Retrospective Sprint 2
of
Group BLOCKS7PG

Block Model Compression Algorithm

a1162576 Karl Asenstorfer

a1806297 Po-Yi Lee

a1804817 Xiaoman Li

a1784375 Yuanpeng Liu

a1782685 Yang Lu

a1797683 Jiaping Qi

a1786785 Hechen Wang

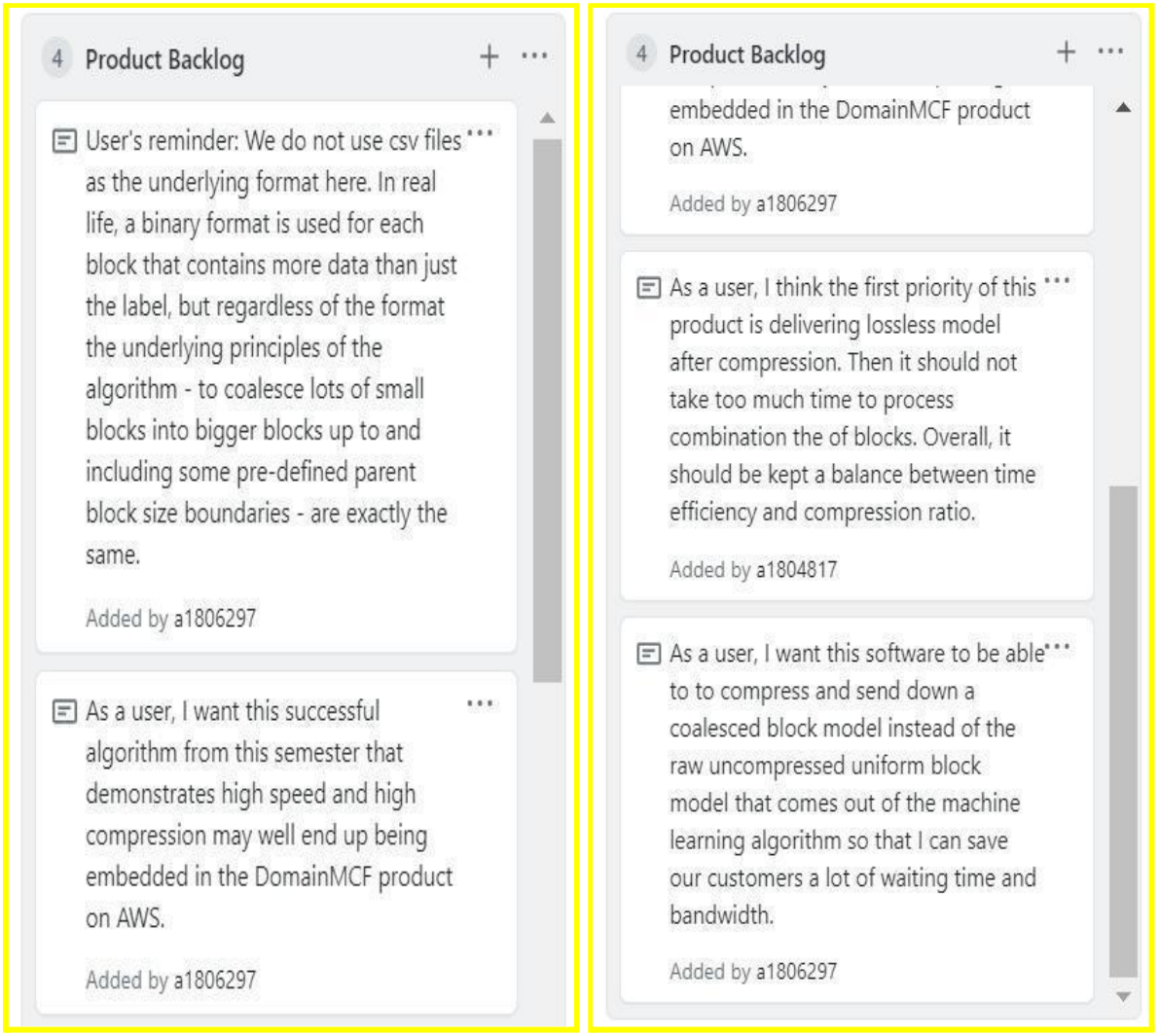
a1784184 Kaiyang Xue

a1811518 Liuyang Yun

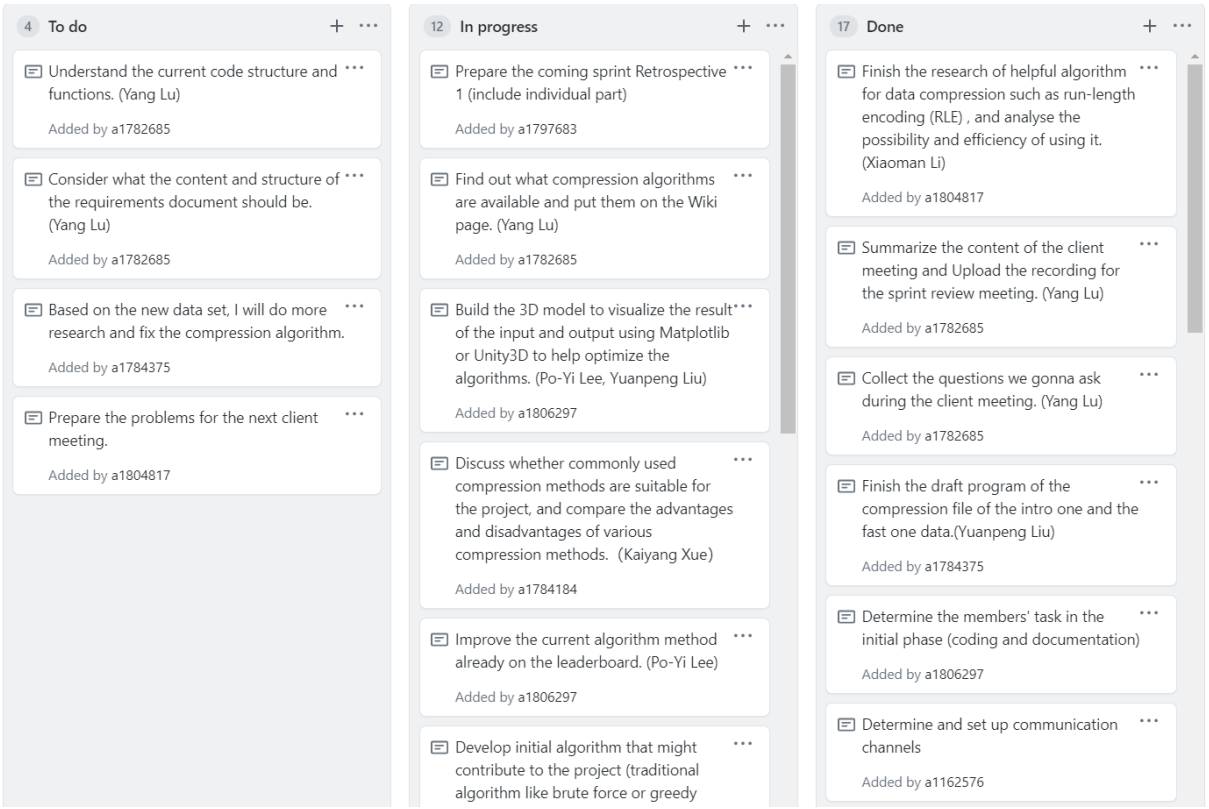
Snapshot Week 5 of Group BLOCKS7PG

Product Backlog and Task Board

Product Backlog:



Task Board:


4 To do
Understand the current code structure and functions. (Yang Lu)
Added by a1782685
Consider what the content and structure of the requirements document should be. (Yang Lu)
Added by a1782685
Based on the new data set, I will do more research and fix the compression algorithm.
Added by a1784375
Prepare the problems for the next client meeting.
Added by a1804817
12 In progress
Prepare the coming sprint Retrospective 1 (include individual part)
Added by a1797683
Find out what compression algorithms are available and put them on the Wiki page. (Yang Lu)
Added by a1782685
Build the 3D model to visualize the result of the input and output using Matplotlib or Unity3D to help optimize the algorithms. (Po-Yi Lee, Yuanpeng Liu)
Added by a1806297
Discuss whether commonly used compression methods are suitable for the project, and compare the advantages and disadvantages of various compression methods. (Kaiyang Xue)
Added by a1784184
Improve the current algorithm method already on the leaderboard. (Po-Yi Lee)
Added by a1806297
Develop initial algorithm that might contribute to the project (traditional algorithm like brute force or greedy
17 Done
Finish the research of helpful algorithm for data compression such as run-length encoding (RLE) , and analyse the possibility and efficiency of using it. (Xiaoman Li)
Added by a1804817
Summarize the content of the client meeting and Upload the recording for the sprint review meeting. (Yang Lu)
Added by a1782685
Collect the questions we gonna ask during the client meeting. (Yang Lu)
Added by a1782685
Finish the draft program of the compression file of the intro one and the fast one data.(Yuanpeng Liu)
Added by a1784375
Determine the members' task in the initial phase (coding and documentation)
Added by a1806297
Determine and set up communication channels
Added by a1162576

