

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC CẦN THƠ
KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**



**NIÊN LUẬN CƠ SỞ NGÀNH
NGÀNH KHOA HỌC MÁY TÍNH**

Đề tài
TÁI TẠO 3D CÁC ẢNH CT NGƯỢC SỬ DỤNG
THUẬT TOÁN MARCHING CUBES

Sinh viên thực hiện : Huỳnh Trương Minh Quang

Mã số : B1510210

Khóa : 44

Cần Thơ, 8/2021

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC CẦN THƠ
KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**



**NIÊN LUẬN CƠ SỞ NGÀNH
NGÀNH KHOA HỌC MÁY TÍNH**

Đề tài

**THREE-DIMENSIONAL RECONSTRUCTION OF
CHEST COMPUTED TOMOGRAPHY USING
MARCHING CUBES**

**Giáo viên hướng dẫn:
TS.Trần Nguyễn Minh Thư**

**Sinh viên thực hiện:
Huỳnh Trương Minh Quang**

Mã số: B1510210

Khóa: 44

Cần Thơ, 8/2021

NHẬN XÉT CỦA GIẢNG VIÊN

Cần Thơ, ngày tháng năm
(GVHD ký và ghi rõ họ tên)

LỜI CẢM ƠN

Kính dâng:

Cha, mẹ đã luôn bên con, chăm sóc dạy dỗ cho con khôn lớn, luôn giúp con vượt qua những khó khăn và áp lực.

Xin tỏ lòng biết ơn sâu sắc đến: Ts. Trần Nguyễn Minh Thư đã tận tình hướng dẫn dạy bảo và tạo điều kiện thuận lợi cho em hoàn thành tốt niên luận này.

Chân thành cảm ơn các quý Thầy Cô trường Đại Học Cần Thơ đã tạo điều kiện cho em học tập tốt trong những năm vừa qua.

Cần Thơ, ngày 28 tháng 8 năm 2021

Người viết

Huỳnh Trương Minh Quang

MỤC LỤC

NHẬN XÉT CỦA GIẢNG VIÊN.....	3
LỜI CẢM ƠN.....	4
MỤC LỤC.....	1
DANH MỤC HÌNH.....	3
DANH MỤC HÌNH.....	4
DANH MỤC BẢNG.....	4
DANH MỤC TỪ VIẾT TẮT.....	5
TÓM TẮT.....	6
PHẦN GIỚI THIỆU.....	7
1. Đặt vấn đề.....	7
2. Lịch sử giải quyết vấn đề.....	8
3. Mục tiêu đề tài.....	9
4. Đối tượng và phạm vi nghiên cứu.....	10
5. Phương pháp nghiên cứu.....	10
6. Kết quả đạt được.....	10
PHẦN NỘI DUNG.....	11
CHƯƠNG 1.....	11
MÔ TẢ BÀI TOÁN.....	11
1. Mô tả chi tiết bài toán.....	11
2. Vấn đề và giải pháp liên quan đến bài toán.....	11
2.1 Đọc ảnh DICOM với pydicom.....	11
2.2 Chỉ số Hounsfield.....	12
2.3 Thuật toán marching cubes.....	13

2.4 Phương thức marching_cubes của thư viện skimage.measure.....	14
2.5 Hiển thị bằng thư viện pyqtgraph.opengl.....	14
CHƯƠNG 2.....	15
THIẾT KẾ VÀ CÀI ĐẶT.....	15
1. Thiết kế hệ thống.....	15
2. Cài đặt giải thuật.....	16
2.1 Đọc ảnh DICOM.....	16
2.2 Chuyển đổi sang thang mức xám.....	17
2.3 Marching cubes.....	17
2.4 Hiển thị.....	17
CHƯƠNG 3.....	19
KIỂM THỬ VÀ ĐÁNH GIÁ.....	19
1. Kết quả kiểm tra.....	19
1.1 Giao diện.....	19
1.2 Kiểm thử.....	24
1.2.1 Kịch bản 1: Có sẵn thư mục DICOM.....	24
1.2.2 Kịch bản 2: Có sẵn tệp tin chứa dữ liệu ảnh DICOM.....	26
1.2.3 Kịch bản 3: Có sẵn tệp tin chứa dữ liệu điểm ảnh đã qua xử lý bằng thuật toán Marching cubes.....	29
1.3 Phân tích các bộ ảnh DICOM đã qua quan sát và kiểm thử.....	31
PHẦN KẾT LUẬN.....	32
1. Kết quả đạt được.....	32
2. Hướng phát triển.....	32
TÀI LIỆU THAM KHẢO.....	33

DANH MỤC HÌNH

Hình	Tên hình	Trang
1.1	Mô tả quá trình chuyển từ bộ ảnh 2D sang 3D của ứng dụng Dicom2Mesh (6)	8
1.2	Giao diện tương tác và hiển thị không gian 3 chiều ứng dụng 3Dicom Viewer (7)	9
1.3	Các trường hợp cơ bản và phương pháp xác định các trường hợp của khối lập phương (3)	13
1.4	Hình ảnh khi phóng to của khối sau khi được tái tạo sử dụng marching_cubes	14
2.1	Sơ đồ các bước thiết kế hệ thống	15
2.2	Hàm đọc ảnh DICOM	17
2.3	Hàm chuyển đổi sang ảnh xám	17
2.4	Hàm gọi thuật toán marching cubes	17
2.5	Hàm hiển thị mô hình ba chiều	18
3.1	Giao diện chính	19
3.2	Nhập đường dẫn đến tệp tin DICOM	19
3.3	Các chức năng cho phép đọc tệp tin ảnh có sẵn, đọc dữ liệu điểm trong không gian 3 chiều đã qua xử lý trước và lưu dữ liệu. Và nơi chọn phiên bản để đọc	19
3.4	Đường dẫn đến các tệp tin được lưu khi dùng chức năng save to clipboard	20
3.5	Nơi nhập ngưỡng để thực hiện marching cubes, hình ảnh lược đồ xám và ảnh đầu tiên của bộ ảnh DICOM	20
3.6	Chuyển đổi màu sắc cho mô hình ba chiều khi hiển thị	21
3.7	Các nút chức năng lấy dữ liệu tệp tin DICOM, tạo mô hình, hiển thị mô hình	21
3.8	Giao diện của thẻ 2	21
3.9	Ảnh DICOM ở mặt phẳng ngang	22
3.10	Ảnh DICOM ở mặt phẳng dọc	22
3.11	Ảnh DICOM ở mặt phẳng từ trên xuống	23
3.12	Lựa chọn khoảng ảnh sẽ được dựng mô hình, dựa vào ảnh ở mặt phẳng ngang	23
3.13	Giao diện của thẻ 3	24

DANH MỤC HÌNH

Hình	Tên hình	Trang
3.14	Giao diện của thẻ 1 sau kiểm thử 1	25
3.15	Giao diện của thẻ 2 sau kiểm thử 1	25
3.16	Giao diện của thẻ 3 sau kiểm thử 1	26
3.17	Giao diện của thẻ 1 sau kiểm thử 2	27
3.18	Giao diện của thẻ 2 sau kiểm thử 2	27
3.19	Giao diện của thẻ 3 sau kiểm thử 2	28
3.20	Giao diện của thẻ 1 sau kiểm thử 3	29
3.21	Giao diện của thẻ 3 sau kiểm thử 3	30

DANH MỤC BẢNG

Bảng	Tên bảng	Trang
1.1	Bảng quy đổi xác định các chất bằng chỉ số Hounsfield	13
1.2	Bảng so sánh các mẫu DICOM được dùng	31

DANH MỤC TỪ VIẾT TẮT

STT	Ký hiệu chữ viết tắt	Chữ viết đầy đủ	Ghi chú
1	CT	Computed tomography	Chụp cắt lớp vi tính
2	GGO	Ground-glass opacity	Độ mờ kính mặt đất
3	CTA	Computed tomography angiography	Chụp cắt lớp động mạch

TÓM TẮT

Trong y học hiện đại, để chuẩn đoán bệnh các bác sĩ có thể dùng các máy móc hiện đại chụp ảnh bên trong cơ thể tạo ra các tập ảnh giúp quan sát tìm hiểu cơ thể người. Nhưng với bộ ảnh chỉ ở dạng hai chiều sẽ khó khăn hơn là quan sát một cách rõ ràng với hình ảnh ba chiều cùng với khả năng tương tác.

Theo đó những bộ ảnh y tế có thể được tổng hợp lại tạo thành mô hình ba chiều với các kỹ thuật kết xuất bề mặt như Marching cubes nhằm chiết xuất thông tin từng pixel của tập ảnh hai chiều sang thông tin voxel ba chiều.

Sau khi áp dụng thuật toán Marching cubes, với hầu hết các tập dữ liệu ảnh định dạng DICOM đã có thể chuyển sang dạng hình khối trong không gian ba chiều.

PHẦN GIỚI THIỆU

1. Đặt vấn đề

Trong lĩnh vực y học, ảnh chụp cắt lớp hay ảnh CT được sử dụng chuẩn đoán nhiều loại ung thư, như ung thư gan, phổi, tuyến tụy. Giúp cho các bác sĩ xác định vị trí của các khối u, kích thước và mức độ ảnh hưởng đến các mô lân cận (5).

Ảnh CT tại khu vực bụng, ngực có thể phát hiện khối u hoặc bất kỳ vết sưng viêm nào ở các nội tạng. Nhận thấy được các vết rách của các lá lách, thận và gan. Cũng như giúp các bác sĩ đánh giá các bệnh về xương, mật độ xương và tình trạng cột sống của bệnh nhân (5).

