written-report-ammar-150454388

January 25, 2019

Table of Contents

0.1 Introduction

This report covers the Cloud Computing data mining project on Terapixel Processing Graphic Processing Units (GPUs) bottlenecking by summarising the project's background, objectives, methods and findings. Moreover, this report will also reflect on the project processes and tools. This report however is a summary of the CRISP DM documents produced by this project, hence detailed findings will not be covered here.

0.2 Project Background and Justification

Terapixel and high resolution imaging reconstruction and tiling is becoming a critical tool in many fields. However the high resolution nature of terapixel processing makes it overwhelming in size and difficult to process. Hence, Terapixel processing requires powerful high performance computational resources (like high end GPUs) that are typically offered over cloud servies for scalability. Hence, improving GPU performance of terapixel processing solutions that are delivered over the cloud is important, to help improve terapixel complex processing.

0.3 Project Criteria/Objectives

This project aim is to analyse the bottlenecks and trends of the usage of graphical processing units (GPUs) deployed for terapixel rendering jobs of Newcastle upon Tyne over a cloud computing platform (Microsoft Azure). The specific criteria that this project needs to complete went as below:

- Analyse how well the cloud application parallelises the workload
- Measure and analyse memory usage
- Measure how well the GPUs respond to varying data scales and loads

0.4 Tools, Methodology and Evaluation Design

The project will follow the CRISP-DM hierarchical cyclic process model for data mining (Chapman et al., 2000) to provide the process for data mining in this project. The project and make use of Test Driven Development (PyTests) and literate programing (Jupyter Notebook), which are used to ensure that both tasks of testing documentation can be carried simultaneously with the coding process. Git and Github is used for version control to manage the progress of cookie cutter based project template, the code in that project will also be documented with the help of Sphinx.

The project will be divided into these stages according to the produced CRISP DM based reports:

1. Business Understanding:
    - This report covers the background, justification, objectives and project plan for the data mining project.
2. Data Preparation:
    - This report covers how the data was cleaned and prepared.
3. Data Understanding and Evaluation:
    - This report covers the data exploratory process undertaken in this project, and its evaluation and the evaluation of the project in general.

0.5 Exploratory Findings and Results

The exploratory findings have shown that the rendering task is one of the most important and time consuming tasks in the terapixel rendering process, as it topped the execution times and used the GPUs the most. The project has found that while the GPU itself was well utilised, the memory of the GPU of itself might have not been as well utilised showing lower usages even during rendering, perhaps causing a bottleneck. Moreover, it seems that there was some evidence that some cards seem to have different tolerances to voltage and temperature, as they performed significantly worse when power draw and temperature increased than their peers appearing as outliers, perhaps causing slowdown.

Other interesting results include how it seemed that overall the readings for temperatures and power draw were quite impressive when compared to some modern GPUs showing low readings (averages of <100W and 40C!), which suits the long run time based cloud environment. And, how the scheduling of tasks, while on average slightly balanced, seemed to have made a couple cards that carry significantly less tasks than others.

0.6 Evaluation and Future work

The project seemed to have met most of its criteria and provided a lot interesting outlooks on bottlenecks, however some issues were recognised with the use the lack of exact memory measurements and the lack of variation analysis on a task per task basis (instead relying on just the averages). Hence, further memory analysis with more memory statistics, analysis of tiling scaling and with in task variation analysis might be considered in future projects.

When it came to the techniques and tools used in this project, while the newer cookie cutter python based environment provided useful tools with more intricate controls (e.g. VirtualEnv for isolation), it proved to be a significant hurdle in the early stages of the project. The newer

tools seemed to not have as many conveniences like some R based tools like Project Template (e.g. automatic dataset loading). Nonetheless, the experiences with some of the tool - in particular Sphinx documents and virtual enviroments - will prove useful in future projects.