XAP-Lab: A software tool for designing flexible X-ray acquisition protocols

A. Martínez[1,2], A. García-Santos[1,2], N. Ballesteros[2], M. Desco[1,2,3,4], M. Abella[1,2,3]

[1] Departamento de Bioingeniería e Ingeniería Aeroespacial, Universidad Carlos III de Madrid, Spain

[2]Instituto de Investigación Sanitaria Gregorio Marañón, Madrid

[3]Centro Nacional Investigaciones Cardiovasculares (CNIC), Spain

[4]Centro de investigación en red en salud mental (CIBERSAM), Spain

Corresponding author: Manuel Desco

Email: desco@hggm.es, mabella@hggm.es

Phone: 916248194

Postal address:

Departamento de Bioingeniería e Ingeniería Aeroespacial

Universidad Carlos III de Madrid

Avda. de la Universidad 30
28911 Leganés
Madrid
Spain

Abstract

Background and Objective

The availability of digital X-ray detectors, together with the development of new robotized hardware and reconstruction algorithms, opens the opportunity to provide 3D capabilities with conventional radiology systems. This would be based on the acquisition of a limited number of projections with non-standard geometrical configurations. The versatility of these techniques is enormous, enabling the introduction of tomography in situations where a CT system is hardly available, such as during surgery or in an ICU, or in which a reduction of radiation dose is key, as in pediatrics. Computer simulations are a valuable tool to explore these possibilities before their actual implementation on real systems. Existing software tools generally simulate only standard acquisition protocols, such as cone-beam with circular trajectory, thus not allowing the users to evaluate more sophisticated projection geometries. The goal of this work is to design a simulation tool that enables the design of acquisition protocols with flexible projection geometries.

Methods

We present XAP-Lab, a software tool for the design of X-ray acquisition protocols with flexible trajectories. For a given projection geometry, defined through a graphical user interface, it allows the user to simulate projections using GPU-accelerated kernels, the visualization of the scanned field of view and the estimation of the total radiation dose. The complete acquisition protocol can then be exported with the appropriate format for its use on real systems.

We tested the software by optimizing a tomosynthesis protocol and validating the results with real acquisitions using a SEDECAL NOVA FA radiography system and phantoms for quantitative and qualitative evaluation.

Results

Quantitative evaluation using a phantom showed a mean error under 4 mm for each position, below the ±5 mm tolerance of the system specified by the manufacturer. Visual evaluation on a thorax acquisition also showed a good geometrical agreement between simulated and real projections.

Conclusions

Results showed an excellent matching with simulations, supporting the usefulness of XAP-Lab for the design of new acquisition protocols with non-standard geometries.

Keywords: Simulation; X-ray; Acquisition protocol; Radiology system; Tomography; System geometry.

1. Introduction

The past decades have seen a rapid evolution towards the use of digital equipment in radiology. The introduction of digital detectors, together with a flexible robotized movement of the system components opened the possibility of obtaining 3D information from conventional X-ray systems, using particular geometrical configurations. This may allow radiologists to introduce tomography in situations where a CT system is not available, such as during surgery or in an ICU, while keeping the radiation dose delivered to the patient as low as possible.

In this context, computer simulations are a valuable tool to explore the possibilities of new acquisition strategies before their implementation on real systems. Several simulation tools that model the X-ray image acquisition [1-6] of 2D images or tomographic studies have been developed in recent years. Although they generally include high performance projection kernels and allow users to configure the acquisition geometry, they are limited by restrictions regarding source and detector positioning, and/or are not intended for simulating a sequence of positions, which reduce their ability to simulate new acquisition protocols based on not-standard setups. The ImaSim package [1] only supports planar and CT/CBCT geometries; the tools described in [2, 3] allow adjusting the source position and orientation relative to the imaged object and the source-to-detector distance (SOD), but the detector orientation is fixed to be orthogonal to the beam. CATSIM [6] was focused on modelling the physics of CT acquisition (realistic quantum and electronic noise models, finite focal spot size and shape, finite detector cell size, detector cross-talk, detector lag or afterglow, bowtie filtration, finite detector efficiency, non-linear partial volume and scatter) but it did not include the simulation of non-standard acquisition geometries. The tool in [4] allows setting the position and orientation of source and detector independently, but it is aimed at the simulation of synthetic objects that need to be defined using computer-aided drawing (CAD) models. The CONRAD software framework [5] and ASTRA toolbox [7] allow configuring flexible acquisition geometries and high-performance simulation methods, but are not prepared to export the geometry to real systems out of the box and, in the case of ASTRA toolbox, lack a graphical user interface to facilitate the geometry design process.

In this work, we present XAP-Lab, a tool that enables the design of acquisition protocols with flexible projection geometries adapted to specific system configurations, providing the simulated projections in near real-time as well as additional information such as the scanned field of view (FOV) or the total radiation dose.

2. Simulation tool

The main components of an X-ray system are the X-ray source, the detector, and the sample holder (or bed). As all of them have specific constraints in its range of movement, the achievable positions of these elements can be defined by a set of geometrical parameters. A given arrangement of X-ray source and detector defines a projection position; the complete trajectory of the acquisition protocol will be given by a set of projection positions. The acquisition protocol is completed with the parameters of the source (mAs and KV).

