

A.S. Kussainov^{1*} , Y.V. White^{2,3} , M.A. Em¹ ,
E.T. Myrzabek¹ , E.T. Salavatova¹ , S.T. Zhaldybayev¹ 

¹al-Farabi Kazakh National University,
Almaty, Republic of Kazakhstan

²East Georgia State College, Swainsboro, USA

³University of Tennessee Space Institute, Tullahoma, USA

*e-mail: arman.kussainov@gmail.com

Transition to mammography in the regular computed tomography simulation and reconstruction software

Abstract. This paper presents a modification of the previously developed and maintained computed tomography simulation and reconstruction software for the mammography case. New additional modules are designed to process the mammography data. Mammography provides incomplete information about the subject taken from the limited number points of view but as a result has potentially minimized exposure to radiation for the biological tissue under study. We implement this method by providing new standalone independent mammography modules in our software package. These modules are responsible for creating the projections' set according to the operator's input of exposure angles, phantom's structure, and other multiple recording parameters. These additional modules reconstruct the data from these generated sets of projections or take the real medical data as input. Contrast and features' recognition are particularly important elements of the study due to the limited number of projections in set. Our software could be used in combination with any real commercial mammography scanner as well as for research purposes to train medical and physics personnel and study for novel methods of contrast and image enhancement.

Key words: Computed tomography, mammography, X-rays, FDK algorithm, backprojection, image reconstruction.

Introduction

Computed tomography (CT) is essentially a medical diagnostic and screening procedure employing the soft part of the X-ray spectrum generated by a tube operated at about one hundred of kilovolts and less. The known modifications of this modality, suited for different parts of the human body, are projection radiographic imaging and X-ray mammography [1].

Due to an ionizing nature of X-ray radiation mammography is considered as recommended for the "women who place a higher value on the potential benefit than the potential harms" [2]. Screening and diagnostic mammography radiation doses may be as high as 36 mGy per person, see [3], [4], [5]. Thus, a great number of efforts are dedicated to make medical screening procedures safe. Localizing the area of exposure to the organ of interest is a part of these efforts. Human breasts are paired organs and

represent complex body parts rich in blood vessels and connective tissues with distinct particularities depending on the age and health group [6].

The question of good image contrast and resolution is of utmost importance due to the complex nature of the tissue under study. The imaging artifacts come in abundance due to the complexity of the tissue of interest but could be verified by other imaging modalities like magnetic resonance imaging, etc., see [7]. The rich physics of X-ray interaction with breast tissue [8] and imperfections in numerical reconstruction procedures are the other major sources of imaging artefacts. Besides this, the hardware itself, like the ever-increasing size of the flat detectors, causes the multiple artefacts like scatter artefacts in the study of Wei Zhao et al., [9].

The simulations of this particular imaging technique are of big interest and importance [10]. All these studies are strongly dependent on the phantom in use, see [11], [12]. In general, they are roughly

falling into two groups of digital and real physical anthropomorphic phantoms. Real physical phantoms require tedious systematic studies of the tissue's absorption properties complimented by studies of synthetic and biological materials used to mimic these tissues, see [13], [14]. Unique combination of the digital and real physical phantom has been demonstrated using a 3D printer [15]. Nowadays, the 3D printing material could be anything starting from plastic and ending with stem cells.

Ideal CT case assumes that the phantom is recorded from as many angles as possible, so the contrast variation at two adjacent points will be maximized by a contribution from as many projections as possible. The lack of this data, random and systematic variations in recording and

reconstruction conditions, as well as the imperfection of our computer model incapable of grasping the whole multitude of physical phenomena contributing to the case lead to numerous imaging artifacts [16].

Mammography, due to the limited range of exposure angles, needs additional efforts to preprocess and postprocess the images, plus the introduction of several filtering techniques, etc.

Methods and materials

Mammography modules were built using the framework of the CT project developed in [17]. On the next Figure 1, the map of the project, including all main classes, input and output text, and graphical data are shown.

