VnV Plan Presentation

FBP CT Image Reconstruction

Qianlin Chen

Department of Computing and Software
McMaster University

February 14, 2025

Overview

SRS Verfication Plan

Initial Review

Assigned reviewers, instructor (Dr. Smith) and peer reviewer (Joe). will conduct a **manual review** of the SRS document. In this step, a structured SRS Checklist will be used to systematically evaluate key aspects such as requirement completeness, consistency, feasibility, and traceability.

Issue Tracking

Reviewers will provide feedback by documenting identified issues in an issue tracker (**GitHub Issues**). Each issue will be assigned to the document owner for later resolution.

SRS Verfication Plan Continue

Revision and Resolution
: The author will address reported issues by making necessary modifications to the SRS. Any unresolved concerns will be discussed among the reviewers.

Final Documentation
: After revisions, author will conduct a final review to confirm that all identified issues have been adequately addressed.

Automated Testing and Verification Tools

Unit Testing

pytest will be used to test **individual functions**, such as backprojection and filtering operations, to verify that they produce the expected output given test input data.

Performance Analysis

cProfill will analyze **execution time** and then identify performance bottlenecks in computationally intensive functions like fourier transforms and interpolation.

examples

Static Analysis

flake8 will enforce Python **coding standards**, ensuring maintainability and readability.

Automated Testing and Verification Tools Continue

Test Coverage

coverage.py will measure test coverage to ensure that all critical functions are tested and validated against various datasets.

Visualization

matplotlib will be used to visualize reconstructed images, allowing for **manual verification** of reconstruction accuracy.

Continuous Integration (CI) tools are **unnecessary** for this project since it does not require automated builds, tests, or deployments like a web application. Additionally, **visual verification is crucial** for image reconstruction, and CI automation cannot replace the need for manual inspection of reconstructed images.

Tests for Functional Requirements Overview

The tests for functional requirements in SRS can be divided into **input** and **output** tests.

R1 correspond to **input-related** tests, ensuring that the system correctly processes input images, and projects data.

R2 correspond to a **input-related** tests based on the user's selection of the filter type.

R3 correspond to **output** tests, verifying the correctness of reconstructed attenuation values after back projection.

Input Tests for Functional Requirements

R1: The system must accept a grayscale 2D input image, apply the Radon Transform to generate projection data, and ensure that intensity values are normalized between 0 and 1.

test-input-intensity-id1

Control: Automatic

Initial State: Pending input.

Input: A 2-D NumPy array (M by N matrix), where:

- Number of row (M): Number of detector positions
- Number of column (N): Number of projection angles.
- Value in the matrix: X-ray intensity measurements, normalized between 0 and 1.

Output: A valid 2-D matrix array (M by N matrix), where:

- number of row (m): number of detector positions
- number of column (n): number of projection angles.
- Value in the matrix: Represent integrated projection data, between 0 and 1.

Input Test for Functional Requirements Continue

Test Case Derivation:

This test ensures that the system correctly accepts a valid grayscale 2D input image, applies the Radon Transform, and produces a valid sinogram matrix.

The system should return an output that **maintains consistency in shape and correctly reflects the values after applying the log transform.**

How test will be performed: The test will be automated using PyTest.

Input Test for Functional Requirements Continue

R2: The system must allow user to choose filtering methods, either a Ramp or Shepp-Logan filter.

test-input-filter-type-id2

Control: Automatic

Initial State: Pending input.

Input: A filter type (selected from the options: "ramp", "shepp-logan").

Output: A 1-D array that epresent the frequency response of the chosen filter in the Fourier domain.

Test Case Derivation:

This test ensures that the system correctly processes user-selected filter types.

A valid filter should be applied as expected, where:

- Valid Ramp-filter: The values in the 1D array follow a linear function.
- Valid Shepp-logan filter: The values in the 1D array follow a sinc function.

Output Tests for Functional Requirements

R3: The system must perform the back-projection over all projection angles to reconstruct the attenuation coefficient after filtering.

test-output-filter-id3

Control: Automatic

Initial State: The system is ready to perform back-projection on a given sinogram and filters.

Input:
- Filter type selected from the available options: "ramp", "shepp-logan".
- Output size: Desired size of the reconstructed image.
- Sinogram (M by N matrix).

Output: A valid 2-D reconstructed image matrix (P by P), where:
- Number of rows and columns (P by P): The final reconstructed image size.
- Values in the matrix: Represent attenuation coefficients reconstructed from projections.

Output Test for Functional Requirements Continue

Test Case Derivation:

This test ensures that the system correctly applies the back-projection process to reconstruct an image from a given sinogram.
The output image **maintains the expected input size, with reconstructed attenuation values well preserving the original structure while filtering out frequency noise.**

How test will be performed: The test will be automated using PyTest.

Objectives

The purpose of the validation plan is to define how system validation will perform at the end of the project. The strategy will use to assess whether the developed system accomplishes the design goals. Also, the verification plan includes test strategies, definitions of what will be tested, and a test matrix with detailed mapping connecting the tests performed to the system requirements.

Due to **resource constraints**, validation will be limited to peer-based testing for accuracy and usability. While this ensures basic functionality and user experience evaluation, the absence of domain experts, such as doctors or researchers, prevents validation in real-world clinical or research applications.

Questions?