XAP-Lab, developed in MATLAB 2016b following a modular design to facilitate the addition of new functionalities, enables the design of acquisition protocols for generic or specific X-ray systems. It is operated through the graphical user interface (GUI) shown in Fig. 1.
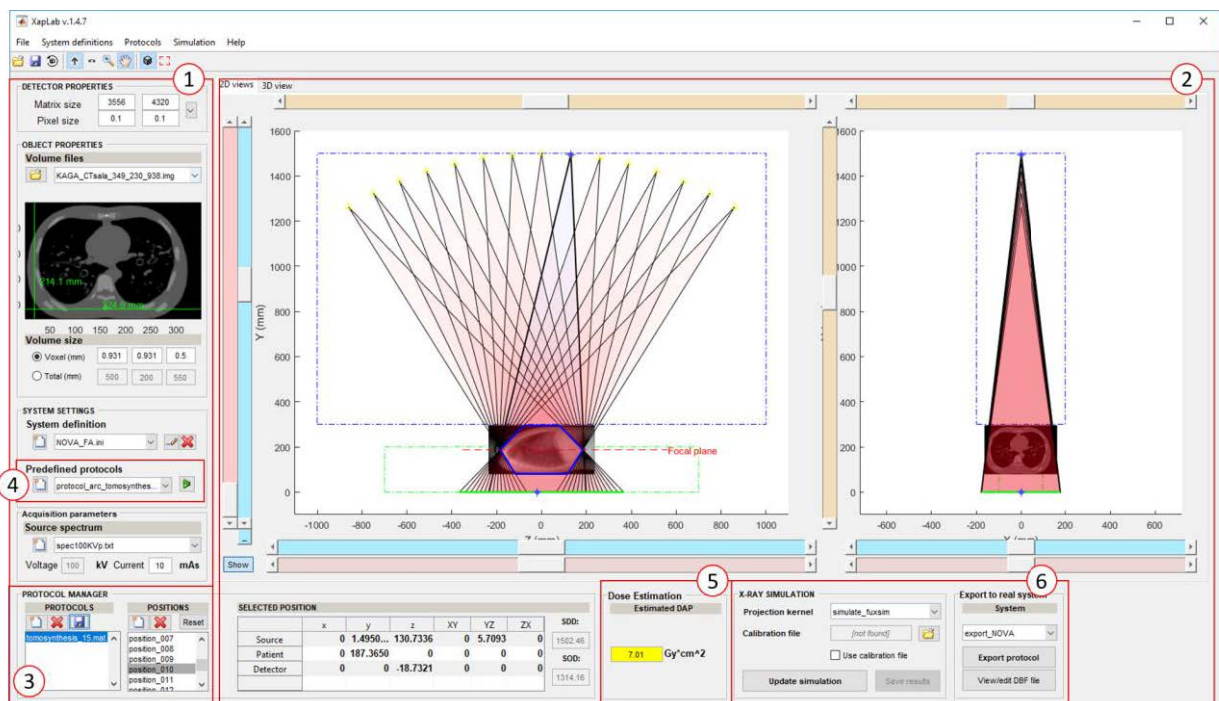

Fig. 1. Graphical user interface of the software, with different panels: detector and sample volume settings (1), geometry creation (2), protocol manager (3), predefined protocols (4), dose estimation (5), and simulation and system export modules (6).

Fig. 2 shows the workflow to create a new acquisition protocol with XAP-Lab.
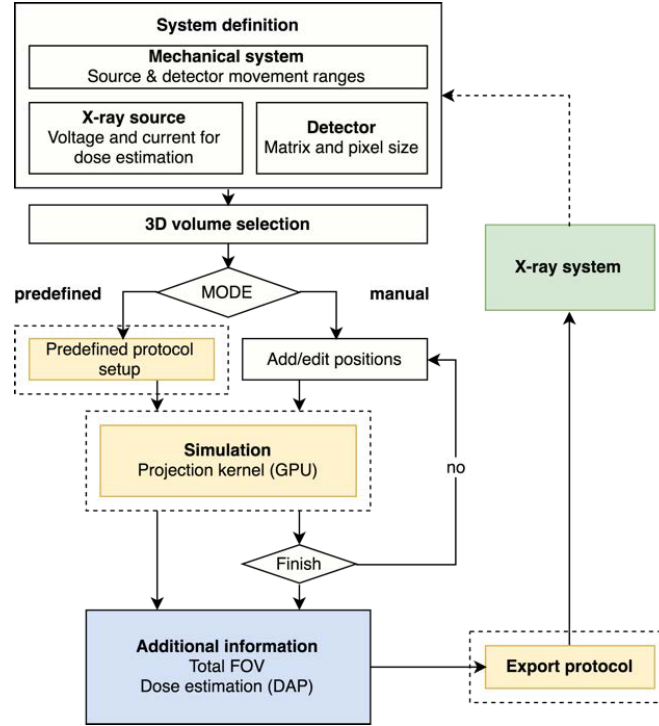
Fig. 2. Workflow of new protocol creation.

The first step is to establish a set of rules and constraints that define the physical arrangement of the three main elements, X-ray source, detector and sample holder for a given X-ray system. This allows the software to define the working space for the operation of the X-ray system, to facilitate the design of protocols suitable for that system. This information is stored in the *system definition file*, a plain text file which specifies the minimum, maximum and default values for each geometrical parameter of the three elements.

The sample is defined as an input volume of density values, which can be either real data previously acquired in a CT or generated by computer simulation. The acquisition protocol is designed interactively by adding individual projection positions. The GUI displays all positions in the trajectory in the *positions* list on panel 3 of Fig. 1, which can be selected to be edited or deleted.

The placement of the source and detector for each projection position can be done manually editing each one of the positions by using the sliders, fine-tuning the individual values through the table or dragging and dropping the graphic markers shown for the *source* and *detector* on each view in panel 2 of Fig. 1. The total scanned FOV, understood as the part of the sample covered by all acquisition positions, is also shown in the 2D views of panel 2 of the interface as semi-transparent polygons to help assessing if the designed protocol sufficiently covers the region of interest (ROI).

To generate projection data, the attenuation of X-rays traversing a sample is modeled as:

$$I = \sum_k I_0(E_k) \cdot e^{-\sum_i mac_i(E_k) \cdot \int_L \rho_i dl} \qquad (1)$$

where $I_0(E_k)$ is the X-ray beam intensity at each energy E_k (energy spectrum) and mac_i and ρ_i are the mass attenuation coefficient and density of material *i* respectively. The energy spectra were generated with the Spektr [8] toolbox using mass attenuation coefficients of different compounds obtained from the NIST database [9]. The density map for each tissue, ρ_i, is generated from the input volume of the sample by thresholding, setting all the voxels corresponding to other tissues to 0 (the sum of all density maps equals the original volume). The threshold values to separate each tissue/material are provided in a text file along with the volume data.

For each tissue/material in the sample, the values of $\int_L \rho_i dl$ in equation 1 are generated by means of FUX-Sim [10], a fast projection/back-projection framework optimized for different families of GPUs and CPUs (CUDA & OpenCL). To this end, we need to translate the positions from the coordinate system used in XAP-Lab to that used by FUX-Sim. Left panel in Fig. 3 shows the coordinate system used to represent the main elements involved in the image acquisition process: the x-ray source, the detector and the sample. The geometry expressed in XAP-Lab's coordinate system (x^s, y^s, z^s) is translated onto the object coordinate system (x^p, y^p, z^p) used by the projection kernel of FUX-Sim.