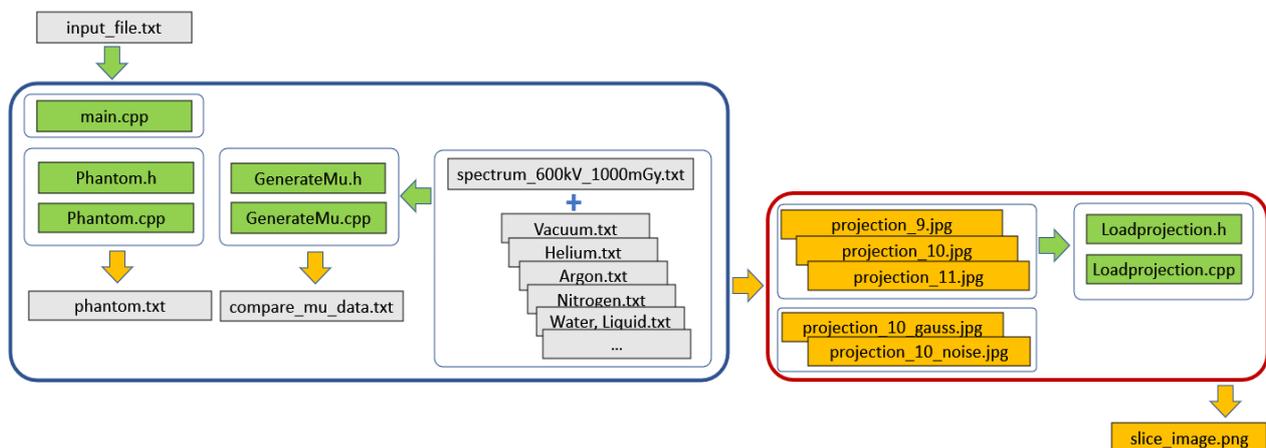


Figure 1– The map of the original CT project files and classes (shown as green) including the input files (shown as grey), generated text (shown as grey as well) and image outputs (orange rectangles). The independent X-ray projection generation and CT image reconstruction parts are shown separately, in blue and red rectangle areas correspondingly.

Two, to the great extent independent, modules for producing X-ray projections and reconstructing the CT image from them are shown separately, see enclosing blue and red rectangles.

Original data with information about the project directories and folders, selected recording and reconstruction regime, the elements and compounds used in simulation, the size and orientation of the reconstructed slices and digital phantom, Fourier space filtering techniques for images, the number of projections generated during recording stage and number of them used in reconstruction (not necessarily the same number) and complete description of the digital phantom are given in the `input_file.txt`.

The digital phantom is generated by the `Phantom.h/Phantom.cpp` class according to the input data in `input_file.txt` file. For the record and possible future modifications, the replica of the phantom is saved in `phantom.txt` file.

Another class `GenerateMu.h/GenerateMu.cpp` is responsible for mapping the data in the original absorption coefficients database files like `Vacuum.txt`, `Helium.txt` etc. files onto X-ray source spectrum energy grid given in the `spectrum_600kv_1000mGy.txt` file. As a result of this procedure, we have `compare_mu_data.txt` file containing information about the absorption coefficients for all elements used in this particular digital phantom mapped to the energy grid of the

source X-ray spectrum using the linear interpolation method. This file is also created just for the records and to verify the correctness of information kept in the memory and used for phantom tracing procedure.

The main.cpp part of the project coordinates the work of these two classes and using the object tracing functionality of the class Phantom.h/Phantom.cpp generates a series of X-ray projections projection_9.png, projection_10.png etc. shown in orange color on the same figure. The projections could be generated with additional Gauss blur and noise added to simulate the real conditions, see projection_9_gauss.png, projection_10_noise.png data.

The data generated by this part of the project is used for further reconstruction procedure by Loadprojection.h/Loadprojection.cpp class to produce slice_image.png CT image.

Figure 2 shows, from top to bottom, the original X-ray projection recorded from our digital phantom at zero rotation angle, the central slice of this

phantom reconstructed by a regular CT backprojection algorithm using our simulation and reconstruction software and the mammography slice taken at the same location using the modified software package. It takes 365 X-ray images to produce a single CT slice image (b) and 179 for the mammography scan slice (c). It takes 150 and 70 seconds to produce images (b) and (c).

The apparent nonuniform intensity distribution across the phantom is inherited by the mammography case and strongly exaggerated by the absence of approximately half of the projected data compared to the regular CT case.

Our digital phantom is constructed as a half of the ellipsoid squeezed along one of the axes to mimic the real mammography recording condition when the human breast is placed between two plates transparent to the X-ray radiation and designed to keep the object immobile. This procedure flattens the breast to some extent, decreasing the effective thickness of the breast tissue.

