Image By Image FDK

M.Aishwarya(coe14b020)

Abstract

Computed tomography (CT) is an imaging procedure that uses special x-ray equipment to create detailed pictures, or scans, of areas inside the body.
The open source version of CT Reconstruction code of Lucid is rtk fdk test which uses rtk FDK Cone Beam Reconstruction Filter. In this project, a new filter i.e. Image By Image FDK Cone Beam Reconstruction filter has been implemented which is more efficient in terms of memory used by the rtk fdk test program.

Introduction

A **CT scan** makes use of computer processed combinations of many X-ray measurements taken from different angles to produce cross-sectional (tomographic) images (virtual "slices") of specific areas of a scanned object, allowing the user to see inside the object without cutting.
The main problem with the rtk fdk test program is that the memory requirement for the program is very high because of large set of input projections which are required for accurate reconstruction which makes it difficult to run on a laptop.

Objective

The main goal is to change the way the input is processed by the reconstruction filter so that the program consumes less memory without losing any input information and producing the same accurate reconstructed output as rtk fdk test without manipulating any of the standard internal filters.

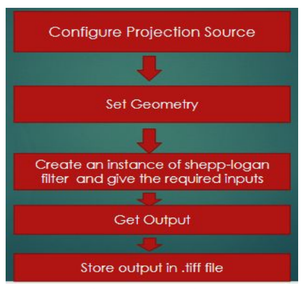
Literature Survey

feldkamp-davis-kress (fdk) cone beam reconstruction: This algorithm is a widely used filtered back projection algorithm for three-dimensional image reconstruction from cone-beam (CB) projections measured with a circular orbit of the x-ray source
Cone Beam CT scan: It is a medical imaging technique where the X-rays are divergent, forming a cone. This method, from each point on the circle we take projection of the object.
FDK Cone Beam Reconstruction Filter: It is a compound filter i.e. this filter takes help of some other filters to produce the final output. It is a mini pipeline filter which combines the following steps:
Itk:: Extract Image Filter to extract sub-stacks of input projections.
FDK Weight Projection Filter for 2D weighting of the projections. Weight filter is applied to the input projection to correct the error occurred due to tilt of detector.
FFT Ramp Image Filter for ramp filtering: The ramp filter is a high pass filter that does not permit low frequencies that cause blurring to appear in the image.
FDK Back Projection Image Filter for back projection of the projections. Filtered back projection is used which first finds the Fourier transform of the projection then finds it inverse Fourier transform. The inverse Fourier transform is back projected to get the original volume.

rtk fdk test Algorithm

rtk fdk test: This test generates the projections of a simulated Shepp-Logan Phantom. A CT image is reconstructed from each set of generated projection images using the FDK algorithm
The Shepp–Logan phantom is a standard test image used for the development and testing of image reconstruction algorithms. This has mainly 10 ellipses. We can generate the projections of this object by using itk shepp-logan filter.

Step 1: Generating Shepp-Logan Phantom Projections



Step2: Reconstruction using rtk FDK Cone Beam Reconstruction Filter:

Memory Requirements:
Input:800 images of size 2048*2048 and float is stored in each pixel.
Memory required for input = (800 * 2048 * 2048 *4)/(10^9) = **13.4 GB**
Output:3D volume of size 1024*1024*1024 and float is stored in each pixel.
Memory required for output = (1024 * 1024 *1024 *4)/ (10^9) = **4.29 GB**
Total memory requirement =17.7GB

Methodology/Contribution

The idea is to modify the rtk FDK Cone Beam Reconstruction filter to take one input projection at a time in order to increase the memory efficiency of the overall program without affecting the accuracy of the reconstructed output volume.
The first step is to develop Image By Image FDK filter to replace the FDK Cone Beam Reconstruction filter and remove the itk extract image filter which is used to extract sub stacks of size 16 from the entire input projection dataset because in Image By Image FDK, the input is only one projection so there is no need of extracting sub stacks.

Image By Image FDK Algorithm

Step 1: Generating Shepp-Logan Phantom Projections
Step2: Generating .tiff files using Image Series Writer: It writes its input data to a series of output files.
Step 3: Creating an instance of rtk Image By Image FDK Filter: An instance of tomography source is created, this is used to store output, set its origin, size, spacing accordingly to fit the output.
An instance of import image filter is created, you need to configure this instance. We need to set its origin, size, spacing as that of the input projections. Import Filter is used to convert 2Dprojection into 3D itk image as the internal filters can only accept 3D inputs.

Image By Image FDK Algorithm

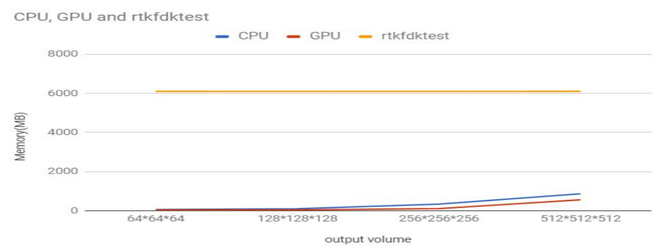
for(i=0; i<NumberOfProjections,i++)
{
-read the 2d input projection
-set it as input to the import filter(import filter converts 2d input into 3d output)
- Set the inputs for feldkamp:
The inputs of feldkamp in the first iteration (i=0) are:
- Empty volume (tomography Source->Get Output())
- Projection0 converted to 3d itk image by import filter(import Filter->Get Output())
- Geometry of that respective projection

The inputs of feldkamp for all other iterations are:
- The output of Image By Image FDK filter in the previous iteration is given as input in the present iteration.
- Projectioni converted to 3d itk image by import filter (import Filter->Get Output ())
- Geometry of that respective projection

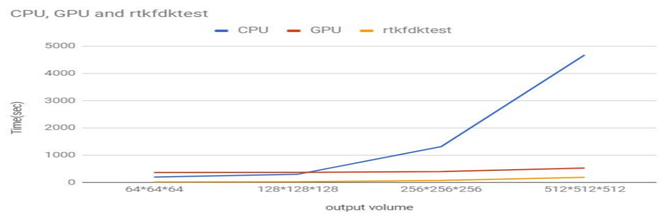
- if(i=NumberofProjections -1)
{
Write the output of feldkamp into a .vtk file
}
}

Results

Input Size: 1048*1048
No of projections=1440
Memory Analysis:



Time Analysis:



Novelty/Application

The main goal is to change the way the input is processed by the reconstruction filter so that the program consumes less memory without losing any input information and producing the same accurate reconstructed output as rtk fdk test without manipulating any of the standard internal filters which was achieved

References

- The ITK Software Guide Book 1: Introduction and Development Guidelines Fourth Edition Updated for ITK version 4.12
- http://www.openrtk.org/Doxygen/rtkfdktest_8cxx.html
- https://en.wikipedia.org/wiki/Tomographic_reconstruction
- www.slideshare.net/likanpatra/computer-tomography-ct-scan