COM4521 - Parallel Computing with GPUs

Assignment Marking and Feedback - 2022

Student Name: Pramod Raghunath Potdar
Username: lip21prp

Total Mark: 93.30%

Note: Any lateness penalty will be applied by the teaching admin team and is not considered when marking.

Marking Breakdown and Feedback

OpenMP - Stage 1	Score / 100	Stage Weight (%)	Weighted Score
	100	10.00%	10.00%

Feedback:

Has the stage been implemented?

The implementation of OpenMP stage 1 is complete and is free from any obvious defect. The code is free from race conditions and does not introduce unnecessary OpenMP features or functionality. The code is structured well and easy to follow. It has appropriate use of comments and can be easily interpreted. The implementation has excellent performance.

Is the implementation correct for the stage when compared to a number of test cases?

Test Case 1, two images provided with the handout, passed the validation for both images. Test Case 2, new image using a size provided in the handout code, passed the validation. Test Case 3, new image using a size not provided in the handout code, passed the validation. Test Case 4, new image using portrait aspect ratio, passed the validation. Test Case 5, new image using landscape aspect ratio, passed the validation.

Choice, justification and performance reporting of the approach towards implementation as evidenced in the report

The description of the approach is excellent. An optimal approach was described for the reductions. Excellent justification was provided for the scheduling approach, choice of loop parallelisation and the tile reductions. The performance is good and measured on multiple images. The performance section correctly identifies limiting factors of performance referring to memory and cache usage. The performance section provides good justification of limiting factors through profiling, benchmarking or theoretical assessment.

Any Additional Comments

OpenMP - Stage 2	Score / 100	Stage Weight (%)	Weighted Score
	100	10.00%	10.00%

Feedback:

Has the stage been implemented?

The implementation of OpenMP stage 2 is complete and is free from any obvious defect. The code is free from race conditions and does not introduce unnecessary OpenMP features or functionality. The code is structured well and easy to follow. It has appropriate use of comments and can be easily interpreted. The implementation has excellent performance.

Is the implementation correct for the stage when compared to a number of test cases?

Test Case 1, two images provided with the handout, passed the validation for both images. Test Case 2, new image using a size provided in the handout code, passed the validation. Test Case 3, new image using a size not provided in the handout code, passed the validation. Test Case 4, new image using portrait aspect ratio, passed the validation. Test Case 5, new image using landscape aspect ratio, passed the validation.

Choice, justification and performance reporting of the approach towards implementation as evidenced in the report

The description of the approach is excellent. An optimal approach was described using simple OpenMP parallelisation Good justification was provided for both the optimal (static) scheduling approach and the choice of loops to parallelise. The performance is good and measured on multiple images. The performance section correctly identifies limiting factors of performance. The performance section provides good justification of limiting factors through profiling, benchmarking or theoretical assessment.

Any Additional Comments

Making a decent case for not optimising is fine. Stage 2 is less than 1% of the runtime, it's not worthy of much effort.

OpenMP - Stage 3	Score / 100	Stage Weight (%)	Weighted Score
	100	10.00%	10.00%

Feedback:

Has the stage been implemented?

The implementation of OpenMP stage 3 is complete and is free from any obvious defect. The code is free from race conditions and does not introduce unnecessary OpenMP features or functionality. The code is structured well and easy to follow. It has appropriate use of comments and can be easily interpreted. The implementation has excellent performance.

Is the implementation correct for the stage when compared to a number of test cases?

Test Case 1, two images provided with the handout, passed the validation for both images. Test Case 2, new image using a size provided in the handout code, passed the validation. Test Case 3, new image using a size not provided in the handout code, passed the validation. Test Case 4, new image using portrait aspect ratio, passed the validation. Test Case 5, new image using landscape aspect ratio, passed the validation.

Choice, justification and performance reporting of the approach towards implementation as evidenced in the report

The description of the approach is excellent. An optimal approach was described using simple OpenMP parallelisation Good justification was provided for both the optimal (static) scheduling approach and the choice of loops to parallelise. The performance is good and measured on multiple images. The performance section correctly identifies limiting factors of performance. The performance section provides good justification of limiting factors through profiling, benchmarking or theoretical assessment.