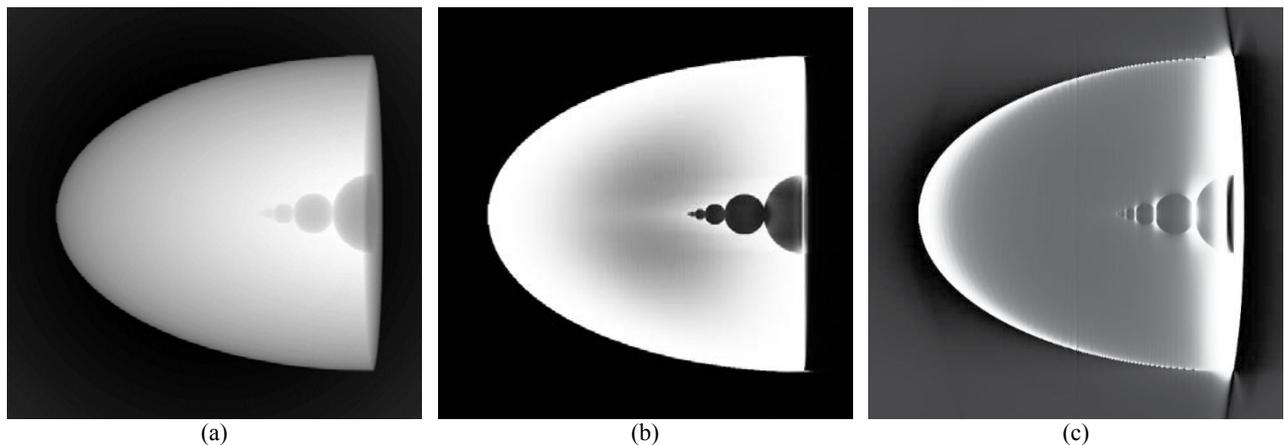


Figure 2 – (a) The 1024x1024 pixel image of the X-ray projection taken at 0-degree angle of rotation during a regular CT scan; (b) The 675x675 pixel image of the CT slice reconstructed at the central plane; (c) The 675x675 pixels image of the mammography slice reconstructed at the same plane; Every slice in figures (b) and (c) is produced by summation of 11, one voxel thick, slices grouped around the target cross section at 338 voxel mark.

The ellipsoid is made of adipose tissue which is basically the fat or triglyceride [18]. The interaction with X-ray radiation is described through the regular absorption and dissipation mechanisms coded in our numerical tracing procedure. The absorption and dissipation are quantified in the database of the X-ray mass attenuation coefficients from the “NIST Standard Reference

Database 126” [19]. Compare to the widely used Monte-Carlo methods for simulation [20] of the

particles’ ensemble and the other statistical events with some probability distribution function, this type of simulation [21],[22] is much easier but still produces physically sound results. For the photons’ energy ranged from 1 and up to 2000 KeV, the mass attenuation coefficient μ/ρ , where $\rho=0.95 \text{ g/cm}^3$ is the density of the tissue, takes the values from $2.628 \times 10^3 \text{ cm}^2/\text{g}$ down to $1.698 \times 10^3 \text{ cm}^2/\text{g}$. This absorption properties are mainly defined by the chemical composition of adipose tissue having

hydrogen (H), carbon (C), and oxygen (O) roughly in 1:5:2 ratio.

Figure 3 gives a comparison of the CT and mammography recording geometries. The top part (a) of Figure shows the sketch of the regular CT scan with the digital phantom aimed to resemble a human breast. The source and detector are moving around the phantom to produce multiple X-ray images taken at different angles of rotation. The middle picture,

part (b), shows the same CT geometry but from the top view, along the Z axis and the transformation needed to be taken to perform the mammography scan are shown at the bottom. In part (c), of the Figure 3, we move the detector plane in the immediate vicinity of the phantom and fix it there, which minimizes the movement artefact and radiation dosage to achieve the reasonable contrast and confine the source movement to the half of the ZY plane.

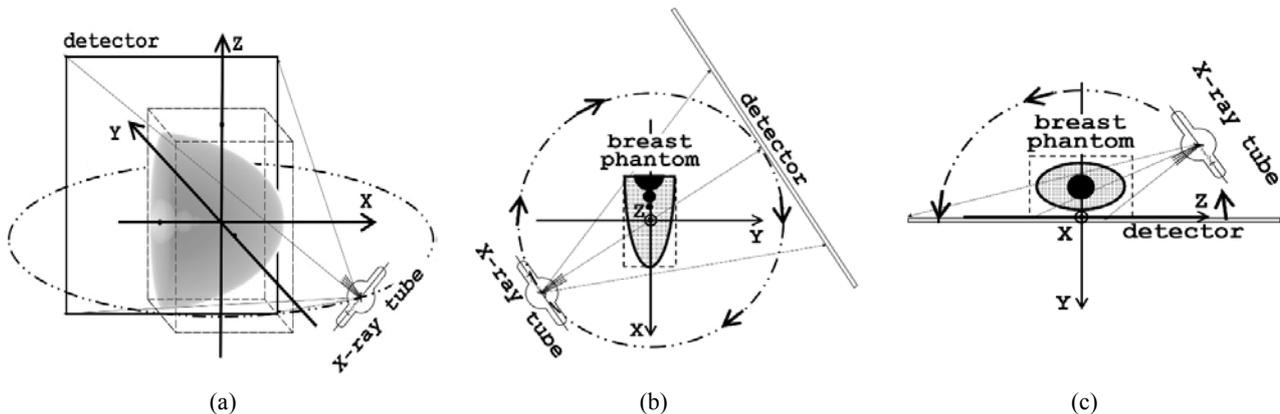


Figure 3 – From top to bottom: (a) The 3D sketch of the regular CT scan of the phantom under consideration (not to scale); (b) Top view of the same scheme showing the position of the phantom relative to the fixed XYZ coordinate system; (c) The same setup transformed to the mammography case.

The other recording condition assumes that in both cases the exposition camera is filled with dried air. This was taken into account while tracing the X-ray outside the phantom. The principal axes of the ellipsoid are 1024, 512, and 256 voxels and the numerical simulation resolution of 72 voxels per centimeter translates it to 14x7x4 centimeters.

