Stability and performance of FDK algorithm for CBCT in multithreaded implementation

A S Kussainov

1 Physics and Technology Department, al-Farabi Kazakh National University, Almaty, Republic of Kazakhstan
2 National Nanotechnology Laboratory of Open Type, al-Farabi Kazakh National University, Almaty, Republic of Kazakhstan
E-mail: arman.kussainov@gmail.com

Abstract. We have written, optimized and tested a C/C++ project designed to simulate a CBCT experiment with an arbitrary shaped and structured three dimensional radiological phantom with high resolution and precision. This phantom was generated and later used to simulate the recording of the multiple cone beam projections under successive rotations of the coupled source and detector around it. This radiological phantom can be used to generate projections in the various testing chambers conditions and configurations including the mammography case. The synthesized projections were used to reconstruct the internal structure of the phantom with FDK algorithm. Our simulation code is also able to accept an arbitrary set of projections from other simulation software or from the real computer tomography setup. The simulation code includes various filtering techniques and has been hardware accelerated using the OpenMP directives.

1. Introduction

There is a constant need for the basic software to simulate, plan and process results of the CBCT experiment [1]. One can use it to process the data acquired from a home-made X-ray scanner, to improve its design, or to develop the algorithms and plan a CT experiment of their own based on the data from an existing commercial scanner. Under the regular projection recording conditions, that is the constant distance between the source, phantom and detector, different only by the successive rotations of the source-detector pair around the phantom, even the simplest implementation of simulation and reconstruction software produces a very good image quality and could be an invaluable tool for the data analysis and medical experiment planning. Hardware acceleration, OpenMP, CUDA directives, etc., are, in particular, crucial in achieving the satisfying performance and image resolution.

2. Methods

When we discuss the gamma rays’ attenuation by matter we need to consider three basic mechanisms of the gamma ray scattering, which are the photoelectric effect, Compton scattering, and pair production. Depending on the situation one or two of these mechanisms may become more prominent than the rest of them. For the sake of numerical simulations, all three of them a commonly combined into the single $\mu$ value, depending on the energy of gamma ray quanta. In this case, we assume the gamma rays attenuation process is reduced to a simple exponential
form \( I = I_0 e^{-\mu \rho x} \), where \( I \) and \( I_0 \) are the intensities (gamma quanta counts) after and before the gamma ray traverses the sample, \( \mu \) is the attenuation coefficient measured in \([cm^2/g]\) and \( \rho \) is the density of the traversed material.

This project is using the tables of X-Ray Mass Attenuation Coefficients for energies ranged from 1 KeV to 20 MeV and for elements with \( Z = 1 \) to 92 plus data for 48 Additional Substances of Dosimetric Interest provided by the Physical Measurement Laboratory at National Institute of Standards and Technology, Gaithersburg, MD, USA. [4]. The \( \mu \) values of the traced material should be put into correspondence with the provided energy spectrum of the X-ray sources.

As the first step in this project, we have created a versatile digital radiological phantom of \( 1024 \times 1024 \times 1024 \) voxels and more with each voxel containing information about the type of material by assigning a unique numerical index to each voxel. Attenuation coefficients \( \mu \) for these materials are retrieved from the database using the index assigned to them. The separate C/C++ class is designed to parse this attenuation coefficients database and relate it to the spectrum of X-ray source in use.

The phantom is optionally saved in the form of a text file but always kept in the allocated random access memory. The shape and the internal structure of this phantom could be used to mimic, by means of parallelograms and ellipsoids, arbitrary artificial and biological structures. Later on, this information is used to create a projection of this phantom by illuminating it with the gamma ray source of particular energy spectrum. In our case we have constructed the high contrast Shepp-Logan phantom of the human head with different resolutions designed to study the effectiveness of reconstruction procedure from the projections recorded in the previous stage.

The classical variant of the high contrast Shepp-Logan phantom consists of the multiple nested and overlapping ellipsoids resembling the human head, see Figure 1 (a)-(b) and reconstructed cross sections on Figures 5 (a)-(b). The regular way to define this phantom is to make the densities of the internal structure to be the fraction of unity, see [2, 3]. The reference material uses the density of water. The thin bright shell encapsulating the structure shown on Figure 1 (a) is taken to be water. There could be multiple areas of overlapping densities of different shapes and sizes and one can study the outcome contrast produced in numerical simulation after the reconstruction stage.

![Figure 1](image1.png)

**Figure 1.** From left to the right: (a) original, head-on projection taken at \( \beta = 0^0 \) for the \( 1024 \times 1024 \times 1024 \), (b) 3D visualization of the internal structure of the similar \( 256 \times 256 \times 256 \) Shepp-Logan phantom.

Figure 2 (a) shows the basic parameters determining the recording geometry. These are the distance from source to the center of the phantom and the distance from the center of the
phantom to the center of the detector. Detector’s width and height ultimately define the size of the projection. The blue arrows are pointed in the direction of rotation.

