COSC 4372: Project and Team Selection

Emily Nguyen	2264927	eanguye6@cougarnet.uh.edu	Undergraduate
Ishita Uddarraju	2169963	iuddarra@cougarnet.uh.edu	Undergraduate
Muhammad Wajeeh	2063915	mwajeeh@cougarnet.uh.edu	Undergraduate

Introduction

This project will simulate conventional X-ray images through a Python-built device. The software will represent X-ray interaction between various tissues. There will also be a GUI created that allows users to enter different parameters, such as beam energy, X-ray angle, and distances between the source, phantom, and film. These variables will be visualized in a resulting "film image" simulated by the Python-built device.

Our group selected this project because of our collaborative interest in X-rays and their foundational use in medicine. X-rays are a commonly used method to create image formation and human body modeling; thus, we found the most interest in understanding the X-ray device. We also decided to implement the project in Python rather than MATLAB to continue building our skillset in a language more closely aligned to software engineering. We will integrate tools such as NumPy, Matplotlib, and PyQt5/Java Script, similarly to the phantom building in Assignment 1.

Methods

We aim to simulate the function of X-rays and how they produce images. In order to do so, we will model how X-rays are absorbed as they pass through different types of tissue. Our simulation will be based on the Beer-Lambert law; the law describes how light/radiation intensity decreases as it passes through matter. Our program will take in a digital phantom as input, where each pixel on the phantom represents a specific tissue type. This will allow our simulation to showcase realistic differences in the contrast between areas like fat, muscle, and bone.

Python and several different scientific and visualization libraries will be used to implement the software. We will use NumPy for matrix and arithmetic operations, Image rendering and histogram visualization will be done via Matplotlib, and finally, we will use PyQt5 or JavaScript for our GUI. The GUI will allow users to adjust the different acquisition parameters themselves and see them in the resulting X-ray produced. For example, the following are some acquisition parameters that can be played with:

- Beam energy (kVp): determines the penetration and contrast of the X-ray beam.
- Source-to-Detector Distance (SDD): affects magnification and intensity at the film plane.
- Source-to-Object Distance (SOD): controls geometric sharpness and beam spread.
- X-ray angle: allows simulation of different projection views.
- Exposure time: influences total photon fluence and dose level.

- Filtration thickness and material: filters low-energy photons to model beam hardening.
- Grid usage: simulates anti-scatter grid effects on contrast and dose.

Each of these parameters will change the simulated image; this will allow users to observe how these changes in the parameters affect the final radiograph

We intend to run a series of experiments in order to vary the parameters in a systematic way. This way, our results can be recorded and reproduced. We will evaluate how the varying parameter values impact image brightness, contrast, and estimated patient dose. Furthermore, quantitative metrics such as average pixel intensity, contrast ratio, and signal-to-noise ratio will also be calculated for each configuration.

Our project will seek to answer the following questions:

- How does changing different acquisition parameters affect the resulting X-ray produced? Does playing with the parameters affect the 2d phantom's results differently than the 3d phantom's results?
- After compression, what is the effect on the image/contrast? Quantify the change. (For Breast)
- What is the effect of X-ray angle on the profile? (e.g., asymmetry, apparent thickness/ path length changes)

The following results will be produced by our simulation and will be used to present how different imaging parameters result in different radiographic profile outcomes:

- A schematic of the phantom showing the two embedded structures along with their corresponding attenuation coefficients.
- Intensity profile plots (signal vs. position) for baseline settings, with additional overlay curves showing how the profile changes when distance, attenuation coefficient, or angle are varied. Include short explanatory notes about how edge locations or magnification shift with different parameters.
- A summary of regions of interest (mean and standard deviation) and contrast between structures for each imaging setup. These demonstrate how profile generation and parameter adjustments affect image characteristics.
- A 2D film image of a simulated breast phantom with a central lesion.
- Comparison of lesion visibility and contrast between the lesion and surrounding tissue, with a qualitative note on detectability.
- A compressed version of the image representing reduced breast thickness.
- A diagram of the test phantom, including material labels and attenuation values.
- A set of 1D intensity profiles showing baseline conditions along with overlays for different distances, attenuation coefficients, or angles.

- A baseline 2D projection of the 3D phantom model.
- Comparative analysis between the baseline and compressed breast cases, highlighting how compression influences visibility and contrast.
- ROI-based results, including mean, standard deviation, and contrast values for test phantom configurations.
- Quantitative comparisons for different simulation tracks (e.g., lesion vs. background) and for parameter variations.
- A summary quantifying the effects of compression and acquisition parameter changes on image quality and contrast.

Challenges

1. GUI Parameters - User input is always likely to cause challenges for the developers because it is an always-changing variable with many edge cases to cover. To allow for live parameter adjustments, there is also a concern of performance lag in the GUI.
2. Accuracy - This project is our first time building a medical device; therefore, there are concerns of numerical accuracy across every material and angle. We plan to use Beer-Lambert's law, which may be difficult to correctly apply for angled X-rays.

Literature Review

Berger M, Yang Q, Maier A. X-ray Imaging. 2018 Aug 3. In: Maier A, Steidl S, Christlein V, et al., editors. Medical Imaging Systems: An Introductory Guide [Internet]. Cham (CH): Springer; 2018. Chapter 7. Available from: https://www.ncbi.nlm.nih.gov/books/NBK546155/ doi: 10.1007/978-3-319-96520-8_7

Lee J. Python-based Image Processing for In-situ X-ray Imaging [Internet]. Ithaca (NY): Cornell Laboratory for Accelerator-based Sciences and Education (CLASSE); 2020 Aug 3. Available from: https://www.classe.cornell.edu/rsrc/Home/StudentOpportunities/SunrisePrem/John_Lee_Technical_Memo.pdf

Radiology Education Division. X-ray Film Reading Made Easy [Internet]. Grand Forks (ND): University of North Dakota School of Medicine & Health Sciences; [cited 2025 Oct 7]. Available from: https://med.und.edu/education-training/radiology/_files/docs/xray-film-reading-made-easy.pdf