

1. Face recognition

1.1. Bianca Krieger, Marianne Delmaar 1-6-2015

1.2. Doel

Controleren hoe snel de student implementatie is.

1.3. Hypothese

De student implementatie is efficiënter dan de default implementatie.

1.4. Werkwijze

Er zal een timer worden gestart. De code zal een x aantal keer worden uitgevoerd. Daarna zal de timer worden gestopt. Er zal een gemiddelde tijd worden berekend door de tijd die verstreken is door de code uit te voeren te delen door het x aantal keer dat de code is uitgevoerd. Dit zal over meerdere afbeeldingen worden getest.

1.5. Resultaten

Afbeelding	Tijd Student impl.	Default impl.
Child-1.png	1314,9us 1298,9us 1300,2us	73194,1us 52917,5us 56051,4us
Female-1.png	1007,0us 1007,0us 1003,6us	62703,3us 54523,7us 45230,4us
Female-2.png	1306,8us 850,0us 735,2us	28839,2us 18839,2us 27853,4us

1.6. Verwerking

De functie wordt 10x uitgevoerd. Daarna zal de tijd worden vastgesteld hoe lang de functie er over gedaan heeft om 10x uitgevoerd te moeten worden. Daarna wordt de tijd gedeeld door 10 om een gemiddelde te krijgen.

1.7. Conclusie

Zoals verwacht is de Student implementatie veel efficiënter dan de Default implementatie.

1.8. Evaluatie

Hoewel de meetresultaten duidelijk aantonen dat de studenten implementatie sneller is, moet wel gezegd worden dat de standaard implementatie een net iets andere uitkomst heeft. Het zou daarom kunnen worden beargumenteerd dat de extra tijd die de standaardimplementatie nodig heeft voor de berekeningen ook daadwerkelijk nodig zijn om een zo kloppend mogelijke uitkomst te genereren. Echter komt onze implementatie goed genoeg in de buurt en hebben wij bewust de nadruk gelegd op een zo hoog mogelijke performance, wat dus gelukt is.