FUX-Sim defines a virtual detector placed orthogonally to the line that passes through the source and the origin. The intersection of this line with the virtual detector's plane, p_{vd}, corresponds to the center of the detector. The real detector position in the object coordinate system is then expressed as a set of linear displacements and angular tilts applied to a virtual detector position, as shown in the right panel of Fig. 3. The linear displacement, Δz, is first calculated so that the source-object line passes through the center of the detector, coincident with the p_{vd} point of the virtual detector. The tilting, φ_{yz}, is then calculated as the angle formed between the real and virtual detectors.
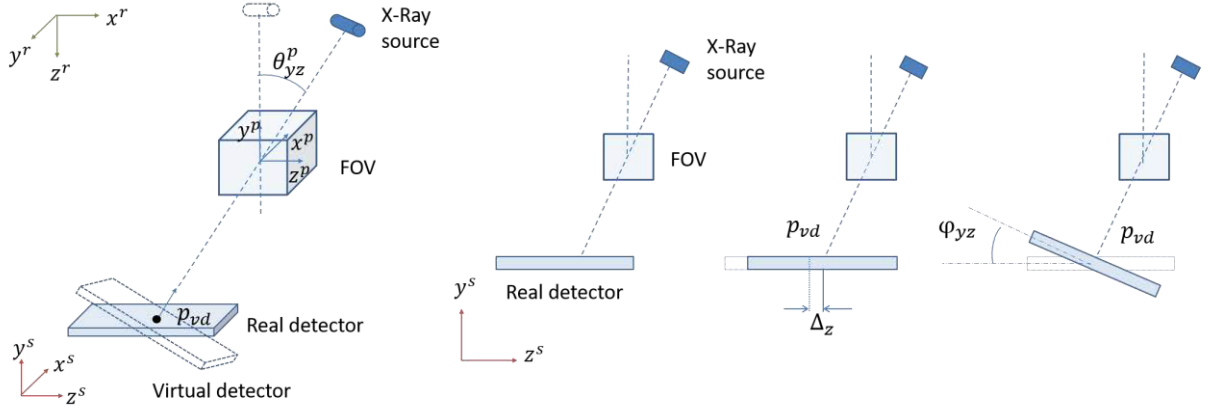
Fig. 3. Left: Coordinate system for XAP-Lab (x^s, y^s, z^s), FUX- Sim (x^p, y^p, z^p) and the real system (x^r, y^r, z^r).

Right: Calculation of the detector position in the object coordinate system.

The final projection data are stored as

$$Proj = -log\left(\frac{I}{\sum_k I_0(E_k)}\right) \tag{2}$$

To assess the tradeoff between image quality and radiation dose, XAP-Lab makes use of the Dose Area Product (DAP), a widely used metric to assess the radiation risk [11], calculated as

$$DAP(mGy \cdot cm^2) = n_0 \cdot C \cdot \left(\frac{KV}{80}\right)^\alpha \cdot \left(\frac{100}{d}\right)^2 \cdot T \tag{3}$$

where n_0 corresponds to the reference throughput, C represents the tube current (mAs), d is the source-to-detector distance in cm, T is the field size in cm^2, and α characterizes the dependence of the throughput with kilovoltage, obtained in a previous calibration step (by measuring the throughput for different KVp values and performing a linear regression). The reference throughput is measured at 80 kVp and a source-to-detector distance of 100 cm according to

$$n_0 = \frac{D_{80kV,100cm}(mGy)}{C_0(mAs)} \tag{4}$$

The sum of the DAP values for each *position* (that reflect the total dose) is displayed in real time in the panel 5 of the GUI (Fig. 1).

Finally, if the real mechanical system allows programming the acquisition geometry, and exposes the coordinate system it uses, an export module can be used to translate the complete simulated trajectory into the corresponding parameters in the real system. For this task, we define *system export* modules that translate the coordinates of each element in XAP-Lab (x^s, y^s, z^s) to the specific coordinate system

(x^r, y^r, z^r) and save the resulting configuration to a file with the appropriate format to be read by the physical system, or to a readable text file if the mechanical system requires manual input by the user. Given the high variability between different acquisition systems, these modules have to be customized *ad hoc* and are only possible if the system adequately documents how to adjust their geometry, either by configuration files or user interfaces.