The actual size of the phantom simulation volume, available for filling it with an arbitrary combination of objects of various shapes and materials, is shown by the dashed line parallelepiped sized in 512x256x512 voxels. The size of the ellipsoid that does not fit into this volume is automatically being cut off from simulations. The distance from the center of the phantom to the center detector is 1040 voxels (14 cm) and the distance from the source to the phantom is 2000 voxels (28 cm). The detector's size is assumed to be 1024 by 1024 voxels. The size of the reconstructed slice images, see Figure 2, is controlled by the FOV (Field of View) parameter which is taken to be 675 by 675 voxels.

Besides the regular filtering techniques of the FDK method, there is the slice half thickness parameter, that is, how many 1 voxel thick slices times two plus one are added to one image to increase

SNR (signal to noise) ratio and overall contrast of the images.

Rather than building a complete numerical ductal network in the digital breast phantom [23], we wanted to qualitatively describe the methods' resolution, and so the phantom was filled with empty spheres of gradually decreasing size placed along the biggest axis of the ellipsoid. The diameters of the spheres are 128, 64, 32, 16, 8, and 4 voxels correspondingly, which makes them 1.78, 0.89, 0.44, 0.22, 0.11 and 0.06 cm small. Similar periodic structures of variable size to test resolution were used in [24]. In all cases, the central slice in ZX plane was reconstructed as an obvious choice to maximize the contrast and uniformity of illumination conditions.

The closer look at Figures 3 (b) and (c) tells us that the mathematical description of the backprojection algorithm stays precisely the same for the mammography case as long as we preserve the direct mapping from the projections onto the plane of the reconstructed slice. The addition of the imaginary plane to Figure 3 (c) placed perpendicular to the source's beam and following the source from left to the right in a circular path, staying behind the phantom makes this statement clear.

The X-ray spectrum was taken for the G-297 rotating anode X-Ray tube from the Varex Imaging data specification for 100 KeV with a maximum peak fluence of 1.57×10^7 photons per mm^2 at 8.5 KeV, see Figure 4.

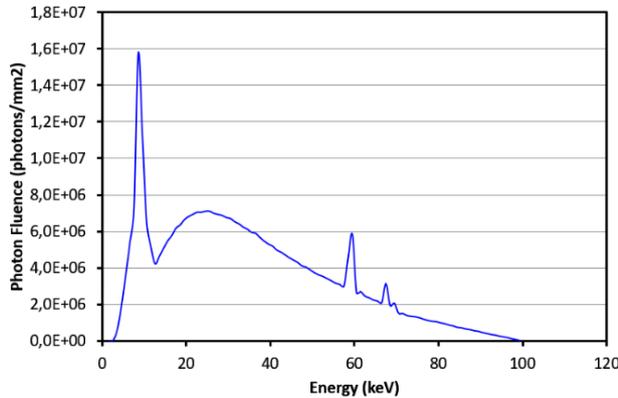


Figure 4 – TASMICS [26] (tungsten anode spectral model using interpolating cubic splines) spectrum at specified air kerma

Total fluence is summed up to 3.61×10^8 photons per mm^2 and the average energy is 35 keV. HVL (Half Value Layer) for Al and Cu are 0.0634 and 0.0089 mm correspondingly. Specific air kerma was set for 100 mGy. This particular X-ray spectrum was used for numerical simulations of the phantoms described previously in the text.

The universal reconstruction method described by the well-known FDK algorithm named after Feldkamp, Davis, and Kress, see [25]. For finite number of projections, FDK algorithm could be described as a summation of the previously recorded projections back on the plane of interest, see next equation (1)

$$f(x, y) = \Delta B \sum_{i=1}^M \frac{1}{U^2(x, y, \beta_i)} \tilde{Q}_{\beta_i}, \quad (1)$$

where \tilde{Q}_{β_i} stands for projection selected from the series of the recorded ones and selected for backprojection, β_i is the rotation angle for the source-detector couple for this particular projection and $\Delta B = 2\pi/M$, where M is the total number of projections, in general recorded from 0 to 2π value of β_i . Normalization factor $U^2(x, y, \beta_i)$ have somewhat complex structure depending on the geometry of the source beam, its trajectory and other factors and has explicit dependence on x and y , in case of a simple

parallel beam geometry. In simple cases it is just a scaling factor. Tilda sign, which is used in \tilde{Q}_{β_i} projection's notation, expresses the fact that the original projection has been filtered in Fourier space by $H(w)$ filter, see the next expression

$$H(w) = |w|b_w(w) \quad (2)$$

and $b_w(w) = \begin{cases} 1, & |w| < W \\ 0, & |w| \geq W \end{cases}$

This partially solves the reconstructed image smearing caused by the presence of the projections' data in the areas otherwise containing no real image. The W value measures the size of the frequencies' window in Fourier space. The value of τ is the sampling rate for the digital image of the projection and directly relates to the size of its pixels.