Các hình ảnh y khoa là một khía cạnh phổ biến để nghiên cứu trong y học, đặc biệt là ảnh DICOM vì nó có thể được tạo bởi nhiều loại thiết bị khác nhau. Sự tái cấu trúc các ảnh DICOM thành hình ảnh 3 chiều là cần thiết để phản ánh một cách chính xác kết quả của một lần kiểm tra (1). Tái tạo và hiển thị hình ảnh y khoa ba chiều có ý nghĩa lớn trong chẩn đoán, lập kế hoạch và mô phỏng y tế, phẫu thuật, phẫu thuật chỉnh hình, lập kế hoạch bức xạ và giải phẫu (Lai Khin Wee, 2011).

Tái tạo 3D nói chung, bao gồm cả xây dựng model 3D từ một tập hợp các lát cắt. Các lát cắt này đều là ảnh 2 chiều và thường là ảnh DICOM. Từ một số bước và thủ tục nhất định mô hình 3D của cơ thể sẽ được tái tạo (4).

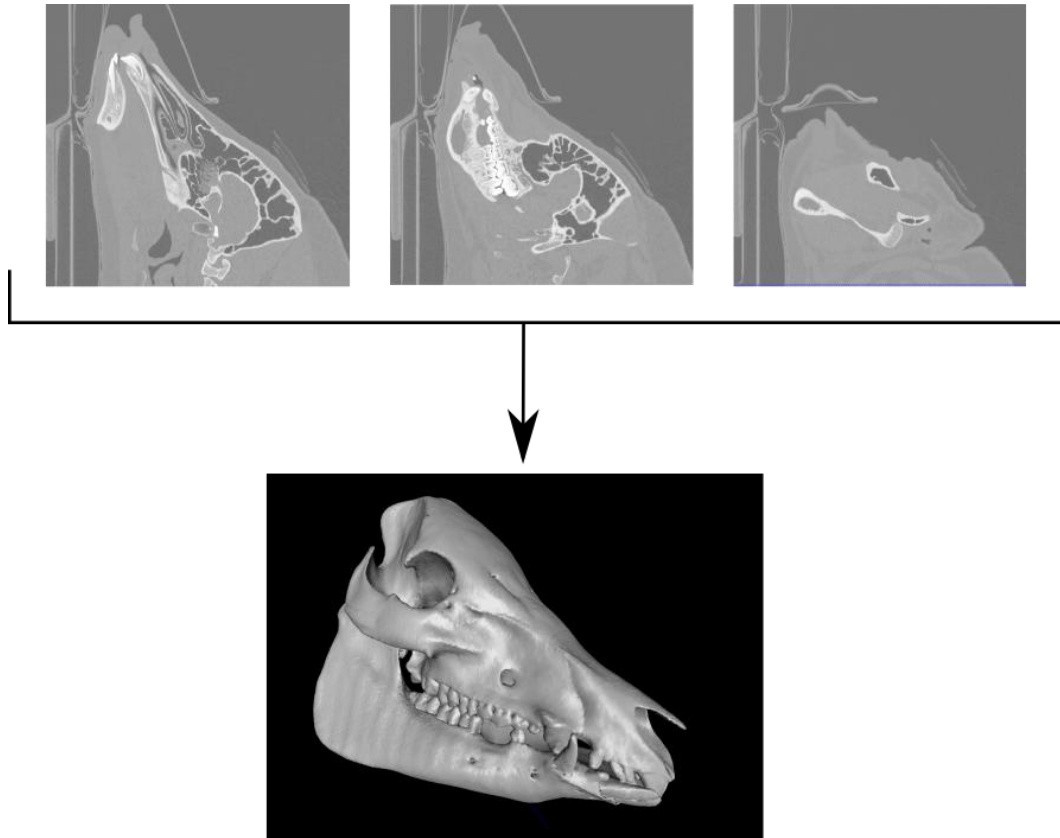
Ứng dụng của tái tạo 3D rất rộng, từ các phân tích bình thường của bác sĩ đến các quyết định quan trọng. Hệ thống tái tạo 3D là thiết yếu để hỗ trợ họ ra các quyết định quan trọng đó. Hệ thống sẽ giúp các bác sĩ quan sát theo góc nhìn 3 chiều các lát cắt ảnh và cung cấp cho họ khả năng tương tác xoay, thu phóng (4).

Việc lắp ráp mô hình 3D từ tập hợp các ảnh CT rất có lợi cho các nhân viên y tế. Mô hình 3D rất hữu ích cho các bác sĩ để hiểu giải phẫu bên trong của bệnh nhân hơn là các lát cắt 2D. Trong tất cả các cách để thực hiện thì thuật toán Marching cubes là phổ biến nhất (2).

Trong đề tài này chúng tôi sẽ tái tạo 3D các ảnh CT ngực sử dụng thuật toán Marching Cubes.

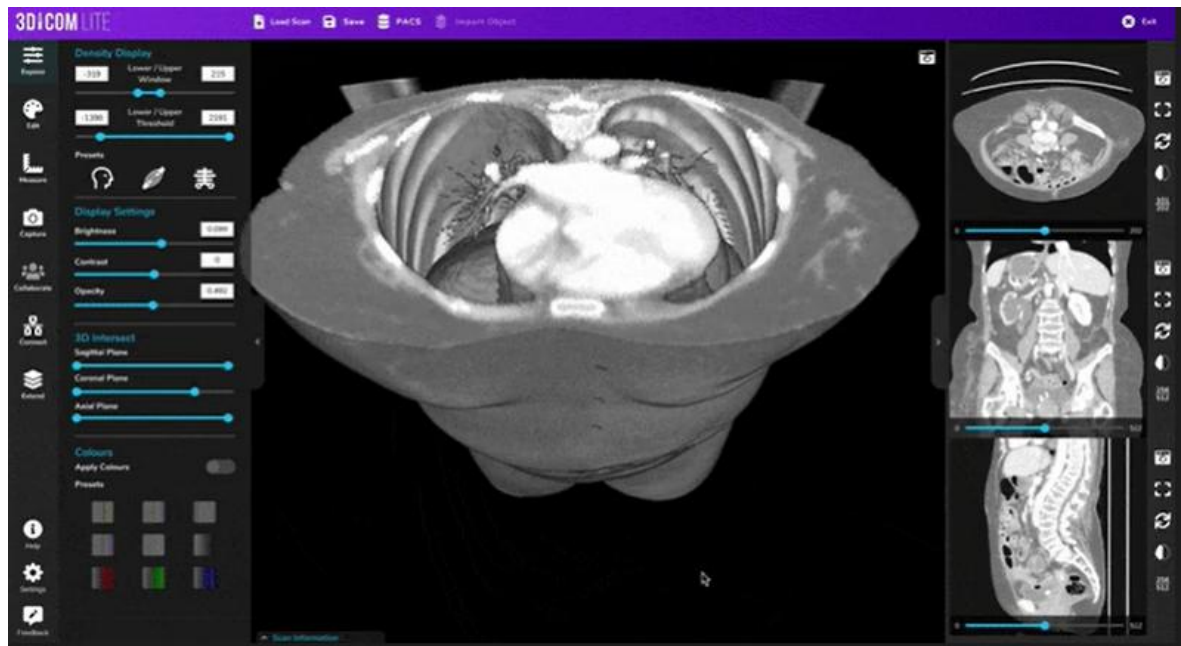
2. Lịch sử giải quyết vấn đề

Ứng dụng Dicom2Mesh được giới thiệu bởi AOT AG, đây là ứng dụng tái tạo dạng ảnh đầu vào là DICOM sang các định dạng 3D như STL, OBJ và PLY sử dụng thuật toán Marching cubes (6).



Hình 1.1. Mô tả quá trình chuyển từ bộ ảnh 2D sang 3D của ứng dụng Dicom2Mesh (6)

Ứng dụng 3Dicom Viewer, một ứng dụng được phát triển nhằm đo lường và trực quan hóa trong không gian ba chiều. Với sự tối ưu của phương pháp tái tạo bề mặt 3D, các ảnh CT, MRI và PET dưới định dạng DICOM đều được tái tạo và hiển thị trên không gian ba chiều có thể tương tác (7).



Hình 1.2. Giao diện tương tác và hiển thị không gian 3 chiều ứng dụng 3Dicom Viewer (7)

Theo nghiên cứu của T.Senthil Kumara và Anupa Vijai (2011), Kết xuất bề mặt là một cách để trực quan hóa đối tượng bằng dữ liệu hình ảnh dưới dạng một tập hợp cứng của các phần tử cơ bản nhất định, chẳng hạn như voxels, faces, đa giác khác, đoạn thẳng và điểm. Các tập hợp này đại diện cho ranh giới của cấu trúc hoặc toàn bộ cấu trúc. Mặt đẳng lập là một dạng tương tự ba chiều của một đường đẳng lượng. Nó là một bề mặt đại diện cho các điểm có giá trị không đổi (4).

Hai phương pháp phổ biến để xây dựng bề mặt đẳng lập từ khối lượng dữ liệu như ảnh chụp CT/MRI của khuôn mặt người là cấu trúc bề mặt dựa trên đường viền và thuật toán trích xuất bề mặt đẳng lập như marching cubes hoặc marching tetrahedra (4).

Hình ảnh trong kiểu dữ liệu DICOM có siêu dữ liệu ở từng lớp cắt và bao gồm các thông tin như độ dày lớp cắt, khoảng cách giữa các lớp cắt, độ phân giải ảnh. Sự thành công của quá trình tái cấu trúc phụ thuộc vào độ chính xác của các ảnh DICOM và các siêu dữ liệu của chúng (1).