12 In progress

- Develop initial algorithm that might contribute to the project (traditional algorithm like brute force or greedy might be low effciency)
  Added by a1804817
- Try to improve the data structure(Hechen Wang).
  Added by a1786785
- Implement the base framework described in System Architecture of initial report
  Added by a1162576
- document meeting minutes (Yang Lu, Liuyang Yun)
  Added by a1811518
- Organize all of these team meetings. (Yang Lu)
  Added by a1782685
- Create wiki page to collect ideas about compression algorithms (Karl Asenstorfer)
  Added by a1162576
- Enhance the algorithm with multi-threading / multi-processing. (Karl

17 Done

- Determine the frequency and time of meetings for member each week
  Added by a1784375
- Summarize the major content of the twice team meeting and ask for other ideas for the rest of the team members. (Yang Lu)
  Added by a1782685
- Research papers/techniques on block model compression
  Added by a1784375
- Research on the format and meaning of both input data and output data. (Po-Yi Lee)
  Added by a1806297
- Upload the valid code on the competition website, shake out your environment and let us test a first solution at https://titan.maptek.net/ . (Yuanpeng Liu)
  Added by a1806297
- Collected information on the contents of the snapshot and the initial report from the Q&As on discussion board and use the tips to check whether our drafts meet the requirements. (Po-Yi Lee,

12 In progress

Try to improve the data structure(Hechen Wang).

Added by a1786785

Implement the base framework described in System Architecture of initial report

Added by a1162576

document meeting minutes (Yang Lu, Liuyang Yun)

Added by a1811518

Organize all of these team meetings. (Yang Lu)

Added by a1782685

Create wiki page to collect ideas about compression algorithms (Karl Asenstorfer)

Added by a1162576

Enhance the algorithm with multi-threading / multi-processing. (Karl Asenstorfer)

The Parent blocks are separate and can be compressed in parallel. This should decrease the total time at which we compress the blocks.

Added by a1162576

17 Done

Collected information on the contents of the snapshot and the initial report from the Q&As on discussion board and use the tips to check whether our drafts meet the requirements. (Po-Yi Lee, Liuyang Yun)

Added by a1811518

Identify the user needs as well as the structure of the model (the colours of blocks are distributed randomly or more integrated like land and the sea), which might influence the use of algorithm. (Xiaoman Li)

Added by a1804817

Sketch out initial software architecture. (Karl Asenstorfer)

Added by a1162576

Develop Set of Coding Standards. (Karl Asenstorfer)

Added by a1162576

Set up Static Code Analysis. (Karl Asenstorfer)

Added by a1162576

Configure static analysis tools (Karl Asenstorfer)

Added by a1162576

Sprint Backlog and User Stories

There are no major modifications to the Sprint Backlog and User Stories, and we are mainly trying to implement the software architecture mentioned in the initial report. The abstraction of the model was to be initialized implementation.

Also, after the client meeting, we realized that we still needed to collect information on various compression algorithms and put them into a wiki page. And to do that, we need a better division of labor.

Definition of Done

In the current phase:

- The code we develop is required to take standard input (strings of the form "x, y, z, x_size, y_size, z_size, 'domain'") and produce the result on standard output described in the project documentation.
- Either a .exe file or a Python script must be submitted to a verification service: MAPTEK TITAN.
- According to the user stories, we can submit our code once it improves the compression rate and processing speed, no matter how good they are.
- The datasets of input block models we implement must be comma-separated values (CSV) where each line encodes a block as a string of the form "x, y, z, x_size, y_size, z_size, 'domain'" and the code we develop is required to output a stream of the same format.
- The algorithm we develop must process a block model in slices of no more than parent block thickness at a time, rather than loading the entire input stream into memory first.
- All pull requests of code must be submitted on our GitHub repository, tested, and reviewed by two other team members (or one in the case of documentation/admin).
- The branches must pass the static code analysis before being merged. The Static Analysis ensures that the codes meet the PEP8 style standards (what all python code should aim for) and other issues such as unused variables and cyclomatic complexity. PEP8 is especially important for the developers because it mandates a set of conventions for things such as class/function names, spacing, and comment style. It allows all the developers to be able to read every developer's code in the same way.
- In the initialization implementation of the abstract classes of the model, we build the Model class. The Model is the internal representation of the block data model. The Model holds the current set of ParentBlocks which constitute the current slice of the model. The model also holds the mappings between domains and domain tags, which the Blocks use.
- In the initialization implementation of the abstract classes of the model, we build a ParentBlocks class. The ParentBlocks subdivide the model exactly with no remainder. Each Parent Block contains a collection of Blocks. Each Parent Block has a size and position. All of the Blocks that the Parent Block contains are positioned

relative to the ParentBlock position. The ParentBlock implements the iterator protocol, so can be iterated over natively in for loops.