With the ability to reconstruct cross-section images from the sequence of real data taken from medical or experimental CT scanners, our software is now capable to produce the full-scale CT and mammography simulation of different digital phantoms of high resolution and complex, user-defined structures. The 8-core Intel Core i7-4790K desktop computer with 32 Gb RAM was used for simulations.

Results and discussion

The input_file.txt, if modified to work in the mammography case, requires a couple of yes flags on lines 4 and 5 to set the mammography case and specify that the generated projection will be stored locally on the hard drive and to the directory specified by the path on line 3. The value of source.to.phantom:2000 still needs to be defined explicitly on line 10, although the value of phantom.to.detector is irrelevant now and filled with arbitrary digits, because it will be recalculated in the code to move the object close to the detector:

```

1 path.mu.data:Attenuation coefficients\\NIST
  compounds\\
2 path.spectrum.data:100kV.txt
3 path.projection.data:C:\\CT_project\\mammography\\
4 mammography.yes
5 projections.yes
6 element.id:Vacuum.0
7 element.id:Air, Dry.1
8 element.id:Water, Liquid.2
9 element.id:Adipose Tissue.3
10 source.to.phantom:2000
11 source.elevation:0
12 phantom.to.detector:99999999
    
```

```

13 field.of.view:512
14 slice.plane:XY
15 target.cross.section:320
16 slice.half.thickness:0
17 zero.padding.factor:0
18 cut.background:0.0
19 tukey.window.alpha:0.0
20 detector.width:1024
21 detector.height:1024
22 start.angle:-89 // from a lesser number to bigger one
23 end.angle:89
24 projections.number:179
25 image.compression:9

```

The lines 13 through 19 define the orientation, size and graphics properties of the reconstructed mammography slice. Lines 20 through 25 give the size of the detected X-ray projection, and the start and end angle for the total number of 181 projections taken with 1 degree step.

The parsing of the input_file.txt takes place in the main.cpp block, thus, historically, the loop initiating the recording of consecutive projections, that is the rotation of the source/ detector pair also resides in main.cpp. In order to initiate the recording of the sequence of mammographic projections, we need to go through the following steps:

```

1 float t_before = 0, t_after = 1.0;
2 int m_amo = 0;
3 //Phantom tracing and creating the projections
4 for(float d_egree=start_angle; d_egree<=end_angle;d_egree+=degree_step){
5     if (mammography_yes==0){
6         int shift_degree=int(d_egree)%90;
7         t_before=(float(source_to_phantom)-(float)sin((45.0f+shift_degree)/180.0f*pi)
8             *float(max_dim)/1.4142f)
9             /float(source_to_phantom+phantom_to_detector);
10        t_after=(float(source_to_phantom)+(float)sin((45.0f+shift_degree)/180.0f*pi)
11            *float(max_dim)/1.4142f)
12            /float(source_to_phantom+phantom_to_detector);
13        p_hantom.create_projection(-source_to_phantom,source_elevation,
14            phantom_to_detector,d_egree,detector_width,
15            detector_height,interpolated_mu_data,
16            muvalues.s_pectrum,interpolation_points,
17            elements_count,muvalues.total_hw,
18            image_compression,voxels_per_cm,t_before,t_after,
19            m_amo, projections_dir);
20    }
21    else{
22        m_amo = 1;
23        phantom_to_detector=static_cast<int>(depth/2.0f)+1;
24        cout << "\n\nThe phantom.to.detector value is set to 1/2 (half) of the
25        phantom.depth and equals to " << phantom_to_detector << "\n\n";
26        float source_to_detector_center=static_cast<float>(source_to_phantom
27            +phantom_to_detector);
28        int sourcetophantom=static_cast<int>(source_to_detector_center
29            *cos(d_egree*pi/180.0f))-phantom_to_detector;
30        int sourceelevation=static_cast<int>(source_to_detector_center
31            *sin(d_egree*pi / 180.0f));
32        t_before=(float(sourcetophantom)-float(depth)/2.0f)
33            /float(sourcetophantom+phantom_to_detector);
34        t_after=(float(sourcetophantom)+float(depth)/2.0f)
35            /float(sourcetophantom+phantom_to_detector);
36        p_hantom.create_projection(-sourcetophantom,sourceelevation,
37            phantom_to_detector, d_egree,detector_width,
38            detector_height,interpolated_mu_data,
39            muvalues.s_pectrum,interpolation_points,
40            elements_count,muvalues.total_hw,
41            image_compression,voxels_per_cm,t_before, t_after,
42            m_amo, projections_dir);
43        if ((end_angle==start_angle)(projections_number==1)){break;}
44    }
45 }

```

To compare to different recording strategies, CT and mammography, one should take a look, side by side, at lines 2-19 and 22-43.

Backprojecting part of the code is introduced into the Loadprojection.h/Loadprojection.cpp class. The snippet of the C/C++ code with the Open MP parallelization directives implemented by Microsoft in Visual Studio 2015 is listed below. As usual, the names of the variables have one to one correspondence with the physical parameters of the real setup.