3. Mục tiêu đề tài

Tái tạo và hiển thị hình ảnh y khoa ba chiều có ý nghĩa lớn trong chẩn đoán, lập kế hoạch và mô phỏng y tế, phẫu thuật, phẫu thuật chỉnh hình, lập kế hoạch bức xạ

và giải phẫu. Trong đề tài này, thuật toán tái tạo về mặt Marching cubes và quy trình tái tạo ảnh 3D được nghiên cứu cả thực nghiệm trên ảnh CT ngực. Ứng dụng đọc ảnh CT và hiển thị mô hình 3D sau khi tái tạo được xây dựng.

4. Đối tượng và phạm vi nghiên cứu

Ta có nghiên cứu đối tượng là các dạng ảnh y tế với định dạng DICOM, cụ thể là các ảnh chụp cắt lớp ở khu vực ngực.

Phạm vi nghiên cứu trong khoảng thời gian từ 09/08/2021 đến 03/11/2021.

5. Phương pháp nghiên cứu

Xây dựng ứng dụng thực tế dựa vào các nghiên cứu gần giống hoặc giống.

6. Kết quả đạt được

Phần mềm đọc ảnh y tế với định dạng DICOM được xây dựng với ngôn ngữ python. Ảnh định dạng DICOM đã có thể được đọc và tái cấu trúc thành dạng khung. Với dạng khung này dữ liệu có thể tạo cái tệp định dạng STL và hiển thị trực tiếp trên không gian ba chiều có các tương tác cơ bản như xoay và thu phóng.

PHẦN NỘI DUNG

CHƯƠNG 1

MÔ TẢ BÀI TOÁN

1. Mô tả chi tiết bài toán

Với các tệp ảnh thu được sau khi thực hiện chụp CT, chủ yếu là định dạng DICOM. Từ tập dữ liệu các ảnh DICOM này ta cần đọc được dữ liệu của mỗi ảnh như là một ảnh JPG hoặc PNG thông thường, hay có thể nói ta cần đọc được dữ liệu màu của từng điểm ảnh. DICOM sẽ cần một thư viện đặc biệt giúp đọc tệp DICOM và chuyển đổi sang dạng ảnh thông thường để có thể trích xuất thông tin điểm ảnh được.

Thu thập dữ liệu của từng điểm ảnh, đánh dấu vị trí của từng dữ liệu sao cho các ảnh DICOM có tên được xếp ngay liền sau của nhau sẽ nằm tương ứng vị trí liền kề trong không gian ba chiều. Các dữ liệu sẽ cần phải được lưu bằng một cấu trúc dữ liệu thể hiện được sự liền kề và vị trí của nó trong không gian ba chiều, ví dụ như một mảng ba chiều với 3 tham số x , y , z sẽ xác định được vị trí tương đối của điểm ảnh so với các điểm khác trong không gian ba chiều

Xác định thang tổ chức đồ của các ảnh nhằm lựa chọn tái tạo riêng ở mỗi bộ phận khác nhau của cơ thể bệnh nhân trong ảnh. Sau đó áp dụng thuật toán Marching cubes chuyển đổi dữ liệu ảnh trên thành lưới các điểm và đường thẳng trong không gian ba chiều. Thuật toán Marching cubes sẽ cần được xây dựng từ đầu hoặc cần một thư viện hỗ trợ thuật toán Marching cubes giúp đọc được dữ liệu ảnh và chuyển đổi nó sang dạng lưới trong không gian ba chiều.

Sau đó cho hiển thị trong không gian ba chiều. Thư viện đồ họa OpenGL sẽ được áp dụng để đọc được dạng lưới các điểm và đường thẳng.

2. Vấn đề và giải pháp liên quan đến bài toán

2.1 Đọc ảnh DICOM với pydicom

Thư viện pydicom là thư viện hỗ trợ xử lý các ảnh có định dạng là DICOM cho ngôn ngữ lập trình python, pydicom có thể hỗ trợ đọc file, chỉnh sửa, chuyển đổi

sang dữ liệu kiểu ndarray nhằm đọc thông tin pixel, cũng như chỉnh sửa và hiển thị thông tin bệnh nhân.

2.2 Chỉ số Hounsfield

Các giá trị điểm ảnh thu được chỉ là dữ liệu thô, ta chuyển đổi sang dữ liệu chỉ số Hounsfield (thang đo mật độ bức xạ trong mô Hounsfield), những thông số cần thiết để chuyển đổi là tỷ lệ thay đổi của độ dốc (rescale slope) và tỷ lệ thay đổi của giao điểm (rescale intercept)

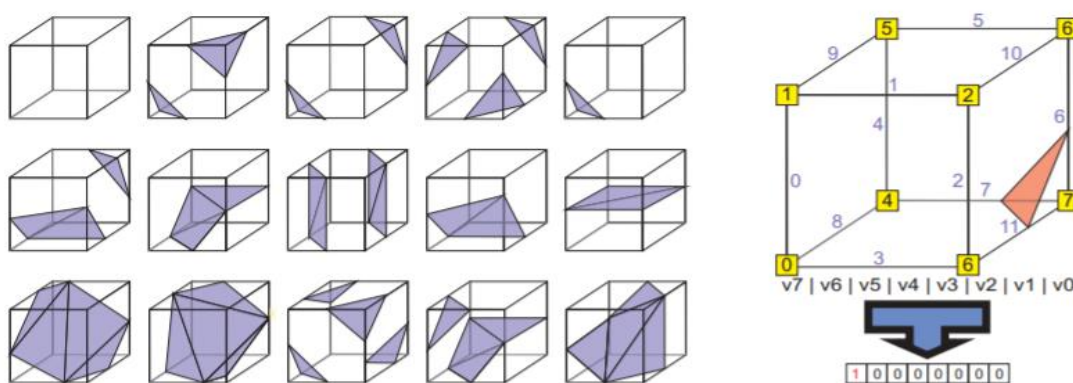
Substance	HU
Không khí	-1000
Phổi	-500
Chất béo	-100 to -50
Nước	0
Máu	+30 to +70
Cơ	+10 to +40
Gan	+40 to +60
Xương	+700 (xương xốp) - +3000 (xương đặc)

Bảng 1.1 Bảng quy đổi xác định các chất bằng chỉ số Hounsfield

2.3 Thuật toán marching cubes

Khoảng trống nhỏ nhất của các tọa độ tương ứng x, y, z . Ta sẽ sinh ra được một hình lập phương.

Với mỗi hình lập phương chạy qua tất các điểm, thì ở tại mỗi điểm sẽ có thể có hoặc không điểm ảnh phù hợp (lọc để chọn ra cơ quan phù hợp với chỉ số Hounsfield hoặc thang mức xám tương ứng với mô mà ta muốn chiết suất) ta sẽ có tương ứng 255 trường hợp. Nhưng ta chỉ cần biết là có 15 trường hợp cơ bản (**hình 1.3**), sau đó với mỗi trường hợp cơ bản ta có thể xoay đổi góc để tạo ra trường hợp khác. Ta có thể quản lý hết 255 trường hợp với mảng bit gồm 8 vị trí tương ứng với 8 điểm của khối lập phương.



Hình 1.3 Các trường hợp cơ bản và phương pháp xác định các trường hợp của khối lập phương (3)

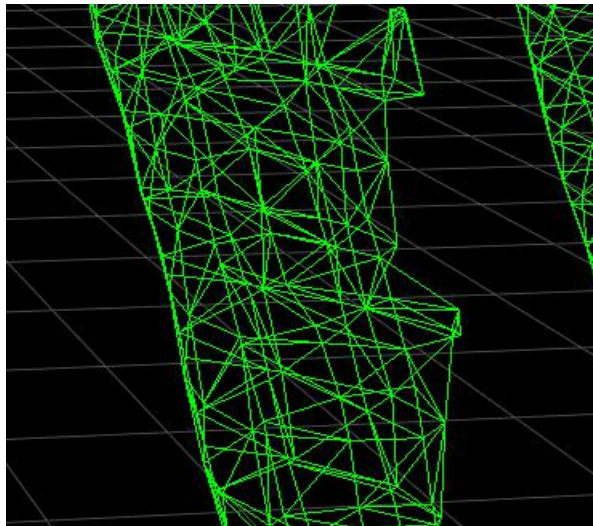
Tương ứng với từng trường hợp sẽ sinh ra số hình tam giác và số điểm tương ứng. Tính toán tương ứng với 8 tọa độ của khối lập phương sẽ tính toán được tọa độ các điểm đó trong không gian 3 chiều, cùng với thông số các điểm nào sẽ nối với nhau để tạo nên hình tam giác sẽ tạo ra được mảng các điểm trong không gian 3 chiều (vertices) và mảng các mặt phẳng (faces), khi hiển thị tất cả các điểm và mặt phẳng này trên không gian 3 chiều sẽ cho ra hình ảnh tương đối giống vật thể thực tế, và khi phóng to ta sẽ thấy nó là sự kết hợp khéo léo của các hình tam giác và các điểm, đường thẳng như **hình 1.4**.

2.4 Phương thức *marching_cubes* của thư viện *skimage.measure*

Phương thức *marching_cubes* áp dụng thuật toán *marching cubes* trích xuất bề mặt lưới 2D thành một khối 3D.

Đầu vào gồm: *volumn* là mảng 3 chiều dữ liệu ảnh, *level* là giá trị đường bao để tìm kiếm các mặt đẳng lập trong khối (nếu không có sẽ tự động chọn giá trị trung bình), *allow_degenerate* là có cho phép suy biến hay không.

