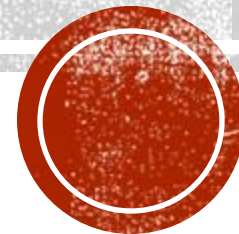


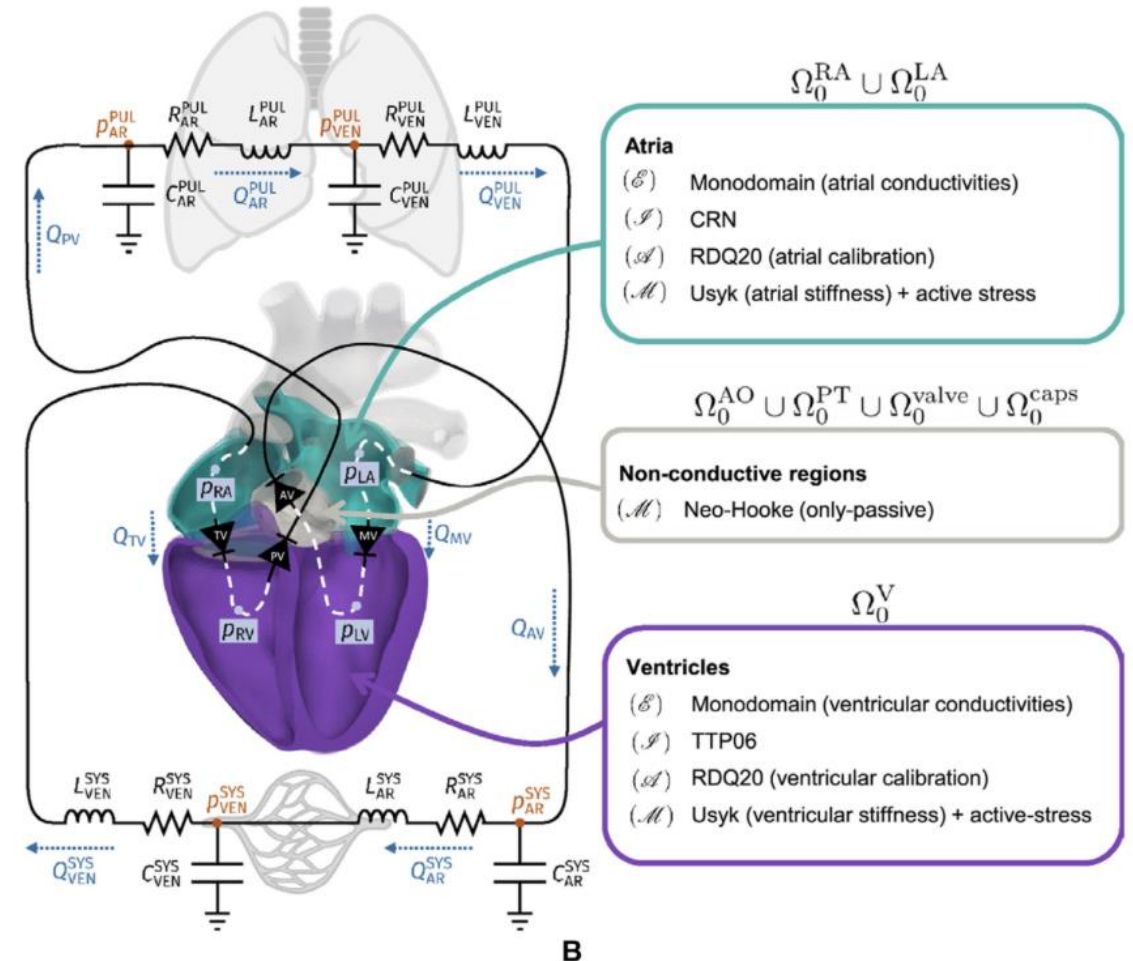
CARDIAC MESH GENERATOR

Vinay Jani



BACKGROUND — DIGITAL TWINS

- Digital Twins are virtual representations of a patient with real-time updates of phenotypic data for precision medicine.
- These involve the generation of 3D models. However, generation of said models has many problems, including:
 - Lack of standardization
 - No co-registration of multiple views
 - Imaging Data (CT, MRI) are 2D.