Software architecture

The software implementation follows an object-oriented approach to represent the main components of the X-ray system and the relationships between them. All classes inherit from MATLAB's *handle* class to allow an easy reference to the existing objects, and to communicate through events to update the system state when any element changes its size and/or position. The UML class diagram is shown in Fig. 4.
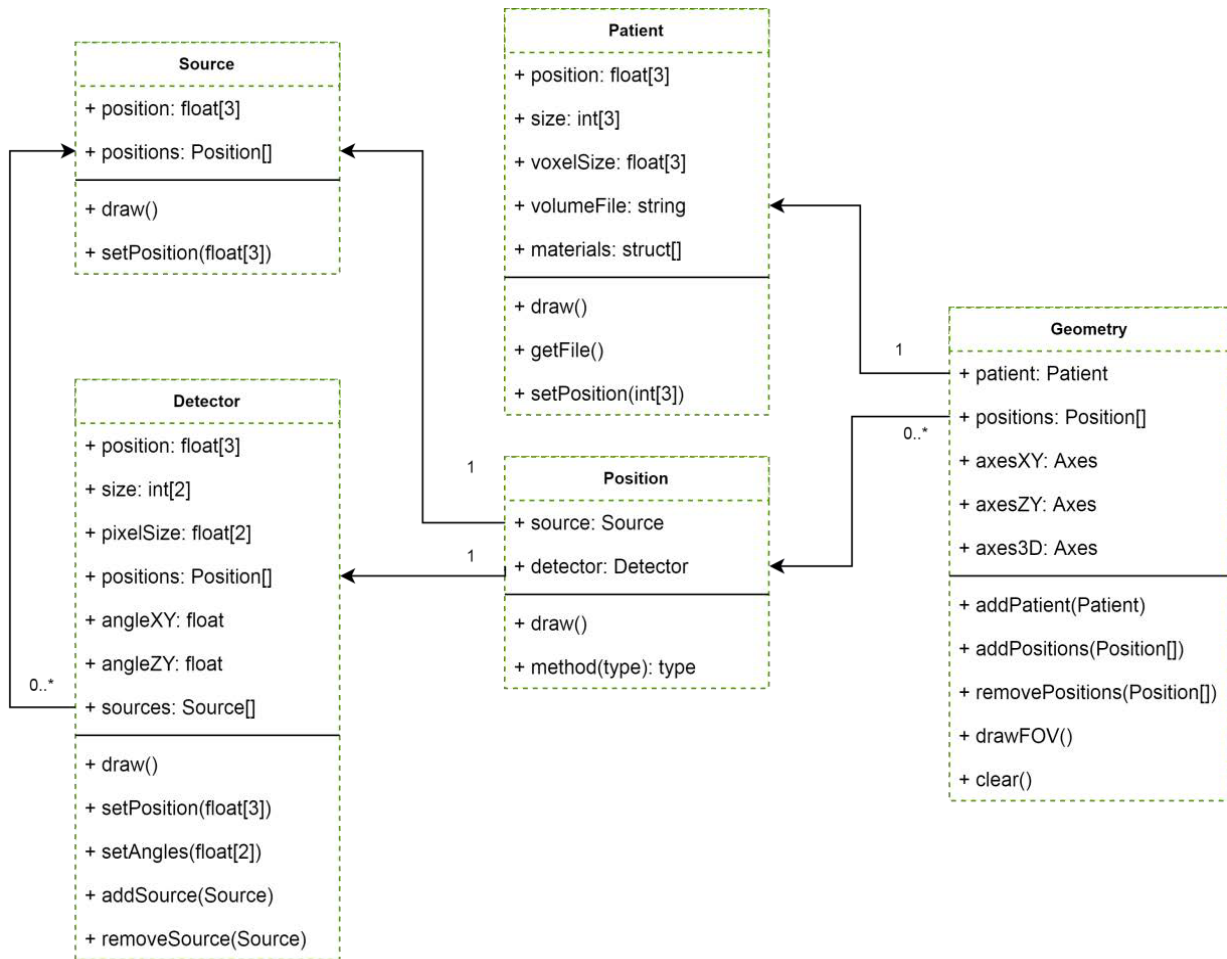

Fig. 4. UML class diagram for the source, detector, patient, position and geometry classes.

The *source* class describes the position of the X-ray source. The *detector* class specifies the position and orientation of the detector, as well as its matrix size and pixel size. It also includes references to one or several X-ray source objects. The *position* class associates a *source* to a *detector*, describing the

geometry of a single projection. Adding a new projection position instantiates a new pair of *source* and *detector* objects in their default positions.

The *patient* class represents the sample to be imaged, and contains its spatial position in the system, matrix size and voxel size. If no additional properties are set, the field of view is calculated from the matrix and voxel size. In order to enable simulating polychromatic projections, the *patient* class may provide additional properties and methods to define its constituting materials. The *volumeFile* property includes the path to a file that stores the density map of the patient. It can be accessed by calling the *getFile()* method, which displays a file selection dialog. The *materials* property stores an array of key-value pairs that define the thresholds for each tissue/material and the path to a text file containing the mass attenuation coefficients for that material.

Additionally, the *source*, *detector* and *patient* classes implement a *draw()* method that represents each object in all available views of the GUI (panel 2 of Fig. 1). The source position is represented by a small yellow triangle, the patient is represented as a box and the detector is represented as a line in each of the 2D views. The X-ray beam is represented as a semitransparent triangle with its vertices on the source position and the detector corners.

Finally, the *geometry* class represents the whole acquisition protocol, and comprises a *patient* object and an array of *position* objects. The *geometry* class also overloads the implementation of some generic MATLAB functions such as *disp()* or *struct()* to improve its representation and to ease the saving/loading of XAP-Lab's geometry data.

Acquisition protocols with predefined trajectories

The design of acquisition protocols comprising a large number of *positions* can be time consuming if each one of the positions needs to be manually set. To facilitate this task, XAP-Lab includes predefined geometries, which automatically generate all the projections positions (trajectory) based on a set of input parameters. Predefined trajectories include circular scan, helical scan, two wide field of view trajectories (tilting or linear source movement), and two tomosynthesis trajectories (tilting or linear source movement. More details on these acquisition trajectories can be found in [10].

The use of these predefined protocols is based on a template function that specifies the interface for the input/output data. As input, it uses an initial *geometry* object with one *position*, which serves as a starting point for the predefined protocol. Additional parameters such as the number of positions or the source to detector distance are requested at the beginning of the function through additional dialogs. Then the code implements the logic to create the subsequent *positions* based on the protocol configuration, modifying the initial *geometry* object. In addition to the *geometry* object, the function returns the *param* struct

containing the protocol parameters and the *output* struct with any relevant information about the created protocol in text format, as well as additional graphic elements to be added to the main GUI's views.

The top panel of Fig. 5 shows an example of a predefined protocol for tomosynthesis that generates *positions* with the source and detector moving along the *z* axis according to the following equations.

$$s_{x,y,z}(i) = \left(0, s_{y_0}, \frac{2 \cdot s_{z_{MAX}}}{N_p} \cdot \left(i - \frac{N_p}{2} \right) \right) \tag{5}$$

$$d_{x,y,z}(i) = \left(0, d_{y_0}, -s_z(i) \cdot \frac{p_{y_0} - d_{y_0+FP}}{s_{s_0} - p_{y_0-FP}} \right) \tag{6}$$