Đầu ra gồm: *verts* là một mảng tọa độ các điểm trong không gian, *faces*: mảng xác định các mặt tam giác thông qua tham chiếu chỉ số đỉnh của tập *verts*, ngoài ra còn có tập *normals* và *values* hiện không được sử dụng.



Hình 1.4 Hình ảnh khi phóng to của khối sau khi được tái tạo sử dụng *marching_cubes*

2.5 Hiển thị bằng thư viện *pyqtgraph.opengl*

Hệ thống đồ họa 3D trong *pyqtgraph* bao gồm khu vực hiển thị và một số mục đồ họa có thể được tích hợp vào.

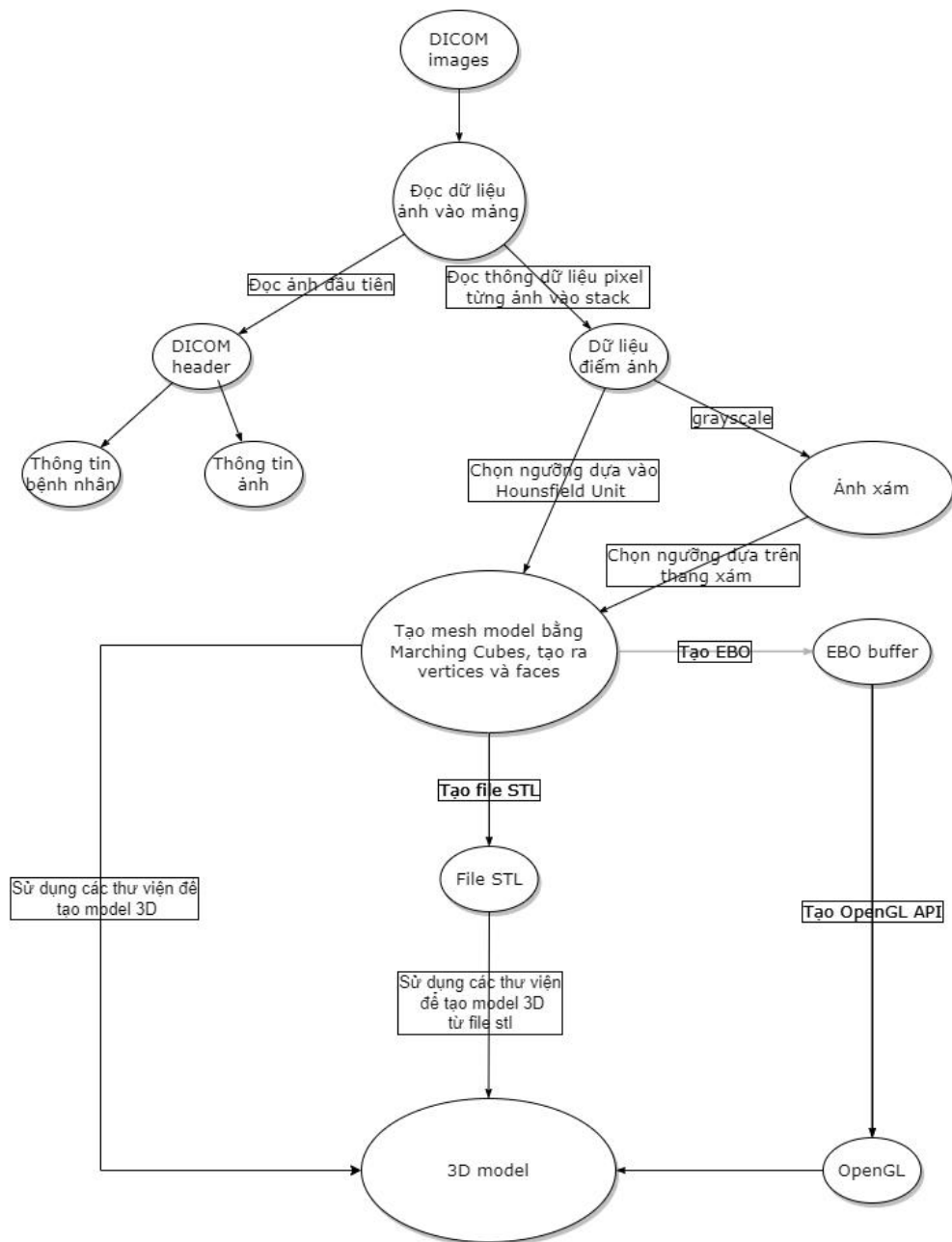
Trong đó có *MeshData* là lớp lưu trữ các dữ liệu trong không gian như điểm ảnh, đường thẳng, mặt phẳng, màu sắc. Và *GLMeshItem* là mục đồ họa dùng để hiển thị lưới tam giác 3D.

CHƯƠNG 2

THIẾT KẾ VÀ CÀI ĐẶT

1. Thiết kế hệ thống

Quá trình thiết kế hệ thống chuyển đổi từ ảnh DICOM đến hiển thị mô hình trong không gian ba chiều sẽ được thực hiện như hình 2.1.



Hình 2.1 Sơ đồ các bước thiết kế hệ thống

Phần mềm đọc bộ ảnh DICOM mà người dùng chọn. Đưa các dữ liệu ảnh đọc được vào mảng ba chiều nhằm xác định vị trí tương đối của điểm ảnh trong không gian ba chiều với các điểm ảnh khác.

Song song đó dữ liệu đọc được từ bộ ảnh DICOM còn cho biết thông tin về bệnh nhân và các thông số ảnh. Dữ liệu điểm ảnh sẽ có tổ chức đồ sự phân bố các điểm ảnh, phần lớn là rất cao do đây là chỉ số Hounsfield Unit, có thể chuyển đổi sang thước xám để thu hẹp ngưỡng ảnh.

Chọn một số nằm trong ngưỡng ảnh để thực hiện thuật toán Marching cubes dựa trên chỉ số ngưỡng ảnh đó. Thuật toán Marching cubes sẽ tạo ra 2 mảng gồm thông số điểm ảnh trong không gian ba chiều gọi là vertices và thông số chỉ số xác định các điểm ảnh sẽ nối với nhau tạo thành hình tam giác gọi là faces-các mặt của các tam giác (hay indices-mảng các chỉ số).

Với hai mảng này, mô hình ba chiều hiển thị bằng ba cách: có thể chuyển thẳng dữ liệu hai mảng vào hiển thị trong không gian ba chiều, lưu lại dữ liệu hai mảng vào tệp STL, tệp này có thể đọc và dùng giống như hai mảng kia, hoặc với 2 mảng trên ta có thể tạo EBO buffer, một loại buffer trong thư viện đồ họa OpenGL giúp tương tác nhanh với phần cứng đồ họa.

Và cuối cùng sẽ hiển thị mô hình trong không gian ba chiều.

2. Cài đặt giải thuật

2.1 Đọc ảnh DICOM

Sử dụng thư viện pydicom để đọc dữ liệu từ các tệp tin ảnh DICOM, dùng hàm `read_file` của pydicom với tham số là tên từng tệp tin ảnh DICOM.

Với mỗi thông tin lấy được mỗi tệp tin ảnh sẽ được lưu vào một danh sách, với mỗi phần tử đều sẽ có thể trích lấy thông tin của bệnh nhân hay thông tin ảnh, ví dụ muốn lấy tên bệnh nhân ta có thể dùng thuộc tính `PatientName` của các phần tử trên.

Thuộc tính `pixel_array` của các phần tử trong danh sách sẽ trả về pixel data với dạng ndarray, các pixel data sẽ được lưu lại bằng ngăn xếp để tạo thành danh sách toàn bộ điểm ảnh.

```
def load_dicom(self, file_path):
    file_path = '\\'.join(file_path.split('/')[0:-1])
    slices = [dicom.read_file(file_path + '/' + s, force=True) for s in os.listdir(file_path)]
    self.dicom_header = slices[0]
    self.images = np.stack([s.pixel_array for s in slices])
```

Hình 2.2 Hàm đọc ảnh DICOM

2.2 Chuyển đổi sang thang mức xám

Ta có thể chuyển đổi sang thang mức xám 0-255

Theo công thức: điểm ảnh $n = (\text{điểm ảnh } n / \text{điểm ảnh lớn nhất}) * 255$

```
def grayscale(imgs):
    # Convert pixel_array (img) to -> gray image (img_2d_scaled)
    ## Step 1. Convert to float to avoid overflow or underflow losses.
    imgs_2d = imgs.astype(float)

    ## Step 2. Rescaling grey scale between 0-255
    imgs_2d_scaled = (np.maximum(imgs_2d, 0) / imgs_2d.max()) * 255.0

    ## Step 3. Convert to uint
    imgs_2d_scaled = np.uint8(imgs_2d_scaled)

    return imgs_2d_scaled
```

Hình 2.3 Hàm chuyển đổi sang ảnh xám

2.3 Marching cubes

Sử dụng hàm marching_cubes của thư viện skimage, với 3 tham số:

Dữ liệu ảnh đã qua xử lý, ngưỡng dùng để tạo cubes khi các điểm ảnh gần với ngưỡng này, cho phép suy biến hay không.

```
def do_marching_cubes(self, threshold, read=False, save=False):
    self.vertices, self.indices, norm, val = measure.marching_cubes(self.processed_images,
                                                                    threshold, allow_degenerate=True)
```

Hình 2.4 Hàm gọi thuật toán marching cubes

Sau đó sẽ nhận được 2 danh sách quan trọng đó là vertices và indices (hoặc gọi là faces)

2.4 Hiển thị

Sử dụng lớp pyqtgraph.opengl để hiển thị mô hình ba chiều

Với các tham số:

Dữ liệu lưới được tạo thành từ điểm ảnh (vertices) và dữ liệu các mặt phẳng (faces) với hàm MeshData