BACKGROUND — MESH GENERATION AND FINITE ELEMENT ANALYSIS

Image Co-Registration and Universal Coordinate Transformation

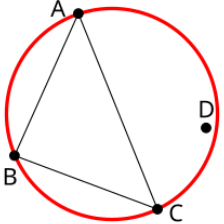
$$\begin{bmatrix} P_x \\ P_y \\ P_z \\ 1 \end{bmatrix} = \begin{bmatrix} X_x & \Delta_i & Y_x & \Delta_j & 0 & S_x \\ X_y & \Delta_i & Y_y & \Delta_j & 0 & S_y \\ X_z & \Delta_i & Y_z & \Delta_j & 0 & S_z \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} i \\ j \\ 0 \\ 1 \end{bmatrix} = M \begin{bmatrix} i \\ j \\ 0 \\ 1 \end{bmatrix}$$

absolute
coordinates

image indices

pixel spacing;
direction cosines

Mesh Generation and Delaunay Triangulation

$$\begin{vmatrix} A_x & A_y & A_x^2 + A_y^2 & 1 \\ B_x & B_y & B_x^2 + B_y^2 & 1 \\ C_x & C_y & C_x^2 + C_y^2 & 1 \\ D_x & D_y & D_x^2 + D_y^2 & 1 \end{vmatrix}$$


$$= \begin{vmatrix} A_x - D_x & A_y - D_y & (A_x - D_x)^2 + (A_y - D_y)^2 \\ B_x - D_x & B_y - D_y & (B_x - D_x)^2 + (B_y - D_y)^2 \\ C_x - D_x & C_y - D_y & (C_x - D_x)^2 + (C_y - D_y)^2 \end{vmatrix} > 0$$

Parameter Optimization: Elliptic PDEs

$$\alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta} = -I^2(Px_{\xi} + Qx_{\eta})$$

$$\alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta} = -I^2(Py_{\xi} + Qy_{\eta})$$

$$\alpha = x_{\eta}^2 + y_{\eta}^2$$

$$\beta = x_{\eta}x_{\xi} + y_{\eta}y_{\xi}$$

$$\gamma = x_{\xi}^2 + y_{\xi}^2$$

$$I = \frac{\delta(x, y)}{\delta(\xi, \eta)} = y_{\eta}x_{\xi} - y_{\xi}x_{\eta}$$