where $s_{x,y,z}$, $d_{x,y,z}$ and $p_{x,y,z}$ are the positions of the *source*, *detector* and *patient* objects for each position *i* and $s_{z_{MAX}}$ is the maximum displacement of the source in the *z* axis indicated in the *system definition* file (for simplification, we assume that the source *z* limits are symmetric so $|s_{z_{MAX}} - s_{z_{MIN}}| = 2 \cdot s_{z_{MAX}}$). The parameters s_{y_0} and d_{y_0} are respectively the source and detector vertical coordinates of the first *position* in the input *geometry* object. The number of projections, N_p and the distance of the focal plane to the center of the object, *FP*, are set through the dialog shown in the top-right panel of Fig. 5. Bottom panel of Fig. 5 shows another predefined tomosynthesis protocol that allows the user to draw an ROI on a sagittal slice of the volume and generates the positions that ensure the whole ROI to be properly illuminated in all the positions.
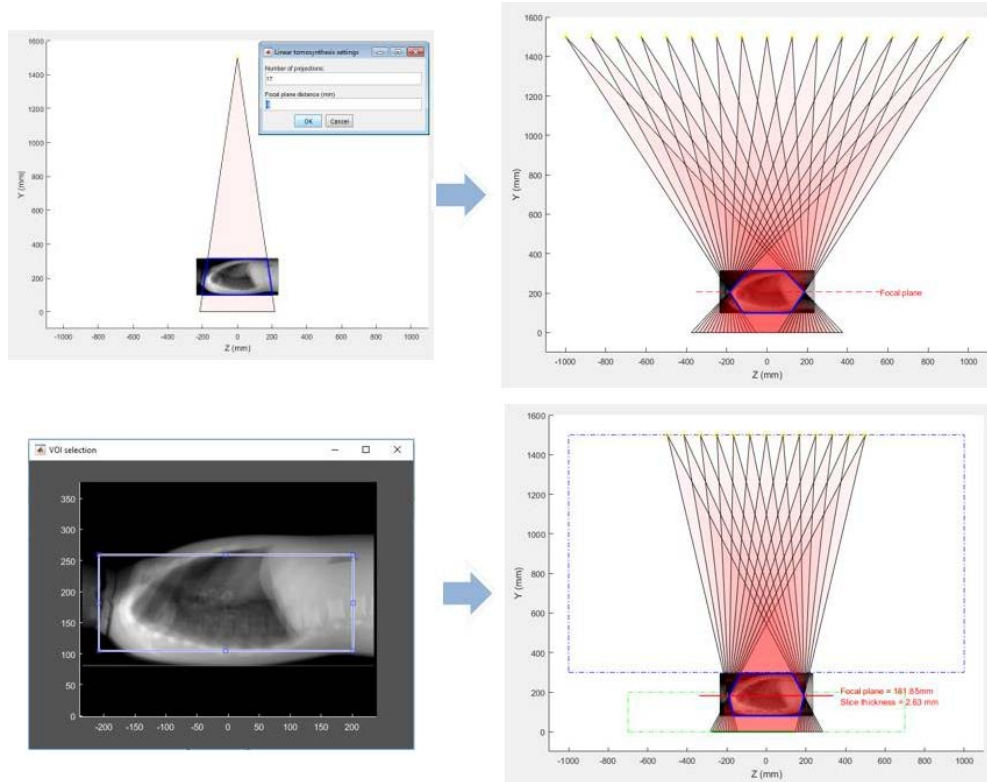
Fig. 5. Two predefined protocol examples showing the windows for input automatic parameters (left) and the resulting positions generated by the predefined protocol (right). Top: Tomosynthesis protocol for a given number of projections. Bottom: Tomosynthesis protocol forcing the scanned FOV to contain a specific ROI of the patient.

XAP-Lab provides a straightforward way to create new predefined protocols based on this template through a button (panel 4 of Fig. 1) that opens a dialog window asking for a descriptive name for the protocol and stores a new .m file based on this template, which can be edited to implement the desired protocol following the guidelines explained above. All stored protocols are automatically detected and made available through the main GUI.

In addition to predefined protocols, which involve writing MATLAB code based on XAP-Lab's geometry classes, there exists the option to directly import the geometry data from a plain text file written in JSON format. These files contain an array of acquisition positions, specifying all relevant parameters (i.e. source/detector position and orientation) and are decoded by XAP-Lab to set up the desired geometry.

The example below shows the contents of a file for a protocol with two positions.

```
{
  "detector":{
    "size":[512, 512],
    "pixelSize":[0.5,0.5]
  },
  "patient":{
    "position":[0,300,0]
  },
  "positions":[
    {
      "source":{
        "position":[0,1000,-200]
```

```
      },
      "detector":{
        "position":[0,0,50],
        "angle":[0,0,0]
      }
    },
    {
      "source":{
        "position":[0,1000,200]
      },
      "detector":{
        "position":[0,0,-50],
        "angle":[0,0,0]
      }
    }
  ]
}
```

Geometrical misalignments

Another useful feature of a simulation system is to enable studying the effects of mechanical misalignments in a real system. We have included in XAP-Lab the ability to select (panel 6 of Fig. 1) a *calibration file* containing the values of these positioning errors, provided that the simulation module in use supports taking into account misalignments. This is the case of the FUX-Sim framework used by XAP-Lab to simulate the images, as it allows specifying the deviations from the ideal position of the source and the detector for each projection. FUX-Sim read these values from plain text files where rows correspond to positions and columns represent the deviation from the ideal value for each parameter (projection angle, source-to-object distance, detector-to object distance, detector orientation and shifts).

3. Evaluation

System setup

We evaluated the XAP-Lab using a SEDECAL NOVA FA digital radiography system, consisting of a ceiling suspension and an elevating table which can be remotely controlled. The images were acquired with a Perkin Elmer XRpad 4336 flat panel detector with a pixel size of 100 μm and a matrix size of 4320×3556 pixels.

We created the *system definition* file (as defined above) that represents the movement ranges of the system's suspension and detector bucky. Then we implemented an export module to translate the simulated geometries onto the NOVA FA format. The resulting acquisition trajectories specifies the individual positions with the longitudinal position of the detector (*z* axis in XAP-Lab's coordinate system) and the position of the source relative to the center of the detector (source-to-detector distance and α angle in left panel of Fig. 6).

