

CS3211 project 1 - Image processing: implementation and analysis

Tan Soon Jin

February 13, 2017

Contents

1	TODO Part 1	2
1.1	Hardware	2
1.1.1	Find out hardwares used in lab and Tembusu clusters	2
1.1.2	How the hardware contributes to the result	2
1.2	Speedup	2
1.2.1	Tabulate result from Lab01 Task 3 & 4	2
1.2.2	Interpret the result based on	2
1.3	Memory Effects	4
1.3.1	Tabulate result from Lab01 Task 6	4
1.3.2	What is causing the behaviour ?	4
1.4	Accuracy	4
1.4.1	Tabulate result from Lab02 Task 1-4	4
1.4.2	What is causing the behaviour ?	4
1.5	Communication. speedup	4
1.5.1	Tabulate result from Lab02 Task 7	4
1.5.2	What is causing the behaviour ?	4
1.5.3	How does openMPI implementation scale when run on multiple processors ?	4
2	TODO Part 2 Demo	4
2.1	Develop as many image processing functions implemented on GPU.js.	4
2.1.1	Input: animation or movie file	4
2.2	Users should be able to see CPU vs GPU comparisons clearly for each functions	4
2.3	Focus on parallelized components	4

2.4	Getting started with GPU.js	4
2.5	Criteria to be judged	4
2.5.1	Anything special for accuracy ? speedup ? workload reduction ?	4
2.5.2	Anything to improve the system ?	4
2.5.3	Adjusting parameters to specify different methods of operation	4
2.5.4	Implementing camera motion/fly-through technique	4
2.5.5	Implement other animations and background not pro- vided	4
2.5.6	Provide evidence of testing	4
2.6	Tabulate result and discuss	4
3	Output	4
3.1	A0112213.zip	4
3.2	Hardcopy	4

1 TODO Part 1

1.1 Hardware

1.1.1 Find out hardwares used in lab and Tembusu clusters

1.1.2 How the hardware contributes to the result

1.2 Speedup

1.2.1 Tabulate result from Lab01 Task 3 & 4

1.2.2 Interpret the result based on

1. Fixed-time
2. Fixed-size

1.3 Memory Effects

1.3.1 Tabulate result from Lab01 Task 6

1.3.2 What is causing the behaviour ?

1.4 Accuracy

1.4.1 Tabulate result from Lab02 Task 1-4

1.4.2 What is causing the behaviour ?

1.5 Communication. speedup

1.5.1 Tabulate result from Lab02 Task 7

1.5.2 What is causing the behaviour ?

1.5.3 How does openMPI implementation scale when run on multiple processors ?

2 TODO Part 2 Demo

2.1 Develop as many image processing functions implemented on GPU.js.

2.1.1 Input: animation or movie file

2.2 Users should be able to see CPU vs GPU comparisons clearly for each functions

2.3 Focus on parallelized components

2.4 Getting started with GPU.js

2.5 Criterias to be judged

2.5.1 Anything special for accuracy ? speedup ? workload reduction ?

2.5.2 Anything to improve the system ?

2.5.3 Adjusting parameters to specify different methods of operation

2.5.4 Implementing camera motion/fly-through technique

2.5.5 Implement other animations and background not provided

2.5.6 Provide evidence of testing

2.6 Tabulate result and discuss

3 Output

3.1 A0112213.zip

3.2 Hardcopy