- We also created the Block class in the abstraction part. This class describes a block, which has a size, a position relative to the start of its ParentBlock, and a domain tag.
- There is also a classmethod to combine all of those blocks and return them into a new block.
- We initialized the implementation of the modular compression system. All the compression algorithms are separated into modules. Each one takes as input a ParentBlock. The compressionEngine is responsible for delegating the compression to the various algorithms. The parentBlocks are then written to the output.
- The Modular Architecture has been implemented, but it still needs to be tested with the runner.py in the Windows environment and then try to merge it.
- We also clarified some information in README, created requirements.txt, and configured flake8.

Summary of Changes:

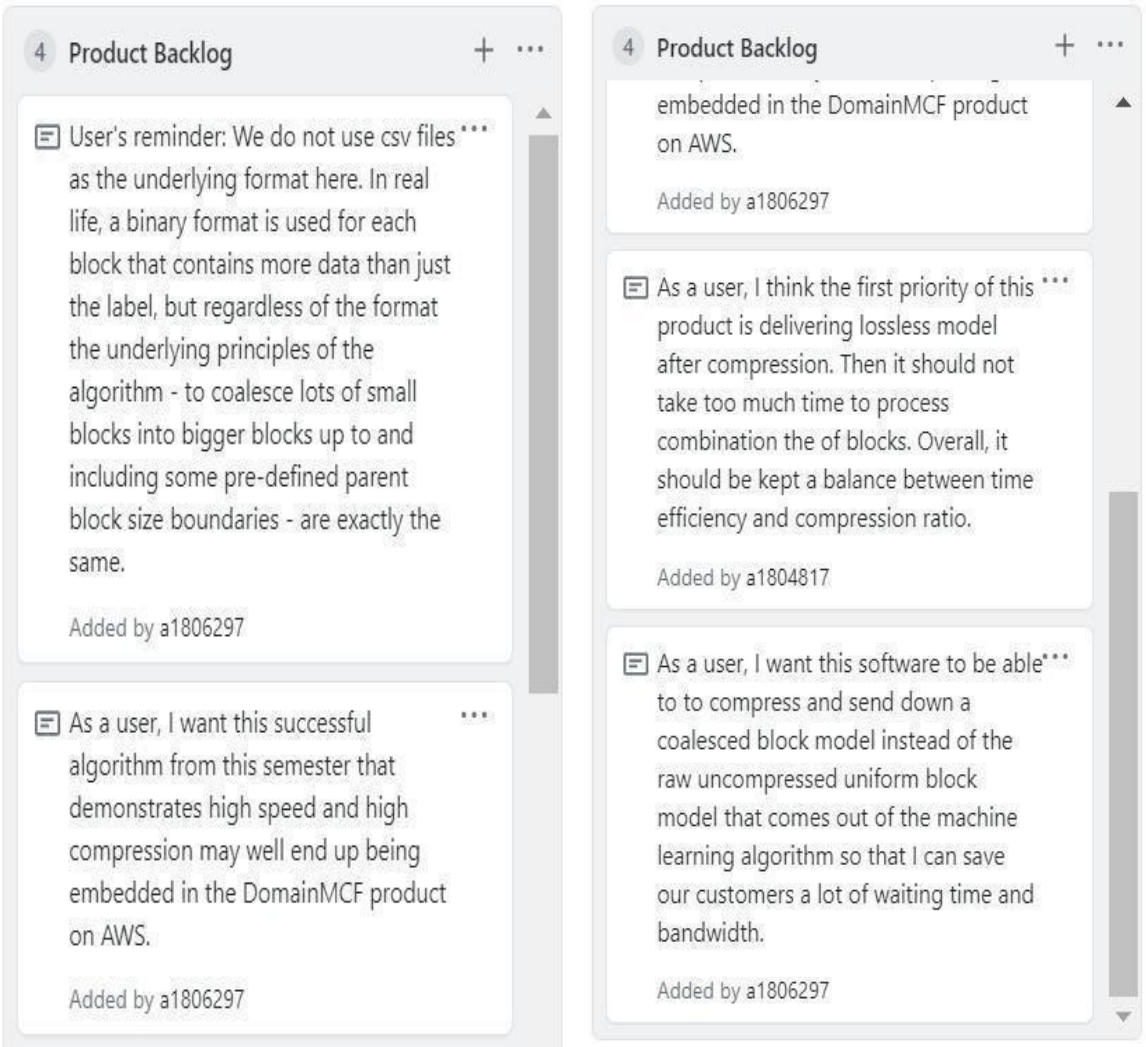
This week we had the client meeting and the sprint review meeting. During the formal sprint review meeting, we discussed what we had already done: test our algorithm for the dataset one and two. In addition, we talked about what we plan to do in the next step: implementing the algorithm which can accept some larger datasets, and trying to visualize our output more clearly. We collected both the technical problems and some of the non-technical problems, which we will probably face when implementing the new algorithm in the next week. At the end of the review meeting, we decided what technical and non-technical questions we planned to ask during the client meeting.