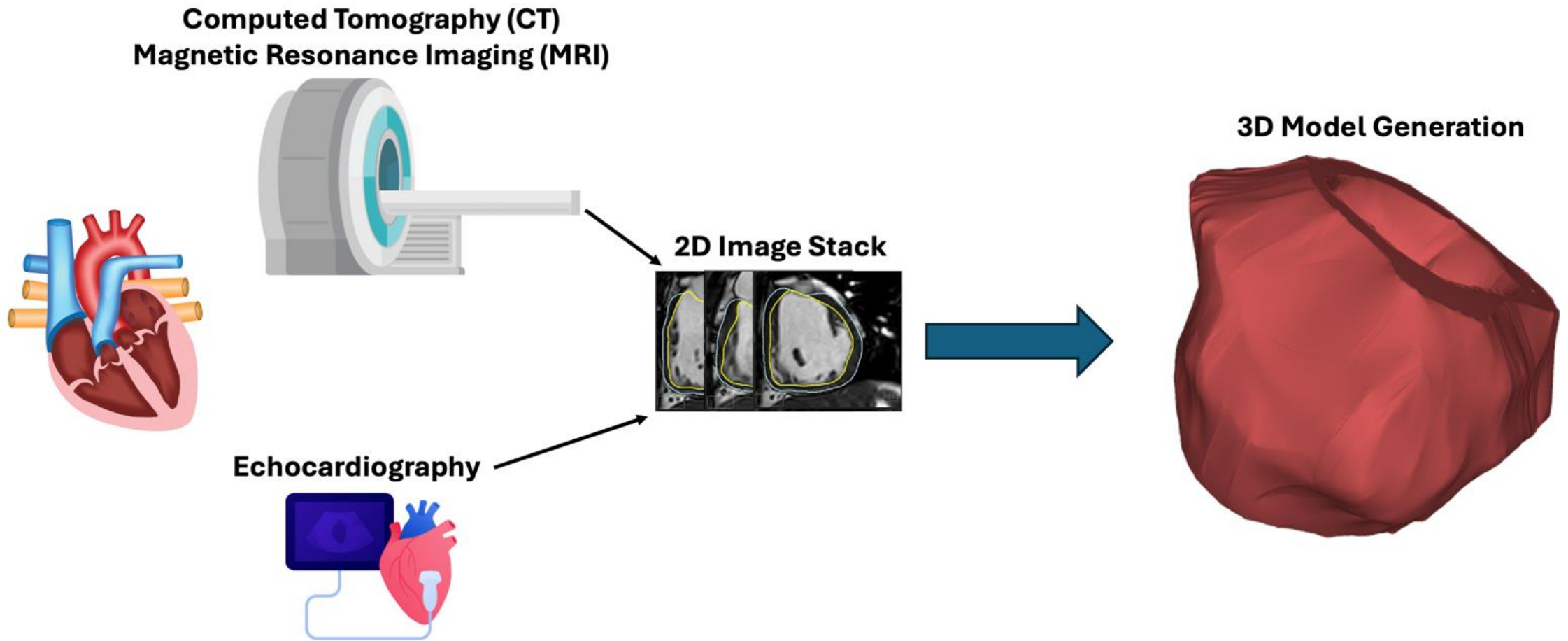


PROBLEM STATEMENT

- A means to visualize and generate 3D models from cardiac MRI data , which can colocalize views and is easily accessible does not exist