On Figure 2 (b) we can see the central part of the cone beam illuminating the phantom along the XY plane. The \((r, \phi)\) parameters are standing for the polar coordinates defining the location of the illuminated/reconstructed, depending on the stage of the experiment, point inside the phantom and \(\beta\) is an exposition angle for the successive rotation of the source and detector around the phantom. If necessary, detector could be moved up or down, parallel to the ZX plane, following the displaced shadow (projection) of the phantom produced by the shifted down or up X-ray source.

The code has the clear entry points and structure allowing us to introduce different modification to the numerical experiment according to the design of the experimental setup. For example, for the optional mammography case, we place the detector close to the phantom and the source is moving now along the circle arc with the center lying on the detector.

There is also an option to record two other modified projections for the research purposes. The first is the original projection which is distorted by the Gaussian filter and the second is again the original projection but now it blurred by the white noise. These projections are meant to simulate the real conditions with imperfections and artifacts.

The quality of the recorded projections, beside the other obvious tracing conditions are controlled by the OpenCV function which allows us to set the image type, its compression and bit depth.

The full numerical simulation of the CBCT experiment has multiple stages. The aforementioned phantom definition and generation stage is followed by the second stage designed to create multiple projections of this phantom taken at the different rotation angles of the source-detector couple around the phantom.

This stage is normally the most computationally intensive one, because the tracing of such an object requires billions of complex operations performed for each voxel. Besides the multithreading, there were multiple other techniques employed to limit the set of calculations only to the area occupied by the phantom.

Reconstruction stage should closely retrace back all the steps from the projections stage with its predefined geometry, otherwise no reasonable image will be reconstructed. The most widely used and implemented method is known to be an FDK method and its variations. The reconstruction procedure is now given by the following formulas.
\[ f(x, y) = \int_{0}^{2\pi} \frac{1}{U^2} \left( R_\beta (s) g(s' - s) \right) \frac{D}{\sqrt{D^2 + s^2}} ds d\beta, \]

where \( g(s) = \frac{1}{2} h(s) \) and \( h(s) \) is the inverse Fourier transform of \( H(\omega) \). That is

\[ H(\omega) = |\omega| b_\omega(\omega), \]

\[ b_\omega(\omega) = \begin{cases} 1, & \text{if } |\omega| < W \\ 0, & \text{if } |\omega| \geq W, \end{cases} \]

and

\[ U(r, \phi, \beta) = \frac{SO + OP}{SO}. \]

All notations could be found on Figure 2. Here \( R_\beta (s) \) and \( s \) are the projection and point’s position along the ray connecting the source and the point on projection which has been traced back. All recording and reconstruction parameters are listed in the special input file file which controls all the procedures and parameters in the experiment. Besides the basic recording parameters operator could control the location of the cross section, its size and orientation, thickens of the cross section and parameters of the Tukey filter applied at the reconstructions stage.

3. Results and Discussions

The eight-core desktop PC which was used for our simulations runs on Intel Core i7 4790K, 4.0GHz processor with DDR-3 16Gb/1866MHz PC14900 RAM. Microsoft C++ compiler and Matlab packages were deployed on Windows 10 OS.

The simulation package consists of three separate classes, see Figure 3. These three separate classes are responsible for the phantom definition, phantom tracing and projections construction, and finally backprojection. The phantom tracing and projections procedure, see Phantom.cpp class, requires the knowledge about the spectrum of the X-ray source and how this spectrum is attenuated by different types of materials contained in the phantom provided by

![Figure 3. Structure of the numerical simulation complex.](image)
the `GenerateMu.cpp` class. Finally, the backprojection is done by the `Loadprojection.cpp` class according to the 3D Radon transform defined in the introduction.

Despite the rather simple model employed, we have observed several interesting phenomena corresponding to the real life CBCT experiment. For example, the beam hardening which is often observed with the polychromatic gamma ray spectra, see Figure 4.

As the gamma ray travels through the body, the low energy component is attenuated, absorbed and scattered more easily, while the remaining high energy part stays on course and is not diminished significantly. Thus, the beam transmission does not follow the simple exponential decay expected for a monochromatic ray and is not properly encoded later in image intensity distribution across the projection. This is invariably observed with the high atomic number materials such as bone, iodine, or metal. Compared to the low atomic number materials such as water, or for the low energy X-ray tubes without prior monochromatic filtering, these high atomic number materials have dramatically increased attenuation at lower energies thus making the passing beam harder resulting mainly in “streaks” artifacts and more complicated patterns.

Figure 4. Beam hardening artifacts observed for the rotated cube in a vacuum.

Figure 5, describes the main results obtained for the Shepp-Logan phantom generated with our simulation program. The size, or more precisely, the volume occupied by the phantom, is $1024 \times 1024 \times 1024$ voxels. The two first pictures have been plotted using the color map in Matlab and displaying the pretty decent match with the original phantom, Figure 5 (a), and its reconstruction, Figure 5 (b), are plotted across the central plane. Knowing that the filtering in the Fourier space is provided mainly by application of the ramp filter, see equation (2), the appearance of the next two profiles on Figures 5 (c)-(d) looks quite normal. All these cross sections have been reconstructed from 361 projections taken from the phantom with one degree step.