During the client meeting, most of the questions were answered smoothly. For issues that need to be addressed further in the implementation process, we'll ask questions on the Slack channel. At the same time, we realized that we needed a more rational division of labor. The task is now divided into several modules: compressing the dataset, visualizing the output, searching for suitable compression algorithms, and more. So we're trying to reclassify these tasks in a much more acceptable way to make sure that everyone is contributing.

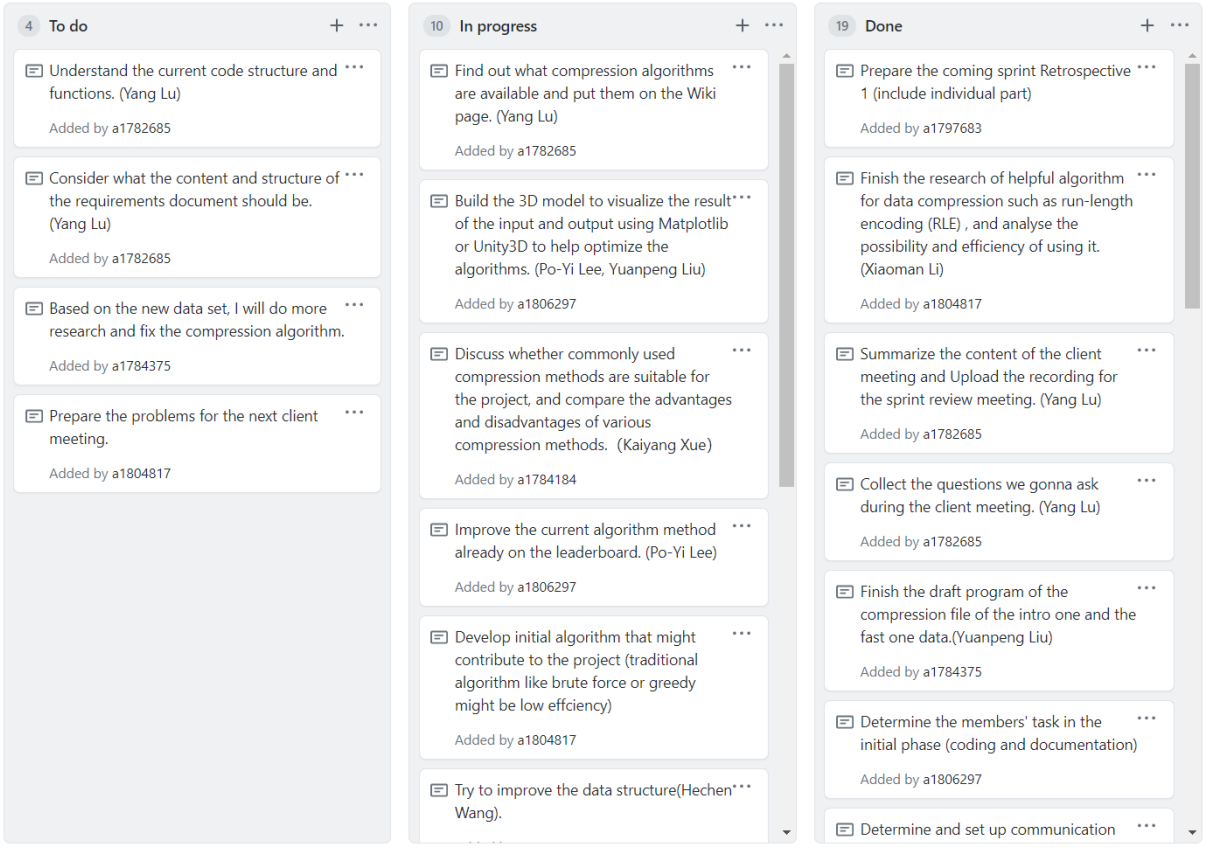
Snapshot Week 6 of Group BLOCKS7PG

Product Backlog and Task Board

Product Backlog:



Task Board:


4 To do
Understand the current code structure and functions. (Yang Lu)
Added by a1782685
Consider what the content and structure of the requirements document should be. (Yang Lu)
Added by a1782685
Based on the new data set, I will do more research and fix the compression algorithm.
Added by a1784375
Prepare the problems for the next client meeting.
Added by a1804817
10 In progress
Find out what compression algorithms are available and put them on the Wiki page. (Yang Lu)
Added by a1782685
Build the 3D model to visualize the result of the input and output using Matplotlib or Unity3D to help optimize the algorithms. (Po-Yi Lee, Yuanpeng Liu)
Added by a1806297
Discuss whether commonly used compression methods are suitable for the project, and compare the advantages and disadvantages of various compression methods. (Kaiyang Xue)
Added by a1784184
Improve the current algorithm method already on the leaderboard. (Po-Yi Lee)
Added by a1806297
Develop initial algorithm that might contribute to the project (traditional algorithm like brute force or greedy might be low effciency)
Added by a1804817
Try to improve the data structure(Hechen Wang).
19 Done
Prepare the coming sprint Retrospective 1 (include individual part)
Added by a1797683
Finish the research of helpful algorithm for data compression such as run-length encoding (RLE) , and analyse the possibility and efficiency of using it. (Xiaoman Li)
Added by a1804817
Summarize the content of the client meeting and Upload the recording for the sprint review meeting. (Yang Lu)
Added by a1782685
Collect the questions we gonna ask during the client meeting. (Yang Lu)
Added by a1782685
Finish the draft program of the compression file of the intro one and the fast one data.(Yuanpeng Liu)
Added by a1784375
Determine the members' task in the initial phase (coding and documentation)
Added by a1806297
Determine and set up communication

10 In progress

algorithm like brute force or greedy might be low effciency)

Added by a1804817

Try to improve the data structure(Hechen Wang).

Added by a1786785

Implement the base framework described in System Architecture of initial report

Added by a1162576

Organize all of these team meetings. (Yang Lu)

Added by a1782685

Create wiki page to collect ideas about compression algorithms (Karl Asenstorfer)

Added by a1162576

Enhance the algorithm with multi-threading / multi-processing. (Karl Asenstorfer)

The Parent blocks are separate and can be compressed in parallel. This should decrease the total time at which we compress the blocks.

Added by a1162576

19 Done

Determine the frequency and time of meetings for member each week

Added by a1784375

Summarize the major content of the twice team meeting and ask for other ideas for the rest of the team members. (Yang Lu)

Added by a1782685

Research papers/techniques on block model compression

Added by a1784375

Research on the format and meaning of both input data and output data. (Po-Yi Lee)

Added by a1806297

Upload the valid code on the competition website, shake out your environment and let us test a first solution at https://titan.maptek.net/ . (Yuanpeng Liu)

Added by a1806297

Collected information on the contents of the snapshot and the initial report from the Q&As on discussion board and use the tips to check whether our drafts meet the requirements. (Po-Yi Lee,

10 In progress

algorithm like brute force or greedy might be low effciency)

Added by a1804817

Try to improve the data structure(Hechen Wang).

Added by a1786785

Implement the base framework described in System Architecture of initial report

Added by a1162576

Organize all of these team meetings. (Yang Lu)

Added by a1782685

Create wiki page to collect ideas about compression algorithms (Karl Asenstorfer)

Added by a1162576

Enhance the algorithm with multi-threading / multi-processing. (Karl Asenstorfer)

The Parent blocks are separate and can be compressed in parallel. This should decrease the total time at which we compress the blocks.

Added by a1162576

19 Done

the tips to check whether our drafts meet the requirements. (Po-Yi Lee, Liuyang Yun, Sprint 1)

Added by a1811518

Identify the user needs as well as the structure of the model (the colours of blocks are distributed randomly or more integrated like land and the sea)，which might influence the use of algorithm. (Xiaoman Li)

Added by a1804817

document meeting minutes (Yang Lu, Liuyang Yun, Sprint 1)

Added by a1811518

Sketch out initial software architecture. (Karl Asenstorfer)

Added by a1162576

Develop Set of Coding Standards. (Karl Asenstorfer)

Added by a1162576

Set up Static Code Analysis. (Karl Asenstorfer)

Added by a1162576

Configure static analysis tools (Karl Asenstorfer)

Sprint Backlog and User Stories

There are no major modifications to the Sprint Backlog and User Stories, and we are mainly trying to implement the software architecture mentioned in the initial report. The abstraction of the model was to be initialized implementation.

Also, after the weekly meeting, we clarify some confusion about the existing framework. Besides that, we decided what we were going to do in the next phase: Test after we got a bigger data set.