The equations in the left diagram:

$$SDD = \sqrt{(y_0^s)^2 + (z_0^s - z_0^d)^2}$$

$$\alpha = \operatorname{atan}\left(\frac{z_0^s - z_0^d}{y_0^s}\right)$$

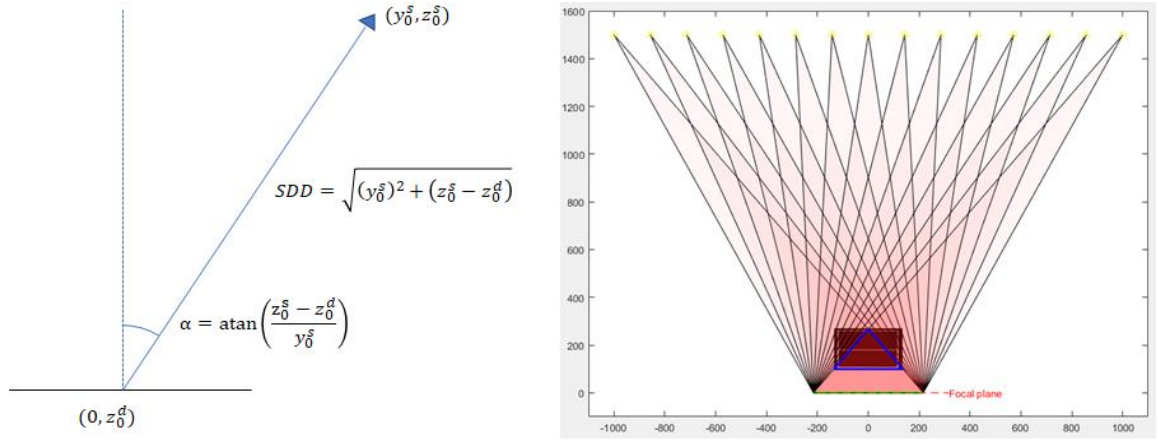with points (y_0^s, z_0^s) and $(0, z_0^d)$.

Fig. 6. Left: Diagram of the conversion from XAP-Lab's coordinate system to NOVA FA position. Right: Diagram representing the 15 positions of the designed tomosynthesis protocol. The blue polygon outlines the total scanned FOV.

A linear tomosynthesis protocol with 15 *positions* was generated following Eqs. (1) and (2), from an initial *geometry* given by the central *position*, with a source-to-detector distance of 1500 mm, a maximum source displacement on the *z* axis of 1000 mm, and a focal distance of -160.3 mm, so that the focal plane was at the detector's height. These parameters resulted in a protocol with the X-ray tube following a linear trajectory and the detector staying at a fixed position. Right panel of Fig. 6 shows the described acquisition protocol as displayed in the ZY view of the GUI.

Finally, we exported the designed protocol to the NOVA FA system format in order to implement the real acquisition.

Geometrical evaluation

In order to allow the quantitative evaluation of the geometrical correspondence between the simulated and real projections, we created a geometric phantom made of methacrylate plus 1.5 mm diameter radiopaque spherical markers embedded at the corners (Fig. 7). We acquired a CT study of the phantom using a Toshiba Aquilion/LB scanner, obtaining a volume with a matrix size of 512×512×1645 voxels and a voxel size of 0.931×0.931×0.5 mm, which was used as the input patient in FUX-Sim.
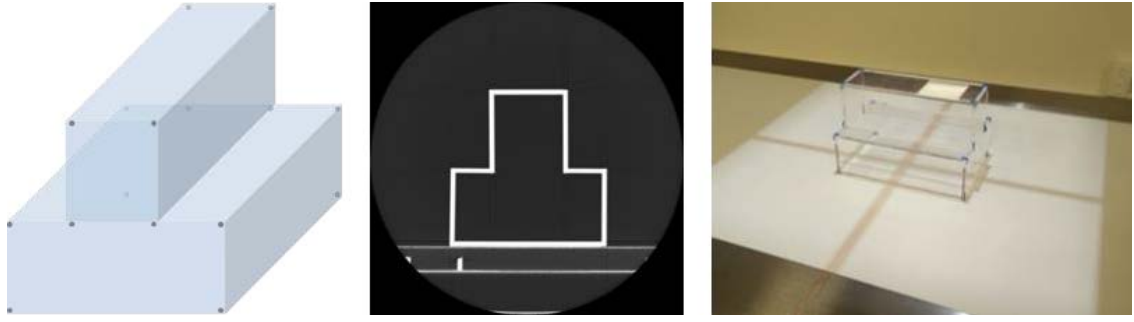
Fig. 7. Geometrical phantom design (left), axial CT slice (center) and photograph of the phantom placed on the NOVA FA bed for acquisition (right).

The simulated projection generated using the CT volume as an input were compared to those acquired with the real system, in terms of position errors of the radiopaque markers. Considering the different phantom position in the Toshiba CT and the NOVA FA system, the CT data was modified to match the position in the NOVA FA system, following the process illustrated in Fig. 8. We first performed a rigid 2D registration of the central simulated projection with the corresponding projection acquired with the NOVA FA. Assuming that misalignments in the *y* axis were negligible, the resulting 2D transformation matrix would be enough to correct misplacements in the *x* and *z* axes. This transformation matrix was applied to the coronal slices of the original CT volume (those parallel to the table) to obtain the registered volume.
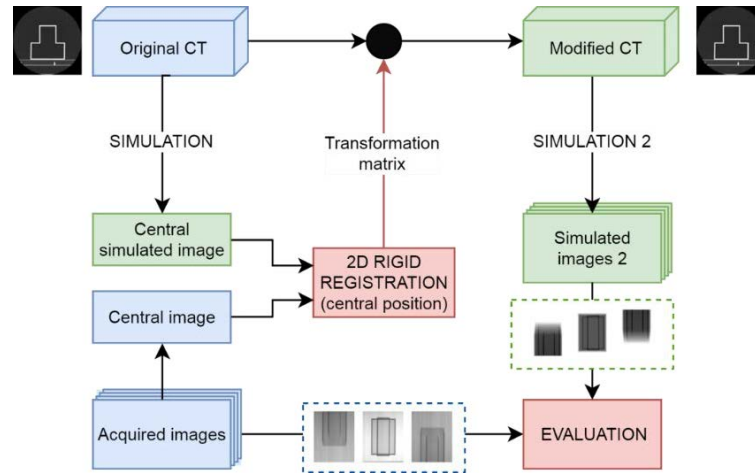

Fig. 8. Workflow of the process followed to modify the CT in order to match its position with that of the real phantom in the X-ray system.

To measure the disagreement in the position of the markers in the real and simulated projection, we used a MATLAB script to automatically locate and match the centroids of the markers in each simulation/projection pair, followed by a manual correction in those cases that failed due to markers overlapping. The error was calculated as the Euclidean distance between each pair of markers.

Qualitative evaluation

A qualitative evaluation was done using the same tomosynthesis protocol using the life-size human thorax phantom PBU-60 shown in Fig. 9 manufactured by Kyoto Kagaku, also acquired previously with a Toshiba Aquilion/LB (512×512×1645 voxels, 0.931×0.931×0.5 mm voxel size).