Figure 5. (a) Cross section for the original $1024 \times 1024 \times 1024$ Shepp-Logan phantom, (b) cross section for the reconstructed $1024 \times 1024 \times 1024$ Shepp-Logan phantom, and (c) profile comparison of the previous two pictures taken at $Y = 460$ and (d) $X = 460$.

The slope of the presumably sharp edges inside these phantom reconstruction profiles is affected by the three steps normalization procedure. In the first step we decide which part of the picture represent the zero level. It could be the empty spaces inside the phantom or, taking into account that the source is relatively far away from the detector and illumination of the detector plane is pretty close to the parallel wavefront condition, we could choose this point somewhere in the place not occupied by the phantom as well. The second step is about the scaling of the intensity level of another reference voxel inside the phantom up or down to the
corresponding value in the original sample. All other voxels are scaled by the same ratio as this leading reference voxel. We know, from the definition of the phantom, the maximum intensity (reconstructed density) observed across the sample is unity, thus, the third stage is to assign this value to all voxels with the intensity level greater than one.

On the next Figure 6 we can see that the image quality degrades very moderately as the number of projections decreases and their spacing around the 360 degree circle increases. This is the consequence of the redundancy of information contained in the full set of projections as well as the good quality of our synthetic projections, meaning that the individual projection is free from artifacts and has uniform recording conditions, exposure, number of pixels etc., compare to the other projection images. It should be mentioned that the time taken to reconstruct the first slice is 3 minutes while for the last two slices it is only 35 and 18 seconds.

Figure 6. Normalized $XY$ plane cross sections for the $1024 \times 1024 \times 1024$ Shepp-Logan phantom taken at $Z = 384$ and reconstructed from the different number of projections. (a) Rotation angle takes values from $0^\circ$ to $360^\circ$ with $1^\circ$ degree step, (b) rotation angle takes values from $0^\circ$ to $180^\circ$ with $2^\circ$ degree step, (c) rotation angle takes values from $0^\circ$ to $180^\circ$ with $4^\circ$ degree step.

The next three sets of slices on Figures 7, 8 and 9 are taken at the different positions of the secant plane parallel to the three cardinal planes $XY$, $YZ$ and $ZX$. They have been also properly normalized in order to reflect the expected density distribution across the phantom.

4. Conclusions

We have presented software package implementing the FDK algorithm for CBCT on C/C++ language. Projections for this algorithm are generated by the special C/C++ class from an arbitrary digital phantom constructed and generated locally. Projections could be supplied by an outside source or X-ray scanner.

We have deployed multiple filters, such as the Hamming window in Fourier space and the Tukey window with modified parameters in direct space. The later one can easily manipulate the intensity distribution across the image without its prior normalization.

We have provided the capability to reconstruct an arbitrary cross section of the phantom parallel to the three cardinal $XY$, $YZ$ and $ZX$ coordinate planes. The thickness of the reconstructed cross section could be also chosen by an operator.

The proposed numerical model of CBCT using FDK method is an effective tool for solving the task of the complex phantom’s internal structure reconstruction for the arbitrary volume of interest of the $1024 \times 1024 \times 1024$ voxels and higher. Hardware acceleration by means of the OpenMP directives is used and provided speed and effectiveness at all stages of the simulation.

The most computationally intensive part, if activated, is the tracing of the original phantom at the first stage of the projections generation. Notwithstanding we were able to implement an
Figure 7. Normalized $XY$ plane cross sections for the $1024 \times 1024 \times 1024$ Shepp-Logan phantom taken at different values of $Z$.

Figure 8. Normalized $ZX$ plane cross sections for the $1024 \times 1024 \times 1024$ Shepp-Logan phantom taken at different values of $Y$. 
Figure 9. Normalized \(YZ\) plane cross sections for the \(1024 \times 1024 \times 1024\) Shepp-Logan phantom taken at different values of \(X\).

effective and fast procedure while taking into account the property of the test chamber and phantom's dimensions.

The simulation package has a big potential for optimization, upgrade and further implementation of the multiple features.

Acknowledgments
This research was commissioned and funded by the ADANI Company, producing and manufacturing Medical, Security, Spectroscopy and Inspection X-ray Systems in Minsk, Republic of Belarus.

Publication efforts, and planing of the further development of phantom tracing and multithreading, were supported by “Fundamental and applied research in related fields of physics of terrestrial, near-earth and atmospheric processes and their practical application” project #BR05236494, administered by the Ministry of Science and Education, Republic of Kazakhstan.

References
[1] Kak A C and Slaney M 1988 Principles of Computerized Tomographic Imaging. (New York, NY: IEEE Press) p 327
[2] MathWorks support page Create Modified Shepp-Logan Head Phantom Image (Available at: https://www.mathworks.com/help/images/ref/phantom.html, Accessed: 2018-06-29)
[3] Jain Anil K 1989 Fundamentals of Digital Image Processing (Englewood Cliffs, NJ: Prentice Hall) p 439
[4] Hubbell J H and Seltzer S M 2004 Tables of X-Ray Mass Attenuation Coefficients and Mass Energy-Absorption Coefficients (version 1.4). (Available at: http://physics.nist.gov/xaamdi, Accessed: 2018-06-29. Gaithersburg, MD:National Institute of Standards and Technology)