Độ mịn, vẽ các mặt của các khối, vẽ các đường viền các khối, màu của viền

```
def show(self, filename=None, vertices=None, indices=None, edgeColor = (0, 1, 0), MODE = 'stl'):
    print('Show time')
    if self.currentMesh:
        self.viewer.removeItem(self.currentMesh)

    if MODE == 'stl':
        points, faces = self.loadSTL(filename)
    else:
        points = vertices
        faces = indices

    meshdata = gl.MeshData(vertexes=points, faces=faces)
    mesh = gl.GLMeshItem(meshdata=meshdata, smooth=True, drawFaces=False, drawEdges=True, edgeColor=edgeColor + (1,))
    self.viewer.addItem(mesh)

    self.currentMesh = mesh
```

Hình 2.5. Hàm hiển thị mô hình ba chiều

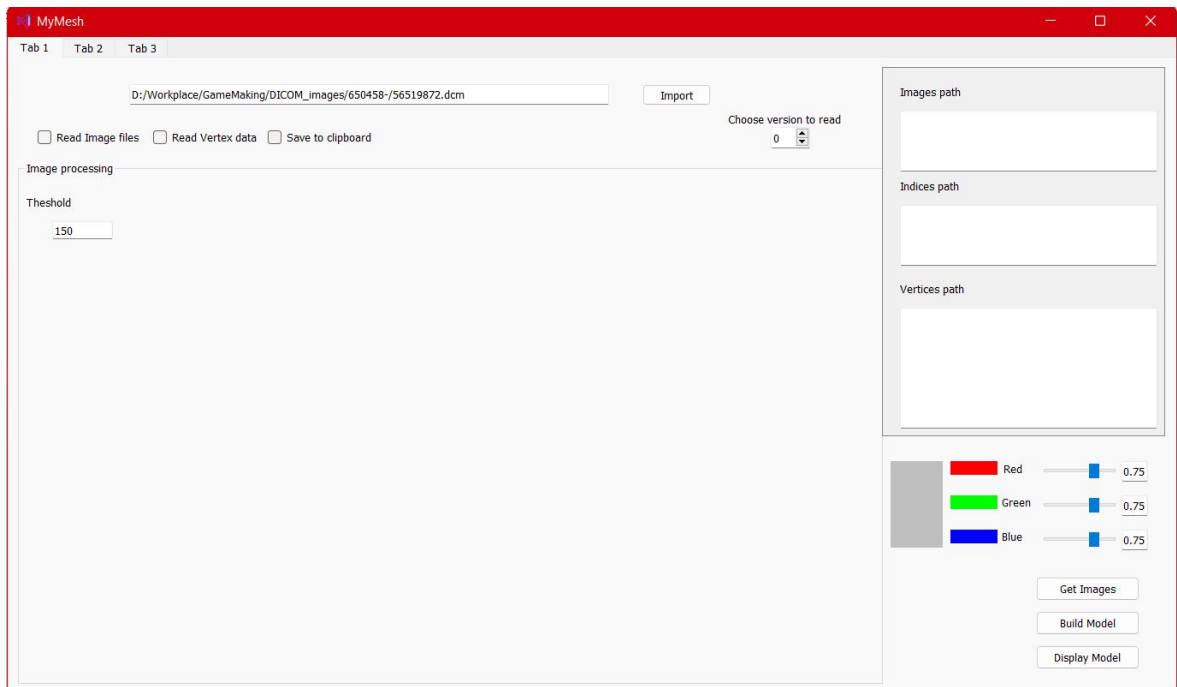
CHƯƠNG 3

KIỂM THỬ VÀ ĐÁNH GIÁ

1. Kết quả kiểm tra

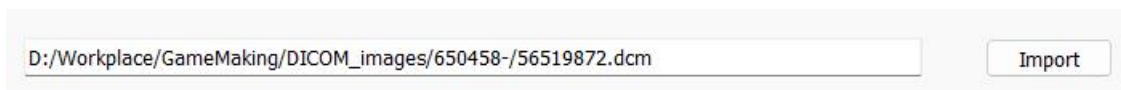
1.1 Giao diện

Giao diện chính, cửa sổ giao diện sẽ gồm 3 thẻ. Thẻ đầu tiên sẽ chứa các thành phần chính là nơi nhập hoặc chọn tệp tin DICOM, chọn ngưỡng và màu sắc cho mô hình. Thẻ 2 sẽ là nơi hiển thị ba chiều của dữ liệu ảnh DICOM. Thẻ 3 sẽ là nơi hiển thị mô hình trên nền 3 chiều.

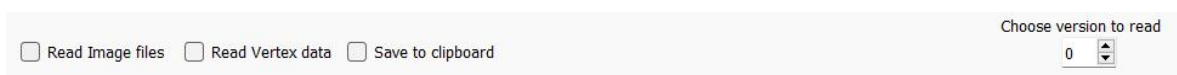


Hình 3.1 Giao diện chính

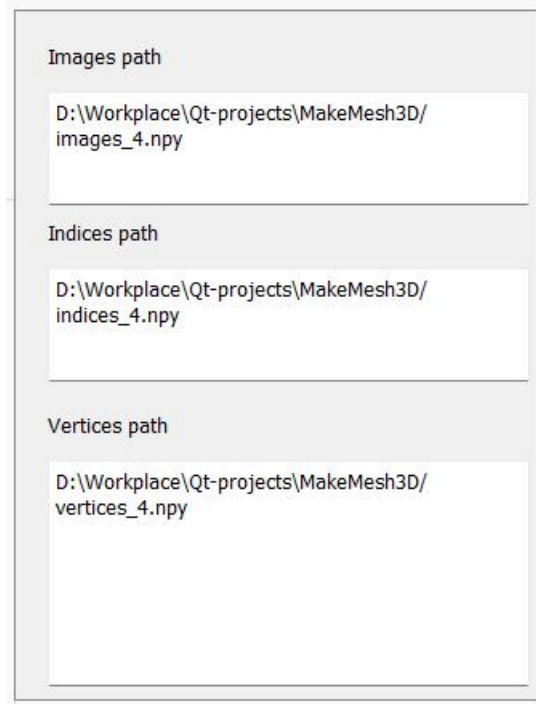
Chi tiết trong thẻ 1



Hình 3.2 Nhập đường dẫn đến tệp tin DICOM



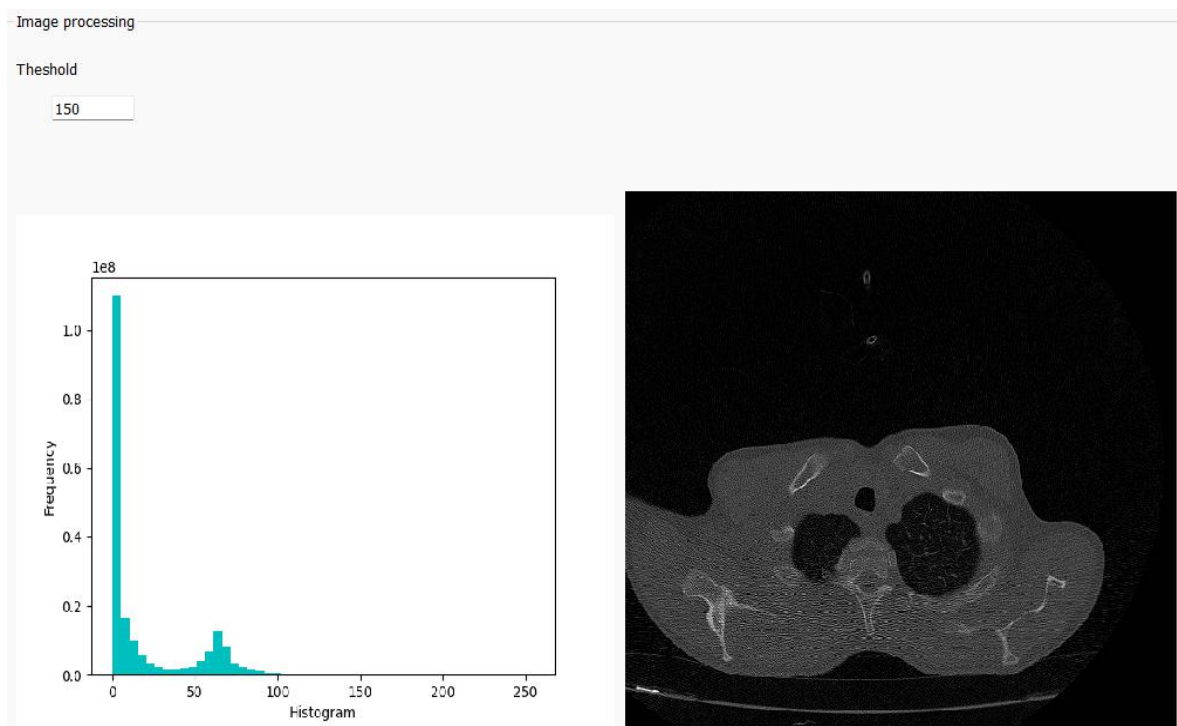
Hình 3.3 Các chức năng cho phép đọc tệp tin ảnh có sẵn, đọc dữ liệu điểm trong không gian 3 chiều đã qua xử lý trước và lưu dữ liệu. Và nơi chọn phiên bản để đọc



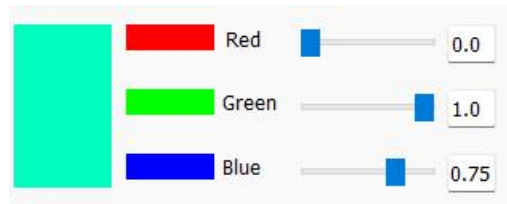
The screenshot shows a software window with three sections, each with a label and a text input field containing a file path:

- Images path**: D:\Workplace\Qt-projects\MakeMesh3D\images_4.npy
- Indices path**: D:\Workplace\Qt-projects\MakeMesh3D\indices_4.npy
- Vertices path**: D:\Workplace\Qt-projects\MakeMesh3D\vertices_4.npy

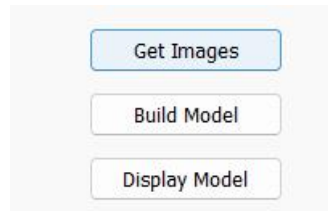
Hình 3.4 Đường dẫn đến các tệp tin được lưu khi dùng chức năng save to clipboard



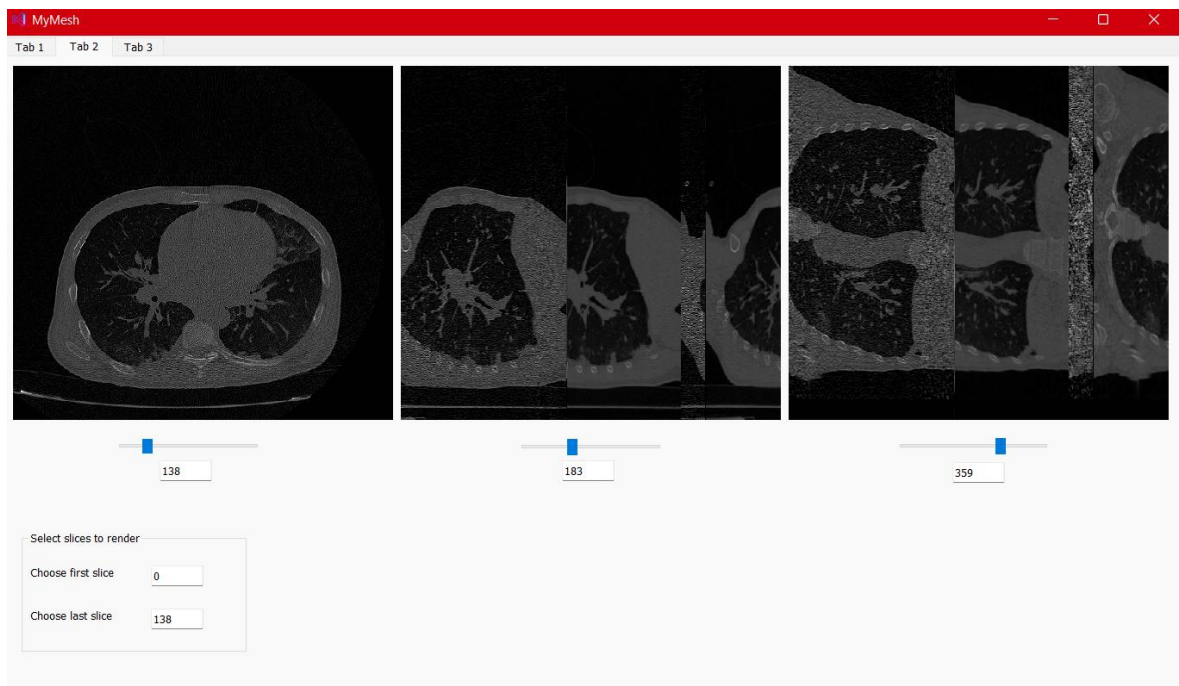
Hình 3.5 Nơi nhập ngưỡng để thực hiện marching cubes, hình ảnh lược đồ xám và ảnh đầu tiên của bộ ảnh DICOM



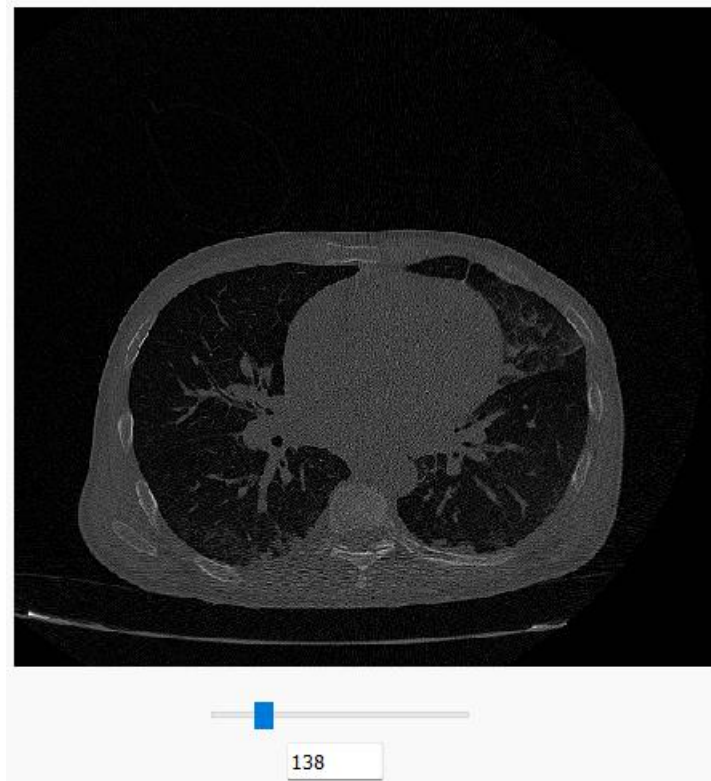
Hình 3.6 Chuyển đổi màu sắc cho mô hình ba chiều khi hiển thị



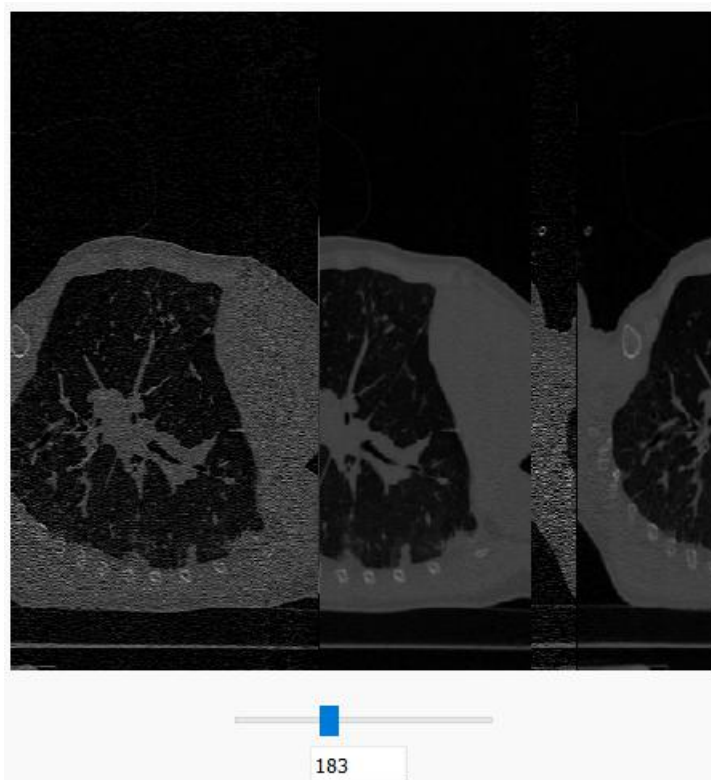
Hình 3.7 Các nút chức năng lấy dữ liệu tệp tin DICOM, tạo mô hình, hiển thị mô hình
Chi tiết trong thẻ 2