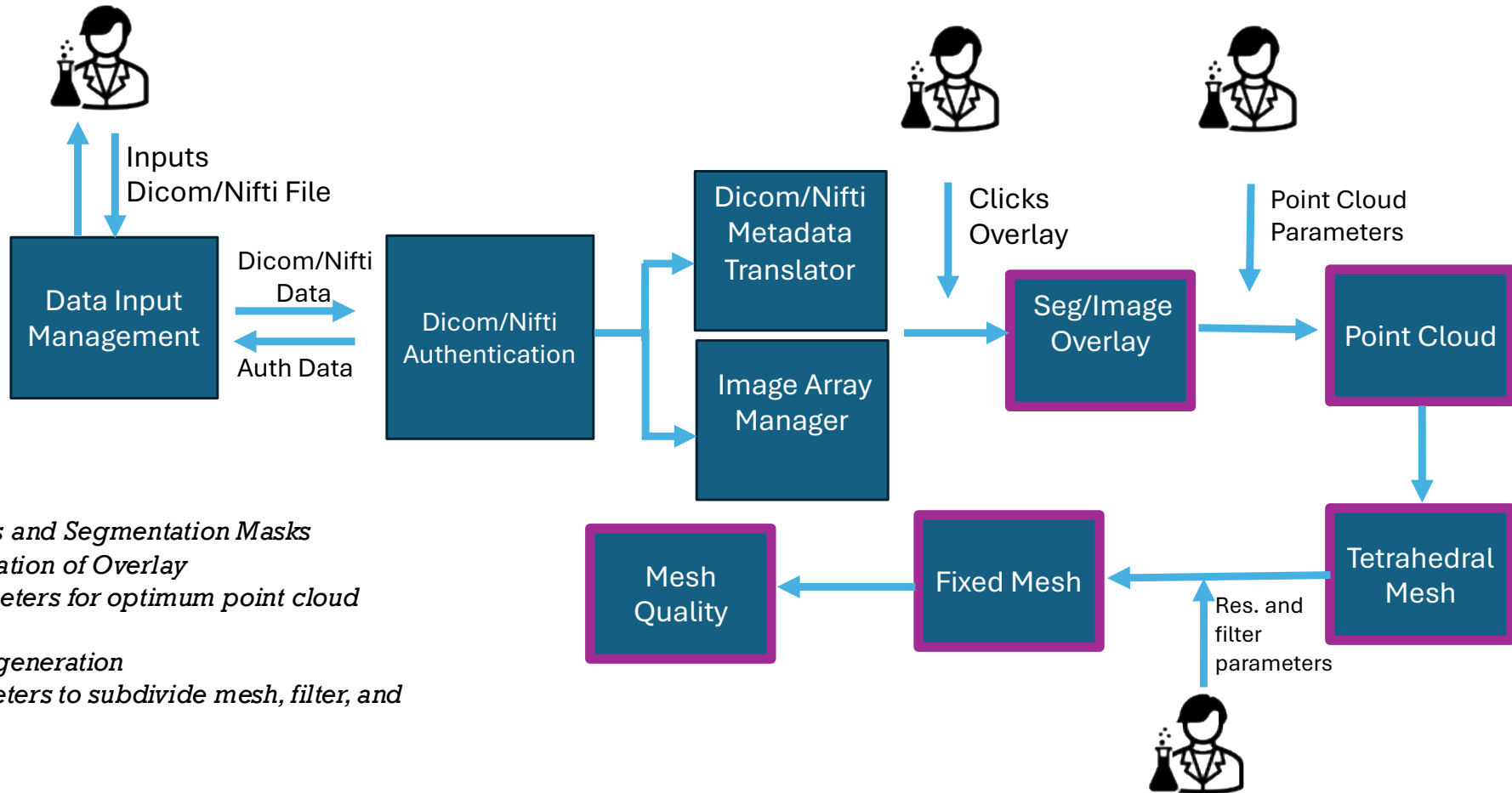


PROPOSED PIPELINE FOR MODEL GENERATION



USE CASES

User Profile: Non-programmer user who uses patient specific cardiac studies and simulations



1. User Inputs Dicoms and Segmentation Masks
2. User clicks visualization of Overlay
3. User selects parameters for optimum point cloud generation
4. User selects mesh generation
5. User inputs parameters to subdivide mesh, filter, and fix mesh