The first piece of code is given for the ZX orientation of the reconstructed slice. The choice of data acquisition scheme “mammo” and target plane orientation, that is along “ZX” plane, are made at the first line. The next lines 2 through 6 move the

detector close to the phantom and define the trajectory of the source. Nested loops on lines 7, 10, and 14 define the location of the reconstructed slice in the fixed (Xt,Yt,Zt) laboratory frame of reference. The ci and ck variables defined on lines 12 and 16 refer to the coordinate system of the detector and map the data from projections taken from the phantom at different angles, back to the reconstructed slice (Xt,Yt,Zt). Multiple cores and threads supported in contemporary computers increase performance of the code by running OpenMP directives on line 9. The maximum performance is achieved by using this pragma only with one nested loop over Xt. Variable c_onvolved refers to the projected data convolved with the high pass filter in Fourier space.

```

1  if((mammo == 1) && (a_ffix == "ZX")) {
2  float source_to_detector_center = static_cast<float>(-y_source + y_detector);
3  float sourcetophantom = source_to_detector_center*c_os
4  -static_cast<float>(y_detector);
5  float sourceelevation = source_to_detector_center*(-s_in);
6  float source_to_detector = source_to_detector_center*cos(view_angle);
7  for (int Yt = t_arget - d_elta; Yt <= t_arget + d_elta; Yt++) {
8  float Yp = static_cast<float>(Yt) - half_slice;
9  #pragma omp parallel for
10 for (int Xt = 0; Xt < slice_size; Xt++) {
11 float Xp = static_cast<float>(Xt) - half_slice;
12 int ci = static_cast<int>(Xp*source_to_detector / (sourcetophantom + Yp)
13 + static_cast<float>(w_detector) / 2.0);
14 for (int Zt = 0; Zt < slice_size; Zt++) {
15 float Zp = static_cast<float>(Zt) - half_slice;
16 int ck = static_cast<int>(sourceelevation - (sourceelevation - Zp)
17 *source_to_detector/(sourcetophantom + Yp)
18 +static_cast<float>(h_detector) / 2.0);
19 if ((ck >= 0) && (ck < h_detector) && (ci >= 0) && (ci < w_detector)) {
20 slice.at<float>(Zt,Xt)+=
21 c_onvolved.at<float>(h_detector - 1 - ck, w_detector - 1 - ci);
22 }
23 }
24 }
25 }
26 }
```

The float $c_{os} = \cos(-view_angle)$; and float $s_{in} = \sin(-view_angle)$; variables are introduced out of the multiple nested loops to speedup the calculations and to be calculated only once per rotation angle.

There are two other similar blocks responsible for reconstruction in XY and YZ planes correspondingly. To clarify the overall backprojecting geometry, they are listed below as well. See the next insert for the XY plane:

```

27 if((mammo == 1) && (a_ffix == "XY")) {
28 float source_to_detector_center = static_cast<float>(-y_source + y_detector);
29 float sourcetophantom = source_to_detector_center*c_os
30 -static_cast<float>(y_detector);
31 float sourceelevation = source_to_detector_center*(-s_in);
32 float source_to_detector = source_to_detector_center*cos(view_angle);
33 for (int Zt = t_arget - d_elta; Zt <= t_arget + d_elta; Zt++) {
```

```

34     float Zp = static_cast<float>(Zt - half_slice);
35     #pragma omp parallel for
36     for (int Xt = 0; Xt < slice_size; Xt++) {
37         float Xp = static_cast<float>(Xt) - half_slice;
38         for (int Yt = 0; Yt < slice_size; Yt++) {
39             float Yp = static_cast<float>(Yt) - half_slice;
40             int ci = static_cast<int>(Xp*source_to_detector / (sourcetophantom + Yp)
41                 + static_cast<float>(w_detector) / 2.0);
42             int ck = static_cast<int>(sourceelevation - (sourceelevation - Zp)
43                 * source_to_detector / (sourcetophantom + Yp)
44                 + static_cast<float>(h_detector) / 2.0);
45             if ((ck >= 0) && (ck < h_detector) && (ci >= 0) && (ci < w_detector)) {
46                 slice.at<float>(Xt, Yt) +=
47                     c_onvolved.at<float>(h_detector - 1 - ck, w_detector - 1 - ci);
48             }
49         }
50     }
51 }
52 // for XY slice the modified factor U2 could be taken out of all loops
53 slice = slice.mul(U2);
54 }