Definition of Done

In the current phase:

- The code we develop is required to take standard input (strings of the form "x, y, z, x_size, y_size, z_size, 'domain'") and produce the result on standard output described in the project documentation.
- Either a .exe file or a Python script must be submitted to a verification service: MAPTEK TITAN.
- According to the user stories, we can submit our code once it improves the compression rate and processing speed, no matter how good they are.
- The datasets of input block models we implement must be comma-separated values (CSV) where each line encodes a block as a string of the form "x, y, z, x_size, y_size, z_size, 'domain'" and the code we develop is required to output a stream of the same format.
- The algorithm we develop must process a block model in slices of no more than parent block thickness at a time, rather than loading the entire input stream into memory first.
- All pull requests of code must be submitted on our GitHub repository, tested, and reviewed by two other team members (or one in the case of documentation/admin).
- The branches must pass the static code analysis before being merged. The Static Analysis ensures that the codes meet the PEP8 style standards (what all python code should aim for) and other issues such as unused variables and cyclomatic complexity. PEP8 is especially important for the developers because it mandates a set of conventions for things such as class/function names, spacing, and comment style. It allows all the developers to be able to read every developer's code in the same way.
- In the initialization implementation of the abstract classes of the model, we build the Model class. The Model is the internal representation of the block data model. The Model holds the current set of ParentBlocks which constitute the current slice of the model. The model also holds the mappings between domains and domain tags, which the Blocks use.
- In the initialization implementation of the abstract classes of the model, we build a ParentBlocks class. The ParentBlocks subdivide the model exactly with no remainder. Each Parent Block contains a collection of Blocks. Each Parent Block has a size and position. All of the Blocks that the Parent Block contains are positioned relative to the ParentBlock position. The ParentBlock implements the iterator protocol, so can be iterated over natively in for loops.
- We also created the Block class in the abstraction part. This class describes a block, which has a size, a position relative to the start of its ParentBlock, and a domain tag.
- There is also a classmethod to combine all of those blocks and return them into a new block.
- We initialized the implementation of the modular compression system. All the compression algorithms are separated into modules. Each one takes as input a ParentBlock. The compressionEngine is responsible for delegating the compression to the various algorithms. The parentBlocks are then written to the output.
- The Modular Architecture has been implemented, but it still needs to be tested with the runner.py in the Windows environment and then try to merge it.
- We also clarified some information in README, created requirements.txt, and configured flake8.

Summary of Changes:

This week, we mainly completed the familiarization with the software framework written by Karl and laid a good foundation for the subsequent development. In addition, the

visualization of the compression results is also underway, and it has been decided to implement it by unity3D. In addition, in the weekly group meeting, we discussed the following detailed division of labour.

We collected some compression algorithms during the implementation process and evaluated their operating efficiency, compression ratio, and compression time. In this way, relatively excellent compression algorithms are selected as the compression algorithms for subsequent large data sets.

Declaration:

I attended:

- the sprint 1 review and sprint 2 planning meeting on 10th September, 2021 with the tutors, Aryaman Dhawan and Will Reid.
- the sprint 2 retrospective meeting on 16th September, 2021 with the other team members.

What went well in the sprint:

In the sprint 2, Karl successfully completed the software framework including development environment and static analysis which can help the compression implementation and the testing of the speed and compression rate. Some of the other members did the testing on the framework and reported to the group if there were any errors and issues. Eventually, we had the useful software framework for the compression project. In addition, we had a better understanding of the problem the team faced in this sprint and found out that we needed more communication on the project and more voice of each member's progress. Most of us were too shy and silent in this sprint to let the teams stop because we had no confidence with our contribution when we had only very little progress or understanding.

What could be improved:

The efficiency of algorithm optimization should be improved quickly in the next sprint because we had only almost a month left to optimise the compression result. The way of our communication should also be changed in the following sprint because the team members were quite confused with the other members' progress according to the sprint review in the meeting. Therefore, the scrum master should delegate tasks more clearly and efficiently in order to make sure every team member knows what they should do in this sprint and also the

deadline of tasks should be decided to make everyone on the track. Lastly, we will number each entry in the sprint backlog and product backlog according to their relationship. This can enable us to trace the tasks in the specified sprint and also connect it with the product requirements and user stories.

What will the group commit to improve in the next sprint:

According to the sprint 2 retrospective meeting, we encouraged each member to speak up on Slack no matter how well each one's progress went. In addition, we decided to make more efforts on the previous and new dataset in the two week's mid break and had set the deadline on the task each member would like to contribute. This could make our progress including the visualisation model and optimizing the compression algorithm more efficiently. We also need to spend more time discussing the best compression algorithm we have so far with each other to enable everyone to be more familiar with the current code.

