

# Measurement analysis of speed across multiple methods of grey-scaling

Niels de Waal (1698041), Jasper Smienk(1700502)

April 10, 2018

# Contents

<b>1</b>	<b>Target</b>	<b>3</b>
<b>2</b>	<b>Hypothesis</b>	<b>3</b>
<b>3</b>	<b>Method</b>	<b>3</b>
<b>4</b>	<b>Results</b>	<b>4</b>
4.1	Raw data . . . . .	4
4.2	Visualized . . . . .	4
<b>5</b>	<b>Processing</b>	<b>5</b>
<b>6</b>	<b>Conclusion</b>	<b>5</b>
<b>7</b>	<b>Evaluation</b>	<b>5</b>
7.1	What went well . . . . .	5
7.2	What went wrong . . . . .	5
7.3	What could be done differently . . . . .	5

## 1 Target

With this measurement analysis we want to find out how fast each grey-scaling method is and compare them to each other and the default implementation.

These results can help improve the speed of the facial recognition because converting an image to grey-scale is one of the steps that need to be done.

## 2 Hypothesis

We suspect that of our methods the *decomposition* will be the fastest, as it required very little computation. Followed by *averaging* and lastly *luma*.

We can't say anything about how they will perform against the default implementation, as we don't know how it works.

## 3 Method

For each grey-scaling method, we will run it 10000 times and see how long it took from the start to the end. From this we will calculate the average time it took.

The test will be run twice for each method, once with the facial recognition, and once without.

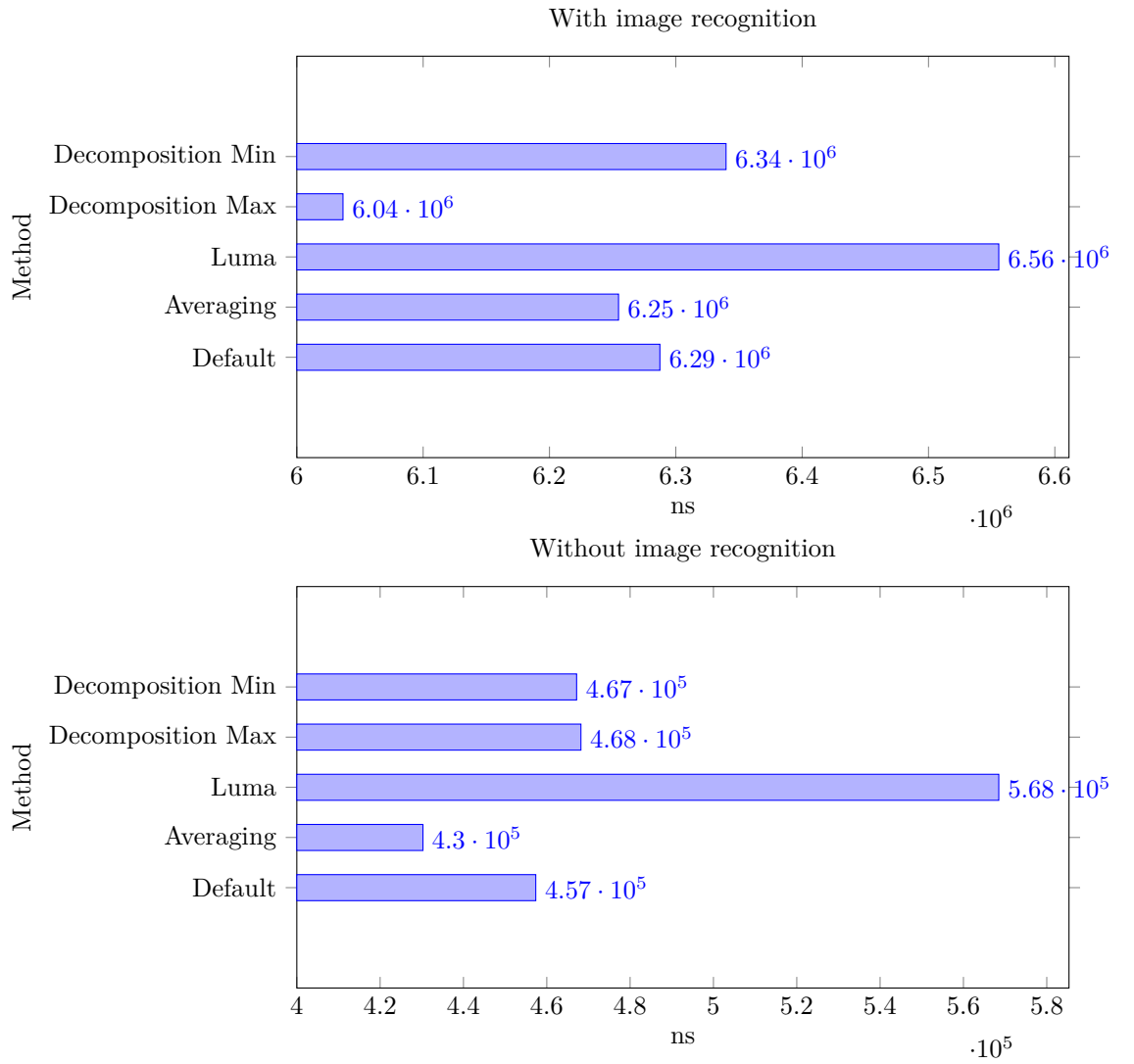
We will make sure the tests are run on the same laptop and keep an eye out on the temperature to make sure it does not thermal-throttle.

## 4 Results

### 4.1 Raw data

Method	Speed total (ns)	Speed greyscale only (ns)
Default	6287400	457354
Averaging	6254653	430275
Luma	6555633	568484
Decomposition (Max)	6036561	468192
Decomposition (Min)	6339750	467159

### 4.2 Visualized



## 5 Processing

When looking solely at which method is the fastest grey-scale method, averaging comes out ahead. Followed by the default method and both decomposition methods. At last is Luma, which was expected.

However, decomposition max comes out ahead when you also consider the computation time of the facial recognition that follows. Then comes averaging, the default and decomposition min. Luma is last yet again.

## 6 Conclusion

Depending on what you need, different methods are applicable. If you just want the fastest method, use averaging. However, if you want to use it with facial recognition, use decomposition max.

An other observation is that the facial recognition with the Luma method is noticeable slower, even though the grey-scaling is a small part of the whole process. This means that even though Luma makes the (subjectively) best looking images for the human eye, they make the facial recognition slower.

## 7 Evaluation

### 7.1 What went well

The project was a very cooperative experience for both parties involved. This cooperation went very well, everyone had their tasks and did them with equal amount of dedication. This made the project an overall pleasant experience.

### 7.2 What went wrong

We both had a vigorous start with the project, however it turned out to be very hard to keep up this speed. Eventually we started to have to divide our time to other projects that required our attention at the time. This made it so we had to do a lot of work near the deadline.

Also the original plan was to use OpenCL and SIMD for parallelism. Because of the state of the relevant tools, this turned out to be quite a bit harder than was originally thought.

### 7.3 What could be done differently

A better research on the more ambitious subjects should be done in order to avoid unnecessary delays.

## References