten kunnen vertellen:

* Het systeem was stabiel bij de het uitvoeren van de zero en first-order, want er zijn geen grote pieken of dalen te zien als uitkomst waardes.

## 3.7 Conclusie

De conclusie die we hebben kunnen halen uit de resultaten is dat er bij een 1080P afbeelding first-order er gemiddeld 66.3 % langer overdoet dan zero-order. Bij een afbeeldingen van 4K doet first-order er gemiddeld 69.0% langer over. Als we de 2 percentages vergelijken met de hoeveelheid pixels er bij dan kunnen we concluderen dat hoe meer pixels er zijn hoe groter het verschil tussen zero en first-order gaat worden.

## 3.8 Evaluatie

Als we de conclusie en de hypothese gaan vergelijken komen we tot de conclusie dat de geschatte tijd ongeveer tussen de 10 % en 5 % lager kan. Het kost dus minder tijd om pixels op te vragen dan verwacht. Verder is opvallend dat we bij de 4K afbeelding er dichter bij de werkelijke percentage zitten dan bij 1080P. dus onze theorie zal uiteindelijk bij een bepaalde resolutie wel kloppen, maar dat is bij de geteste resoluties niet het geval.