```

and for the YZ plane:

```

55     if ((mammo == 1) && (a_fix == "ZX")) {
56         float source_to_detector_center = static_cast<float>(-y_source + y_detector);
57         float sourcetophantom = source_to_detector_center*c_os
58             -static_cast<float>(y_detector);
59         float sourceelevation = source_to_detector_center*(-s_in);
60         float source_to_detector = source_to_detector_center*cos(view_angle);
61         for (int Xt = t_target - d_elta; Xt <= t_target + d_elta; Xt++) {
62             float Xp = static_cast<float>(Xt) - half_slice;
63             #pragma omp parallel for
64             for (int Yt = 0; Yt < slice_size; Yt++) {
65                 float Yp = static_cast<float>(Yt) - half_slice;
66                 int ci = static_cast<int>(Xp*source_to_detector / (sourcetophantom + Yp)
67                     + static_cast<float>(w_detector) / 2.0);
68                 for (int Zt = 0; Zt < slice_size; Zt++) {
69                     float Zp = static_cast<float>(Zt) - half_slice;
70                     int ck = static_cast<int>(sourceelevation - (sourceelevation - Zp)
71                         * source_to_detector / (sourcetophantom + Yp)
72                         + static_cast<float>(h_detector) / 2.0);
73                     if ((ck >= 0) && (ck < h_detector) && (ci >= 0) && (ci < w_detector)) {
74                         slice.at<float>(Zt, Yt) +=
75                             c_onvolved.at<float>(h_detector - 1 - ck, w_detector - 1 - ci);
76                     }
77                 }
78             }
79         }
80     }

```

The definition of the recording geometry variables takes place in all pieces of the code in the first five lines. The OpenMP `#pragma omp parallel for` directive is placed above the first loop over the current active plane. The rest of the code is about casting, backprojecting the scaled image from projections, onto the plane of interest, that is, the mammography image reconstruction plane. The image is scaled because the point-like X-ray source produces a divergent beam, expanding as it passes the

space between the source and the detector, otherwise there will be no scaling at all.

The sample mammography data for multiple slices in YZ plane are shown on the next Figure 5. One can see that the general shape and location of the main features of the phantom are preserved but represent the somewhat severe quality degradation. Additional filtering and tomosynthesis techniques are required to fully recover the data. As expected, the data on each

part of Figure 5 was undersampled due to the physical inability of the system to record the projections behind the detector when the phantom was placed before it. The structures that lay close

enough to the detector are still satisfactory resolved. A type of ghosting artefact is visible on the last part (d) of the Figure 5 for the part of the phantom far away from the central plane.

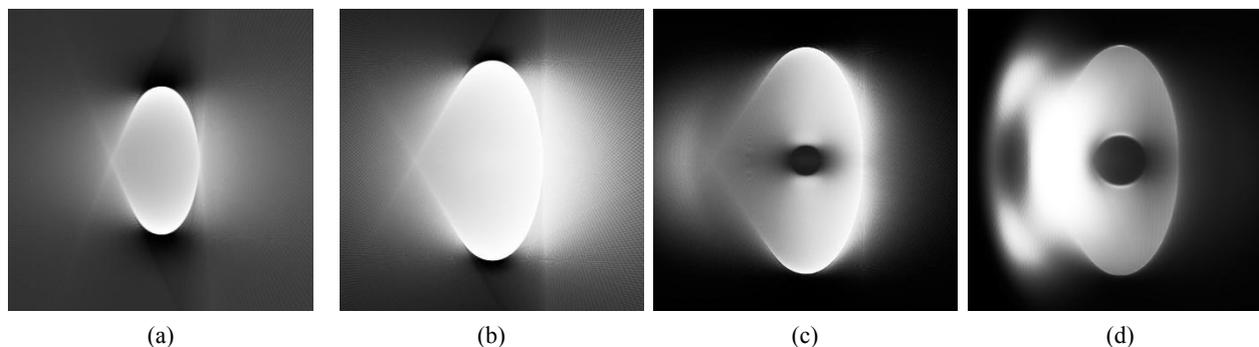


Figure 5 – Multiple slices taken in mammography mode from a digital breast phantom.

Locations of the slices are at the 200, 338, 500, and 560 voxel mark. FOV size is 675 by 675 voxels.

Number of projections used in the reconstruction are 179 evenly spaced in one degree step from -89 to +89 angle values if counted from the negative direction of the Y axis. Reconstruction of each slice takes about 60-70 seconds.

Conclusions

New additional modules transforming our computed tomography software package into fully functional mammography mode have been implemented. The reasonable quality data have been generated, although requiring further work on the contrast enhancement and artifact removal. The basic image processing has been done using the OpenCV image processing libraries compiled with the MS Visual Studio IDE compiler. The 3D reconstruction and data visualization from the set of backprojected slices are produced in Matlab. The software package could be used in combination with any real commercial mammography scanner, for the research purposes and to train medical and physics personnel.

The project is available for download at Github repository.

Acknowledgements

This part of the research was supported by the Physics and Technology Department as well as by the National Nanotechnology Lab of Open Type at al-Farabi Kazakh National University, Almaty, Republic of Kazakhstan. Originally, the work on the project was started at Adani medical equipment production plant.