METHODOLOGY

Overlay and Image Processing



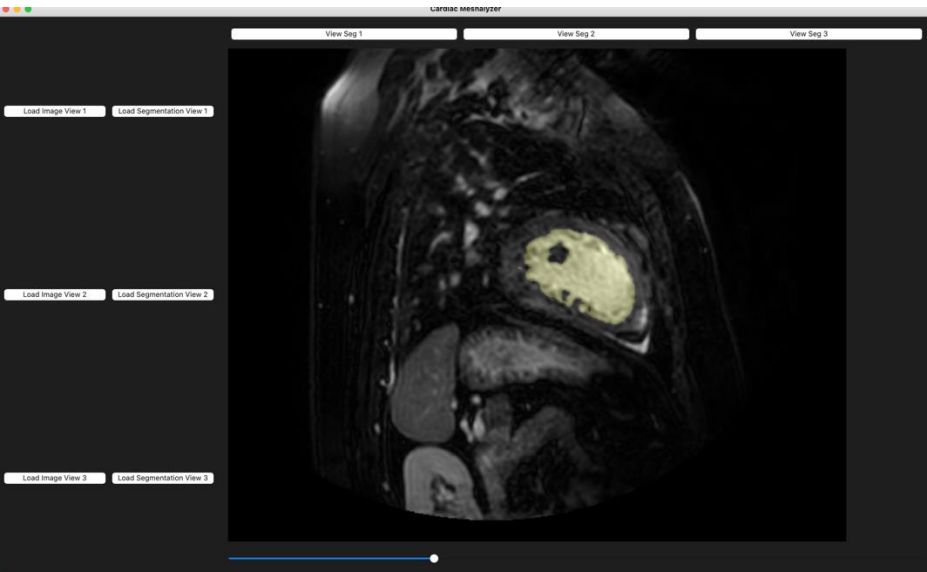
Point Cloud Generation



Tetrahedral Mesh



OUTPUTS DEMO

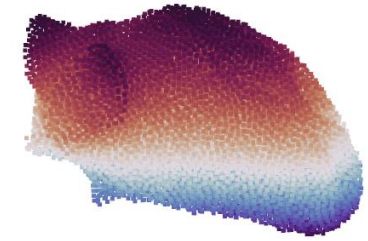


Point Cloud

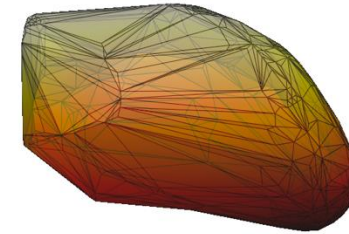
Initial Tetra Mesh

Cleaned Tetra Mesh

Point Cloud Visualization



Tetrahedral Mesh Visualization



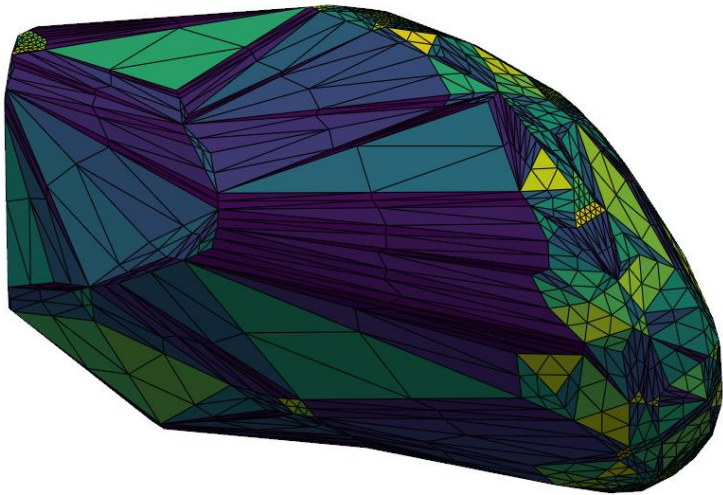
Cleaned Tetrahedral Mesh Visualization



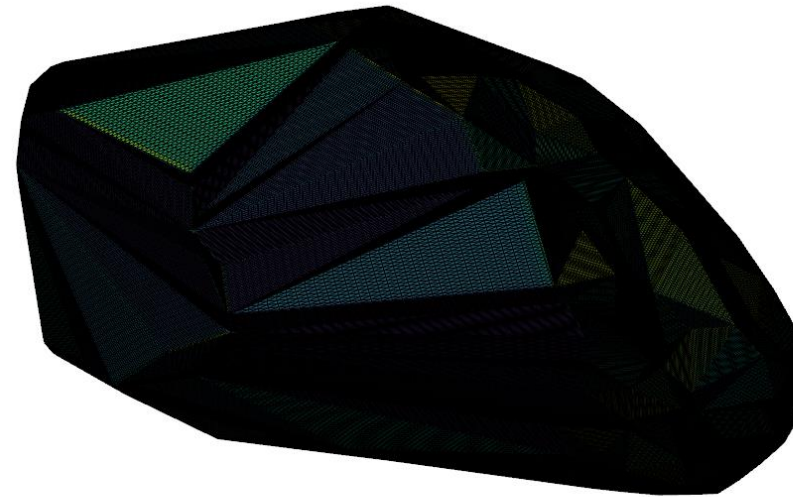
RESULTS

- **Speed** – Can make mesh in less than 1 second and perform a 100 Poisson iterations with 5 subdivisions in less than 10 seconds
- **Filtering and cleaning methods allow for increased resolution**