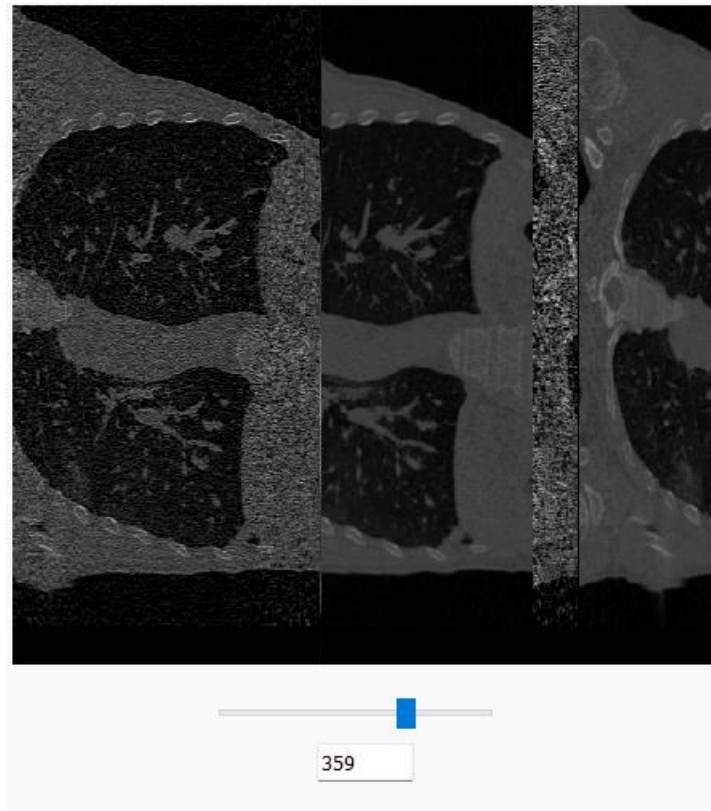
Hình 3.8 Giao diện của thẻ 2



Hình 3.9 Ảnh DICOM ở mặt phẳng ngang



Hình 3.10. Ảnh DICOM ở mặt phẳng dọc



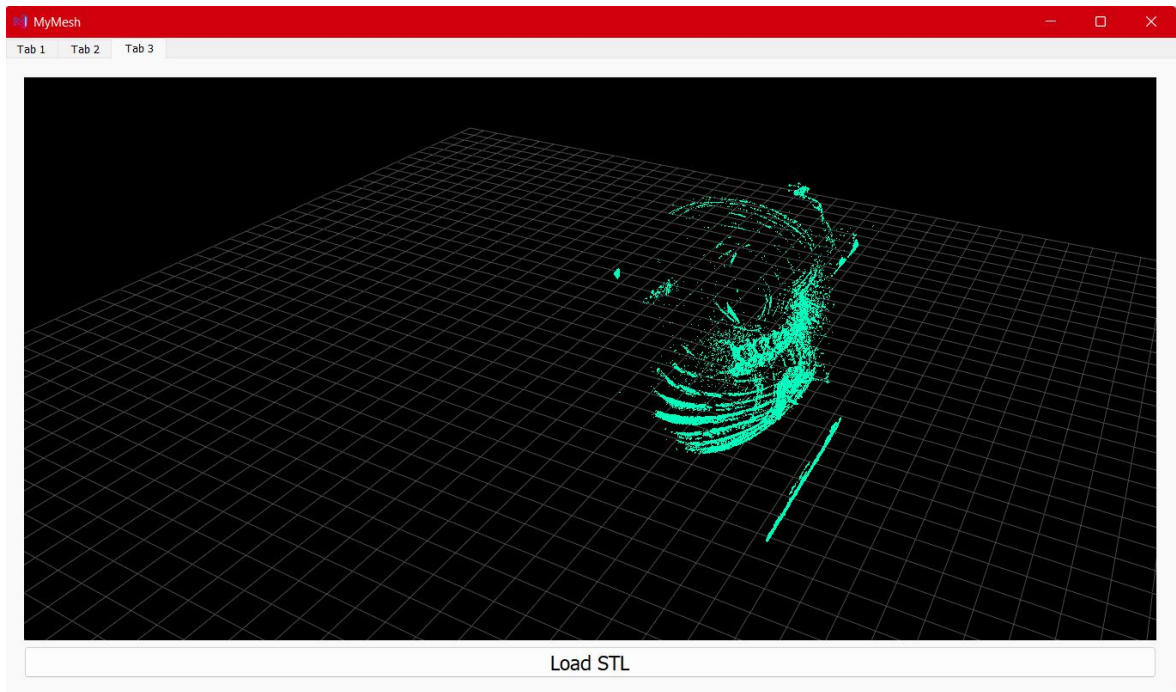
Hình 3.11 Ảnh DICOM ở mặt phẳng từ trên xuống

Select slices to render

Choose first slice	<input type="text" value="0"/>
Choose last slice	<input type="text" value="138"/>

Hình 3.12 Lựa chọn khoảng ảnh sẽ được dựng mô hình, dựa vào ảnh ở mặt phẳng ngang

Chi tiết ở thẻ 3



Hình 3.13 Giao diện của thẻ 3

Ở giao diện cuối, là giao diện hiển thị mô hình trong không gian 3 chiều, mô hình sau khi được xây dựng bằng thuật toán marching cubes có thể được quan sát ở giao diện này. Mô hình có thể được thao tác bằng con trỏ máy tính như phóng to, thu nhỏ, xoay.

Ngoài ra, còn chức năng load stl sẽ giúp ta tải nhưng tệp tin stl vào để quan sát trong giao diện này, tệp tin stl có thể được trực tiếp kéo thả vào giao diện.

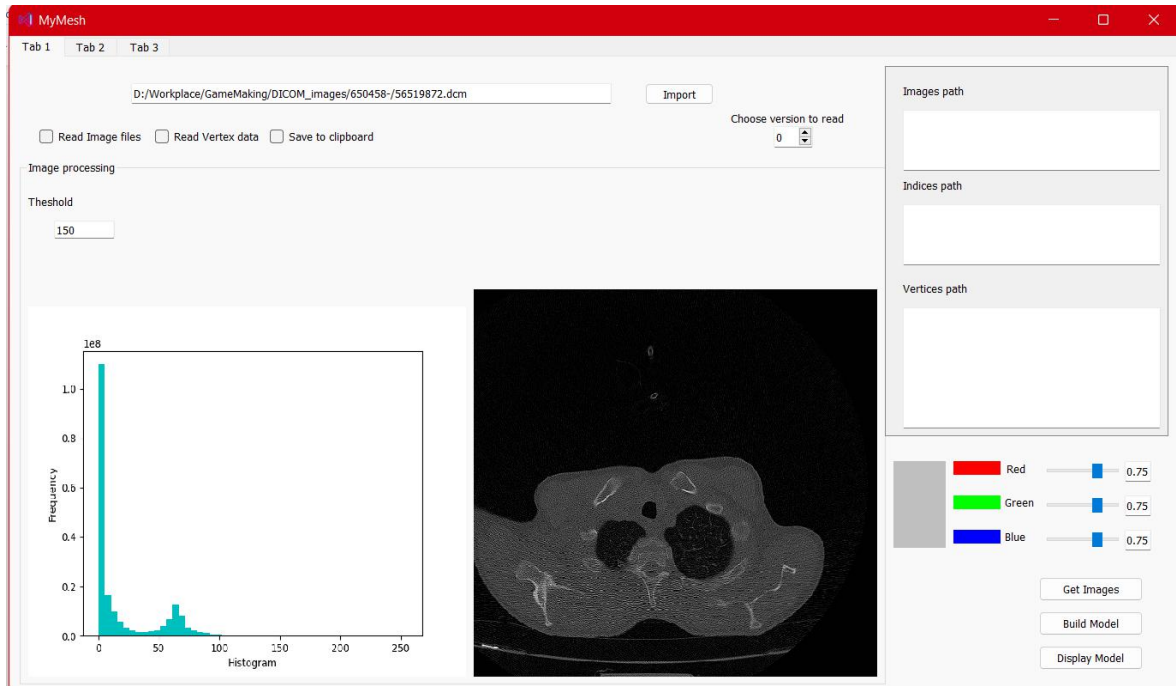
1.2 Kiểm thử

1.2.1 Kịch bản 1: Có sẵn thư mục DICOM

Chọn “import” ở hình 3.2, chọn đến thư mục chứa các tệp tin DICOM, chọn tệp đầu tiên và nhấn “Open”

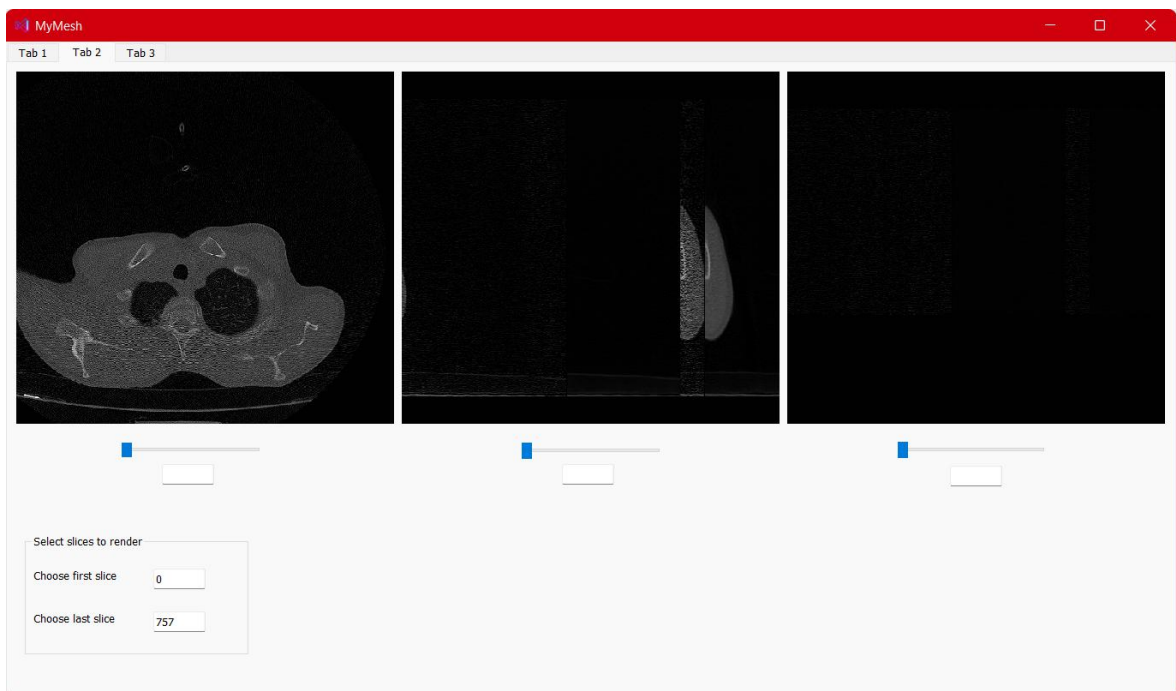
Nhấp vào “Get Images” ở hình 3.6, và đợi cho ảnh lược đồ xám và ảnh đầu tiên của bộ ảnh DICOM xuất hiện ở hình 3.5