References

- 1 Bushberg, Jerrold T., J. Anthony Seibert, Edwin M. Leidholdt and John M. Boone. *The essential physics of medical imaging*. Third Edition. Philadelphia: Lippincott, Williams & Wilkins, 2011.
- 2 United States Preventive Services Taskforce recommendation on breast cancer screening, at web-address: <https://www.uspreventiveservicestaskforce.org/uspstf/recommendation/breast-cancer-screening>, (accessed on February 10, 2021).
- 3 O'Connell, Avice M, and Daniel Kawakyu-O'Connor. "Dedicated Cone-beam Breast Computed Tomography and Diagnostic Mammography: Comparison of Radiation Dose, Patient Comfort, And Qualitative Review of Imaging Findings in BI-RADS 4 and 5." *Journal of Clinical Imaging Science* 2, no.1(2012):1-8.
- 4 He, Ni et al. "The utility of breast cone-beam computed tomography, ultrasound, and digital mammography for detecting malignant breast tumors: a prospective study with 212 patients." *European Journal of Radiology* 85, no.2 (2016): 392–403.
- 5 Hammerstein, Richard G. et al. "Absorbed radiation dose in mammography." *Radiology* 130, no.2 (1979): 485-91.

- 6 Wang, Tonghe et al. "Dosimetric study on learning-based cone-beam CT correction in adaptive radiation therapy." *Medical Dosimetry* 44, no.4 (2019): e71-e79.
- 7 Anthony, Marina-Portia et al. "Artifacts in Breast Magnetic Resonance Imaging." *Current Problems in Diagnostic Radiology* 45, no.4 (2015): 271-77.
- 8 Wienbeck, Susanne et al. "Artifacts Caused by Breast Tissue Markers in a Dedicated Cone-beam Breast CT in Comparison to Full-field Digital Mammography." *Academic Radiology* 24, no.7 (2016): 908-15.
- 9 Zhao, Wei et al. "A model-based scatter artifacts correction for cone beam CT." *Medical Physics* 43, no.4 (2016): 1736-53.
- 10 Walz-Flannigan, Alisa et al. "Artifacts in Digital Radiography." *American Journal of Roentgenology* 198, no.1 (2012): 156-61.
- 11 di Franco, Francesco et al. "GEANT4 Monte Carlo simulations for virtual clinical trials in breast X-ray imaging: Proof of concept." *Physica Medica* 74 (2020): 133-42.
- 12 Bliznakova, Kristina. "The advent of anthropomorphic three-dimensional breast phantoms for X-ray imaging." *Physica Medica* 79 (2020): 145-61.
- 13 McGarry, Conor K et al. "Tissue mimicking materials for imaging and therapy phantoms." *Physics in Medicine & Biology* 65 (2020): 23TR01.
- 14 Sechopoulos, Ioannis et al. "Characterization of the homogeneous tissue mixture approximation in breast imaging dosimetry." *Medical Physics* 39, no.8 (2012): 5050-9.
- 15 Glick, Stephen J, and Lynda C Ikejimba. "Advances in digital and physical anthropomorphic breast phantoms for x-ray imaging." *Medical Physics* 45, no.10 (2018): e870-e885.
- 16 Malliori, Anthi et al. "Development of Physical Breast Phantoms for X-ray Imaging Employing 3D Printing Techniques." *The Open Medical Imaging Journal* 12, no.1 (2020): 1-10.
- 17 Kussainov, Arman and Nurzhan Saduev. "Homemade computed tomography setup with FDK reconstruction software." *Journal of Physics: Conference Series* 1391 (2019): P.012086.
- 18 Nelson, David L., Michael M. Cox. *Lehninger Principles of Biochemistry*. Eight Edition. New York: Worth Publishing, 2021.
- 19 Seltzer, Stephen M. "Calculation of Photon Mass Energy-Transfer and Mass Energy-Absorption Coefficients." *Radiation Research* 136, no.2 (1993): 147-70.
- 20 Sarno, Antonio et al. "Homogeneous vs. patient specific breast models for Monte Carlo evaluation of mean glandular dose in mammography." *Physica Medica* 51 (2018): 56-63.
- 21 Mettivier, Giovanni et al. "Evaluation of the Breast Simulator software platform for breast tomography." *Physics in Medicine & Biology* 62, no.16 (2017): 6446-66.
- 22 Dance, David et al. "Breast dosimetry using high-resolution voxel phantoms." *Radiation Protection Dosimetry* 144, no.1-3 (2005):359-63.
- 23 Bakic, Predrag R et al. "Maidment Mammogram synthesis using a three-dimensional simulation. III. Modeling and evaluation of the breast ductal." *Medical Physics* 30, no.7 (2003): 1914-25.
- 24 Sarno, Antonio et al. "Imaging performance of phase-contrast breast computed tomography synchrotron radiation and a CdTe photon-counting detector." *Physics in Medicine & Biology* 32, no.5 (2016): 681-90.
- 25 Kak, Avinash C., Malcolm Slaney, *Principles of Computerized Tomographic Imaging*. New York: IEEE Press, 1999.
- 26 Punnoose, Jacob et al. "Technical Note: spektr 3.0-A computational tool for x-ray spectrum modeling and analysis." *Medical Physics* 43, no.8 (2016): 4711-17.