Fig. 9. PBU-60 antropomorphic phantom (left) placed on the system's bed. CT slice of the phantom (right).

Results

The top panel of Fig. 10 shows the marker centroids for the simulated and real data overlaid on the real projections for acquisition positions 1, 8 and 15. The bottom panel shows the mean distance between all markers at each of the 15 *position*s of the protocol. The mean error was under 4 mm in all cases, with no dependency with the position along the *x* axis and with a linear dependency along the *y* axis which is the direction of the source movement.
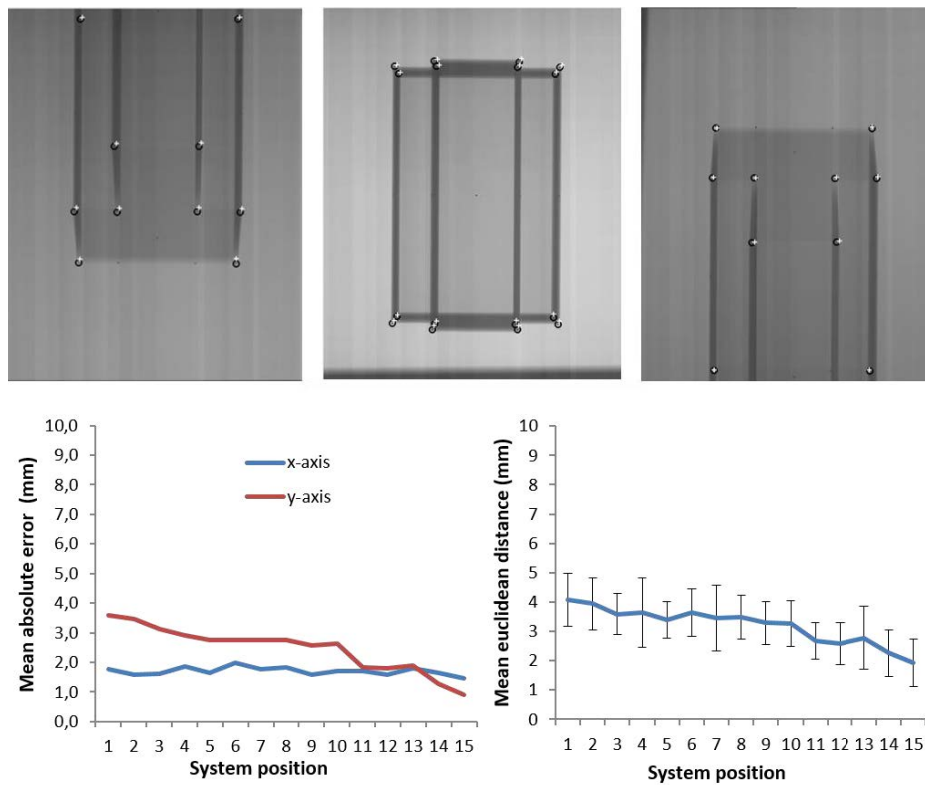
As an example, Fig. 11 shows the real and simulated projections for *positions* 1, 8 and 15. Size measurements of some structures show a good visual agreement between the simulated and real projections, with measured errors below 1%.
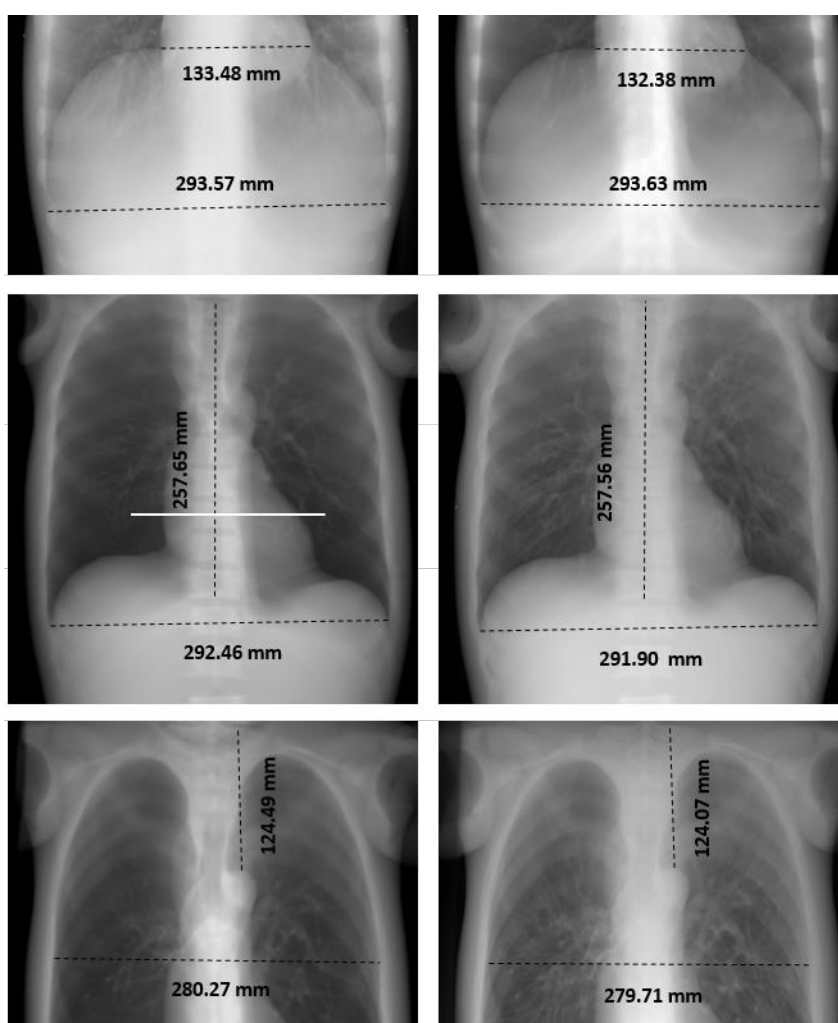
Fig. 11. Real (left) and simulated (right) projections at positions 1, 8 and 15 of the linear tomosynthesis protocol. Dashed lines show size measurements of approximately corresponding structures. White line in the middle plane shows the profile taken across the heart. .

