Optimising Interaction with Teravoxel Images

Sabina Yang, Greg Sands, The University of Auckland, New Zealand

Background

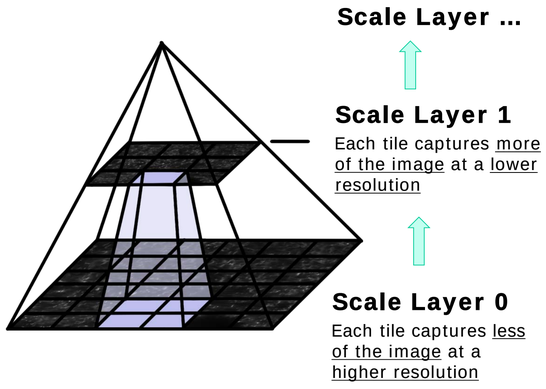
Cardiovascular disease is the leading cause of death in most developed countries, including New Zealand [1]. Obtaining and analyzing high resolution, 3D images obtained from cardiac microscopy can help us better understand these diseases.

These large, 3D images can use up to a terabyte or more of computer storage, and therefore require an efficient storage system in order for these images to be accessible on laboratory computers.

Project Progress

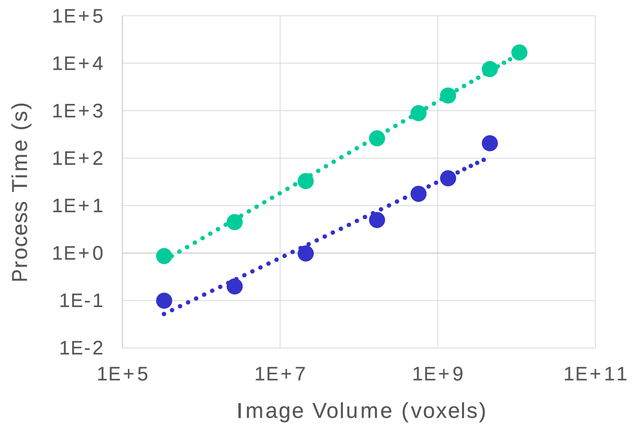
1. Convert image data into a tile-based file format (Zarr file format [2]).
2. Store the tile-based image data across a range of image resolutions.
3. Develop an image volume software to visualize, analyse and annotate the multiresolution, tile-based image data.

Multiresolution Tile-based File Format

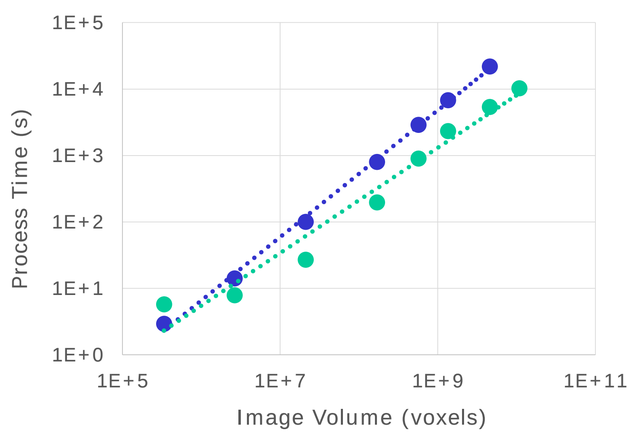


Computational Expense of the Read and Write Step

The computational expense of two read and write methods were investigated, 1. reading all image data into a NumPy array first before storing it into a Zarr array (●); and 2. reading data into a Zarr array an image at a time (●).



Combined Process Time of the read and write step (log-log). It was found that reading into a NumPy array was a significantly faster method.



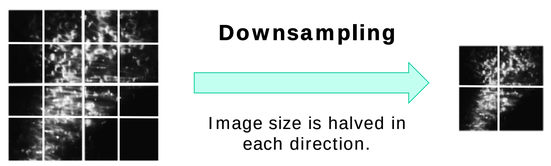
Peak Memory Usage of the read and write step (log-log). It was found that storing in a NumPy array used much more memory at large image volumes. Because of this, the images sizes are limited to within the computer memory.

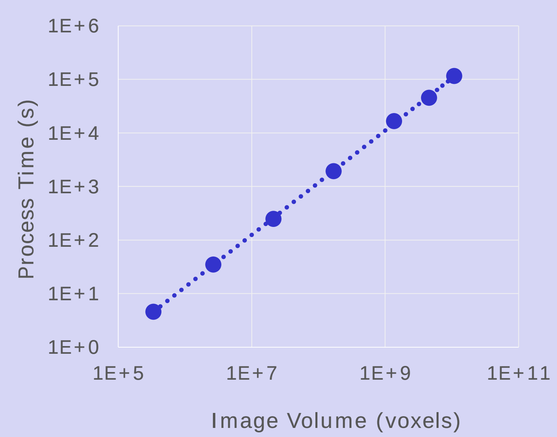
Software Process

1. Read either a single 3D tiff image, or a sequence of 2D tiff images into an array format.
2. Perform 3D image downsampling to reduce the image dimensions by half.
3. Save the image data across multiple resolutions to the Zarr file format.

3D Downsampling of Image Data

The resolution of the 3D images are reduced down to various scale levels through 3D image downsampling. During each downsampling step, the image dimensions are reduced down by half by merging every eight voxel cube into one.





Process Time of the 3D downsampling step (log-log). The process time is linearly dependent on the size of the image, as expected.

Key Functionalities of the Image3D Python Module

`Array3D(object)`
An object which stores the image data in a Zarr array

`Array3D.downsample()`
Performs one downsampling step on the image data

`Array3D.save()`
Saves data into a Zarr file format

`Image3D.read_3d_tif()`
Reads in a single tiff image file

`Image3D.read_tif_dir()`
Reads in a directory of tiff image files

`Image3D.load()`
Loads pre-existing Zarr files into an Array3D object

Future Works

- Investigate techniques to improve the speed of the read and write process using the Zarr array.
- Perform more benchmark tests to see how computational expense can be further reduced.
- Develop an image volume software to visualize and annotate the multiresolution, tile-based image data.

Reference

[1] Site designed and developed bka interactive ltd, N. (2020). *Coronary heart disease | Health Navigator NZ*. [online] Health Navigator New Zealand. Available at: https://www.healthnavigator.org.nz/health-a-z/c/coronary-heart-disease/ [Accessed 4 Feb. 2020].

[2] Bria A, Iannello G, Onofri L, Peng H. TeraFly: real-time three-dimensional visualization and annotation of terabytes of multidimensional volumetric images. Nature methods. 2016 Mar;13(3):192.