Comment on your progress this sprint:

My progress completed:

- Testing of the completed software architecture.

My in-progress tasks:

- Build the 3D model to visualize the result of the input and output using Matplotlib or Unity to help optimize the algorithms
- Manage to Improve the current algorithm method already on the leaderboard.

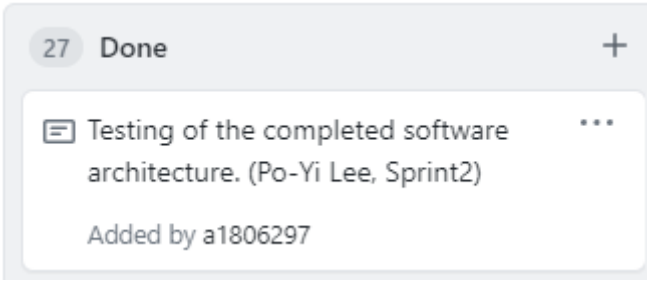


Figure 7: My progress completed

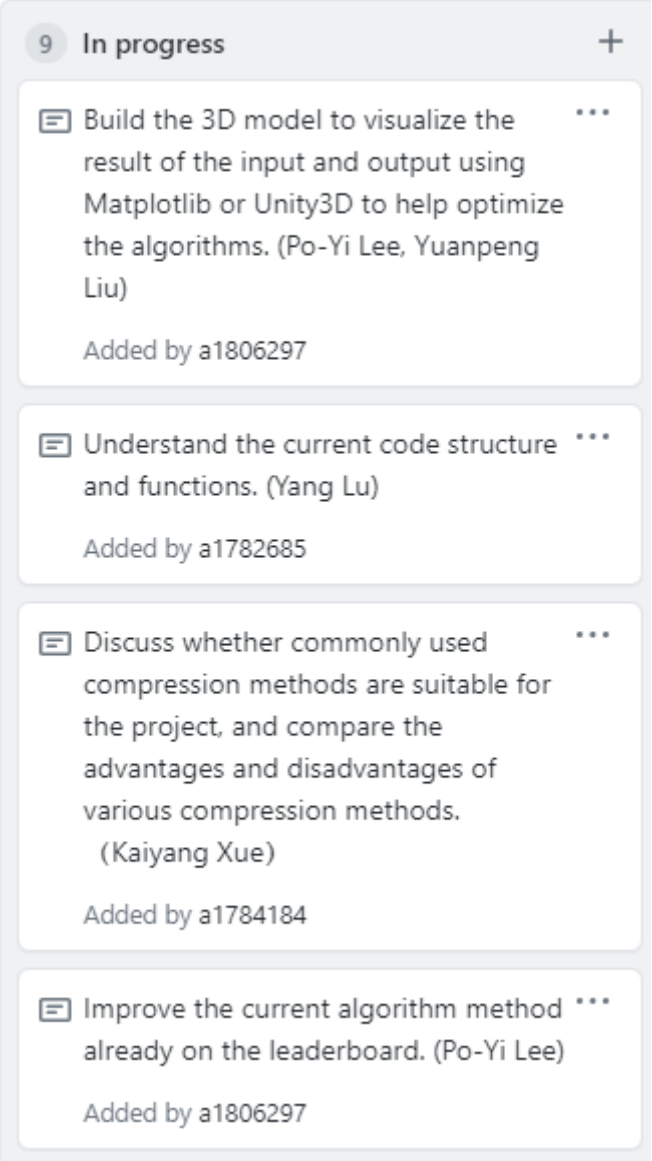


Figure 8: My in-progress tasks

In this sprint, I tested a new software framework to verify its usability and functionality to make sure that it can be used to deal with different datasets in the following sprint. Most of the testing was good but we found out that the parser name of the command on running had some problems. But we considered it not a big deal. The visualization model of compressed block using Unity is still in progress because we are not familiar with C# language and the syntax of Unity. Lastly, I was also stuck with the algorithm optimization for the initial dataset in this sprint and need to spend more time on that.

Requirements Changes:

Two new datasets including the_stratal_one_42000000_14x10x12 and the_big_one_987417600_8x8x5 were released at the end of this sprint by the customer. The most important feature of the new datasets is that the block size becomes very large so that the runtime could be a lot longer than the dataset in the first sprint. Since the parent block size and the compression scope are both huge, the way of reading the input should be considered more carefully in the following sprints. We could read the input blocks of one row or read the input blocks of more rows with the same label before every compression algorithm. The runtime might be affected by the bigger size of the given blocks because it may take more time in every loop function so that we should be more careful with how the compression algorithm works.