The profile taken across the heart in the middle projection (white line in Fig. 11) confirms the good structure matching between simulated and real data (Fig. 12). Given that the projection pixel values depend on many uncertainties such as the X-ray spectrum, the detector model, scatter, and other physical effects, not taken into account in this study, the coincidence in pixel values was not expected,
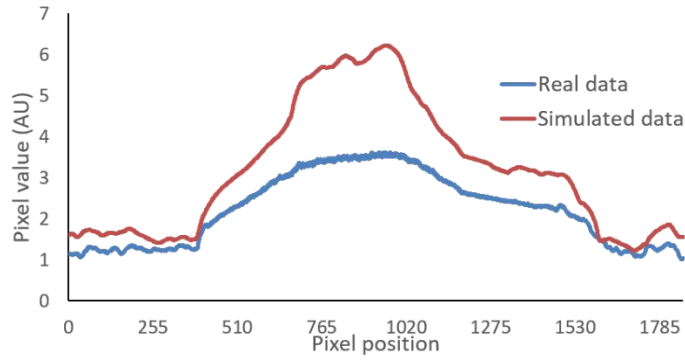
Fig. 12. Horizontal profile taken across the heart on simulated and real data.

4. Discussion

We have presented XAP-Lab, a software tool to support the design of X-ray acquisition protocols. Through a graphical user interface, it allows users to define flexible projection geometries taking into account the constraints of a given X-ray system configuration. The software provides additional information regarding the designed protocol, such as the total scanned field of view and the estimation of the total radiation dose in terms of the dose-area-product. The protocol can be simulated and previewed in near real time thanks to the GPU implementation of projection kernels that take into account the geometry of the system and the polychromatic nature of the spectra. FUX-Sim does not include scatter estimates, but thanks to its modular implementation, it would be straightforward to plug in other external software to provide such information.

As a preliminary evaluation, we used the software to design a tomosynthesis protocol that was afterwards exported to be acquired with a real system. Results showed good agreement between simulated and acquired images. Quantitative evaluation showed a mean error under 4 mm for each position, which is below the ±5 mm tolerance specified by the manufacturer for the SEDECAL NOVA FA system . The visual examination of the thorax acquisition also shows a good geometrical agreement between simulated and real projections. Coincidence in pixel values and contrast was not expected, as it is affected by many uncertainties such as the X-ray spectrum, the detector model, scatter, and other physical effects. However, XAP-Lab was designed to enable an easy introduction of such features through plugins that act as an interface between XAP-Lab's geometry definition and other simulation software packages. As a future improvement, we plan to add support for the publicly available software library MC-GPU [12] to provide scatter estimations, and also for the detector model as another part of the simulation

pipeline. The validation of XAP-Lab on reconstructed images is hindered by the need of a complete geometrical calibration of the real system, which was out of the scope of this work.

The level of precision achieved supports the use of XAP-Lab to assess the practical feasibility of different acquisition schemes on a given real system, as well as to define the optimal system geometrical requirements in terms of movement ranges of the source and the detector for a given application. While the range of movement of the elements in the real system is roughly specified in the current implementation, further requirements like specific trajectories for each element, could be enforced in the *system export* modules to ensure that each position can be reproduced on the system. The modular approach followed in the design and implementation of this tool enables an easy integration of new modules and features, such as new predefined protocols or support for additional acquisition systems.

Acknowledgement

References

[1]   G. Landry, F. deBlois, and F. Verhaegen, "ImaSim, a software tool for basic education of medical x-ray imaging in radiotherapy and radiology," *Front Phys,* vol. 1, 2013.

[2]   V. Fanti, R. Marzeddu, G. Massazza, and P. Randaccio, "A simulation tool to support teaching and learning the operation of X-ray imaging systems," *Medical Engineering & Physics,* vol. 27, pp. 555-559, 2005.

[3]   E. Gallio, O. Rampado, E. Gianaria, S. D. Bianchi, and R. Ropolo, "A GPU Simulation Tool for Training and Optimisation in 2D Digital X-Ray Imaging," *Plos One,* vol. 10, p. e0141497, 2015.

[4]   P. Duvauchelle, N. Freud, V. Kaftandjian, and D. Babot, "A computer code to simulate X-ray imaging techniques," *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms,* vol. 170, pp. 245-258, 2000.

[5]   A. Maier, H. G. Hofmann, M. Berger, P. Fischer, C. Schwemmer, H. Wu, *et al.*, "CONRAD--a software framework for cone-beam imaging in radiology," *Med Phys,* vol. 40, p. 111914, 2013.

[6]   B. De Man, S. Samit Basu, N. Chandra, B. Dunham, P. Edic, M. Latrou, *et al.*, "CATSIM : a new Computer Assisted Tomography SIMulation environment," *Proc. SPIE 6510, Medical Imaging 2007: Physics of Medical Imaging, 65102G,* 2007.

[7]   W. van Aarle, W. J. Palenstijn, J. De Beenhouwer, T. Altantzis, S. Bals, K. J. Batenburg, *et al.*, "The ASTRA Toolbox: A platform for advanced algorithm development in electron tomography," *W. van Aarle, W. J. Palenstijn, J. De Beenhouwer, T. Altantzis, S. Bals, K. J. Batenburg, and J. Sijbers,*

"The ASTRA Toolbox: A platform for advanced algorithm development in electron tomography", Ultramicroscopy (2015), vol. 157, pp. 35-47, 2015.

[8] J. H. Siewerdsen, A. M. Waese, D. J. Moseley, S. Richard, and D. A. Jaffray, "Spektr: A computational tool for x-ray spectral analysis and imaging system optimization," *Med Phys,* vol. 31, pp. 3057-3067, 2004.

[9] J. H. Hubbell and S. M. Seltzer. (1997). *Tables of X-ray Mass Attenuation Coefficients and Mass Energy-Absorption Coefficients*. Available: http://physics.nist.gov/xaamdi

[10] M. Abella, E. Serrano, J. Garcia-Blas, I. García, C. de Molina, J. Carretero*, et al.*, "FUX-Sim: An implementation of a fast universal simulation/reconstruction framework for X-ray systems," *Plos One,* vol. 12, p. e0180363, 2017.

[11] A. A. o. P. i. Medicine, "AAPM report 74: Quality control in diagnostic radiology," 2002.

[12] A. Badal and A. Badano, "Accelerating Monte Carlo simulations of photon transport in a voxelized geometry using a massively parallel graphics processing unit," *Med Phys,* vol. 36, pp. 4878-4880, 2009.