No Smoothing or Filtering



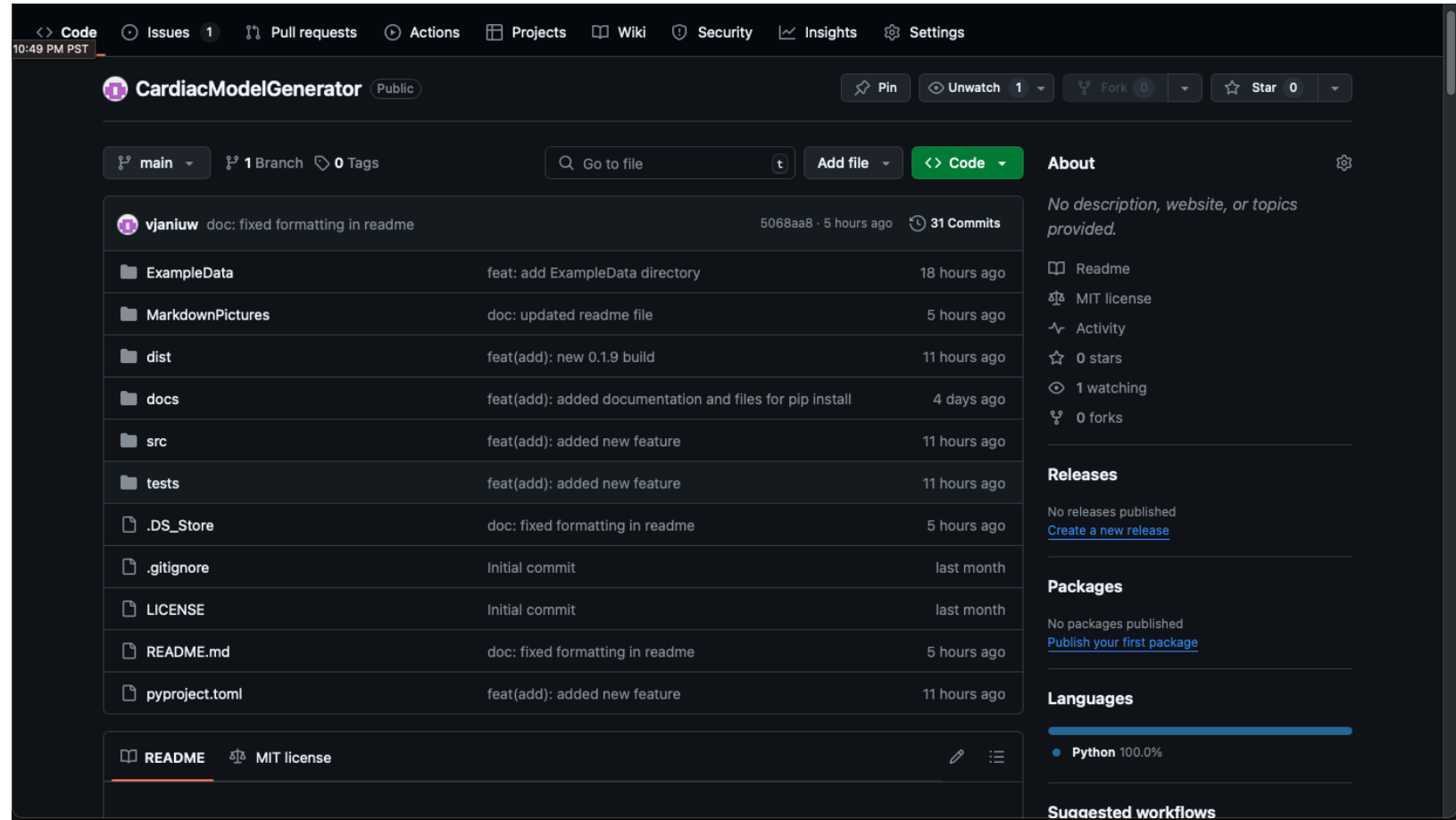
New Smoothing/Subdivision Algorithm



PROJECT STRUCTURE

CardiacModelGenerator/

- ExampleData/
- MarkdownPictures/
- dist/
- docs/
- src/
- tests/
- .gitignore
- LICENSE
- README.md
- pyproject.toml



The screenshot shows the GitHub repository page for **CardiacModelGenerator** by user **vjaniuw**. The repository is public and has 1 branch (main) and 0 tags. The commit history table lists the following files and their commit details:

File	Commit Message	Commit Hash	Time Ago
ExampleData	feat: add ExampleData directory	5068aa8	18 hours ago
MarkdownPictures	doc: updated readme file		5 hours ago
dist	feat(add): new 0.1.9 build		11 hours ago
docs	feat(add): added documentation and files for pip install		4 days ago
src	feat(add): added new feature		11 hours ago
tests	feat(add): added new feature		11 hours ago
.DS_Store	doc: fixed formatting in readme		5 hours ago
.gitignore	Initial commit		last month
LICENSE	Initial commit		last month
README.md	doc: fixed formatting in readme		5 hours ago
pyproject.toml	feat(add): added new feature		11 hours ago

The right sidebar shows repository statistics: 0 stars, 1 watching, and 0 forks. It also includes sections for Releases, Packages, Languages (Python 100.0%), and Suggested workflows.

<https://github.com/vjaniuw/CardiacModelGenerator>



CHALLENGES

1. Object Oriented Programming with WX
2. Meshing Computationally Expensive
3. User Configurable Meshing is non-uniform
4. Co-registration of images in real coordinates
5. Image Segmentation input requires expertise
6. Rigidity of Medical Image Data



APPENDIX



DESIGN INPUTS- LAYERED ARCHITECTURE

User Interface Layer

Data Input
Management

3D Object Visualization
Manager

Point Cloud Manager

Mesh Manager

Buisness/Algorithm Logic Layer

Universal
Coordinate
Converter

Point Cloud
Generator

Mesh
Generator

3D Model Saving
Manager

Mesh Quality
Metric Algorithm

Data Interpretation

Dicom/Nifti
Metadata
Translator

Image Array
Manager

Dicom/Nifti
Authenticator

Mesh Finite
Element
Solver

Mesh Smoothing
and Subdividing
Solver



DESIGN OUTPUT