Any Additional Comments

CUDA - Stage 1	Score / 100	Stage Weight (%)	Weighted Score
	97	30.00%	29.10%

Feedback:

Has the stage been implemented?

The implementation of CUDA stage 1 is complete and is free from any obvious defect. Appropriate use of synchronisation and warp/block techniques to avoid race conditions. The code is structured well and easy to follow. It has appropriate use of comments and can be easily interpreted. Appropriate technique implemeneted for the tile reduction.

Is the implementation correct for the stage when compared to a number of test cases?

Test Case 1, two images provided with the handout, passed the validation for both images. Test Case 2, new image using a size provided in the handout code, passed the validation. Test Case 3, new image using a size not provided in the handout code, passed the validation. Test Case 4, new image using portrait aspect ratio, passed the validation. Test Case 5, new image using landscape aspect ratio, passed the validation.

Choice, justification and performance reporting of the approach towards implementation as evidenced in the report

The description of the approach is excellent. Excellent justification was provided for the approaches used to reduce each tile. The performance is good and measured on multiple images. The performance section includes profiler use to identify limiting factors. The performance section provides some justification of limiting factors through profiling, benchmarking or theoretical assessment.

Any Additional Comments

CUDA - Stage 2	Score / 100	Stage Weight (%)	Weighted Score
	94	20.00%	18.80%

Feedback:

Has the stage been implemented?

The implementation of CUDA stage 2 is complete and is free from any obvious defect. Appropriate use of synchronisation and warp/block techniques to avoid race conditions. The code is structured well and easy to follow. It has appropriate use of comments and can be easily interpreted. Appropriate technique implemeneted for calculation the global image average

Is the implementation correct for the stage when compared to a number of test cases?

Test Case 1, two images provided with the handout, passed the validation for both images. Test Case 2, new image using a size provided in the handout code, passed the validation. Test Case 3, new image using a size not provided in the handout code, passed the validation. Test Case 4, new image using portrait aspect ratio, passed the validation. Test Case 5, new image using landscape aspect ratio, passed the validation.

Choice, justification and performance reporting of the approach towards implementation as evidenced in the report

The description of the approach is excellent. Identified that a single kernel launch with block and global level reductions should be used. Good justification was provided for some of the approaches used to reduce global image aveage. The performance is good and measured on multiple images. The performance section includes profiler use to identify limiting factors. The performance section provides some justification of limiting factors through profiling, benchmarking or theoretical assessment.

Any Additional Comments

CUDA - Stage 3	Score / 100	Stage Weight (%)	Weighted Score
	77	20.00%	15.40%

Feedback:

Has the stage been implemented?

The implementation of CUDA stage 3 is complete and is free from any obvious defect. The stage has been implemented without introducing any race conditions. The code is structured well and easy to follow. It has appropriate use of comments and can be easily interpreted. The implementation uses appropriate methods for accessing memory.

Is the implementation correct for the stage when compared to a number of test cases?

Test Case 1, two images provided with the handout, passed the validation for both images. Test Case 2, new image using a size provided in the handout code, passed the validation. Test Case 3, new image using a size not provided in the handout code, passed the validation. Test Case 4, new image using portrait aspect ratio, passed the validation. Test Case 5, new image using landscape aspect ratio, passed the validation.

Choice, justification and performance reporting of the approach towards implementation as evidenced in the report

The description of the approach is excellent. Advanced methods for caching data accesses were not considered. Some limited justification was provided for the approach to broadcast the mosaic to the final image. The performance is good and measured on multiple images. The performance section includes profiler use to identify limiting factors. The performance section does not include any justification of limiting factors.

Any Additional Comments

Overall Summary

What was good about this assignment?

Your OpenMP solution performed among the fastest on the problem/hardware used for benchmarking (498ms vs class average 1070ms). I'm susprised static default scheduling was so effective, other high performers prefered dynamic (especially for stage 3), although static should theoretically suffice (assuming balanced hardware cores). Benchmarked CUDA perf 425ms, vs class best 401ms, class average 628ms.

What Could be improved about this assignment?

Would have been nice to see greater detail regarding your scheduling investigations, but hard to fault your OpenMP work.