Đề tài: Tái tạo 3D các ảnh CT ngực sử dụng thuật toán marching cubes



Hình 3.14 Giao diện của thẻ 1 sau kiểm thử 1

Nhấn qua thẻ 2 bằng nút “tab 2” trên giao diện chính hình 3.1



Hình 3.15 Giao diện của thẻ 2 sau kiểm thử 1

Nhập vào ô “first slice” là 353 và ô “last slice” là 496 ở hình 3.8

Quay lại giao diện thẻ 1

Chọn mức ngưỡng là 100 ở hình 3.5

Kéo thanh màu red = 0.0, green = 1.0, blue = 0.75 ở hình 3.6

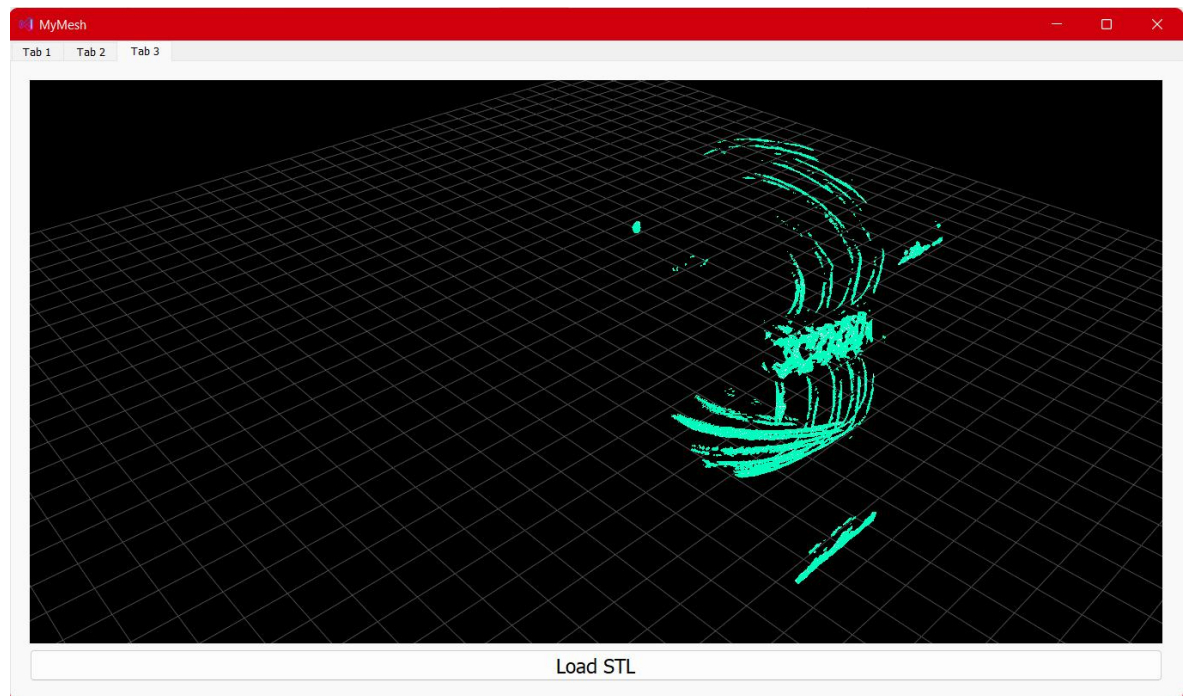
Nhấp vào “Build Model” ở hình 3.7

Sau đó nhấp vào “Display Model” ở hình 3.7

Nhấn qua thẻ 3 bằng nút “tab 3” trên giao diện chính hình 3.1

Đợi một thời gian

Đây sẽ là kết quả



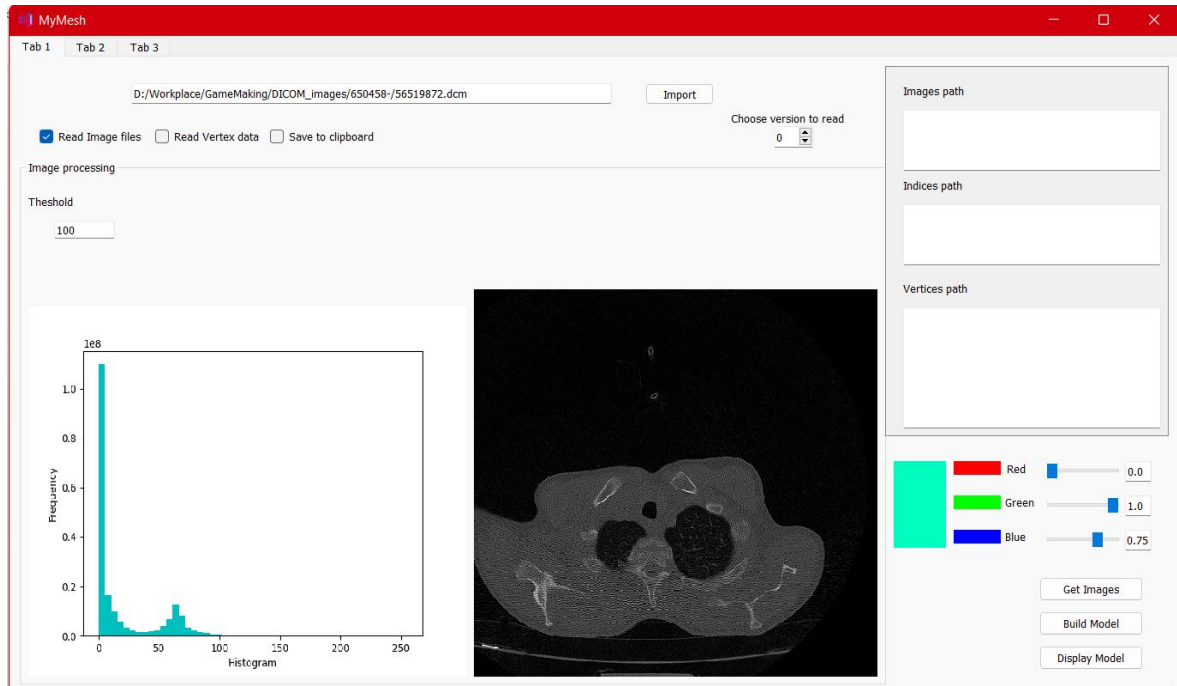
Hình 3.16 Giao diện của thẻ 3 sau kiểm thử 1

1.2.2 Kịch bản 2: Có sẵn tệp tin chứa dữ liệu ảnh DICOM

Chọn “Read Image files” và chọn phiên bản đã lưu là 0 ở hình 3.3

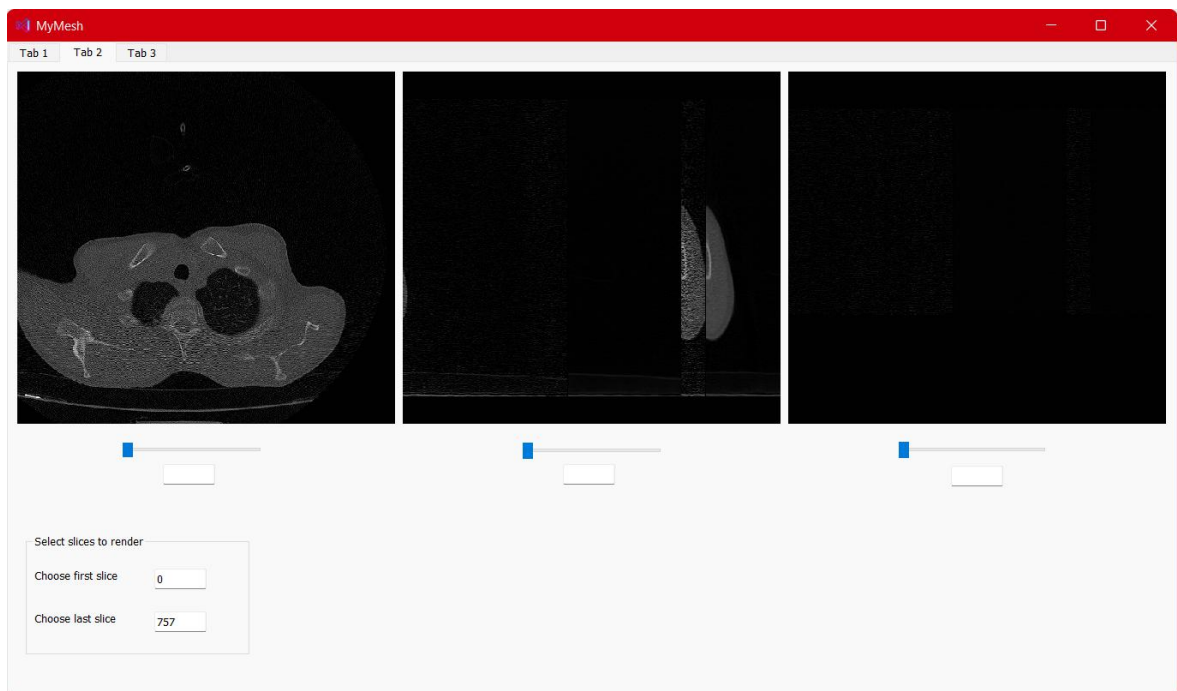
Nhấp vào “Get Images” ở hình 3.6, và đợi cho ảnh lược đồ xám và ảnh đầu tiên của bộ ảnh DICOM xuất hiện ở hình 3.5

Đề tài: Tái tạo 3D các ảnh CT ngực sử dụng thuật toán marching cubes



Hình 3.17 Giao diện của thẻ 1 sau kiểm thử 2

Nhấn qua thẻ 2 bằng nút “tab 2” trên giao diện chính hình 3.1



Hình 3.18 Giao diện của thẻ 2 sau kiểm thử 2

Nhập vào ô “first slice” là 0 và ô “last slice” là 199 ở hình 3.8

Quay lại giao diện thẻ 1

Chọn mức ngưỡng là 100 ở hình 3.5

Kéo thanh màu red = 0.0, green = 1.0, blue = 0.75 ở hình 3.6

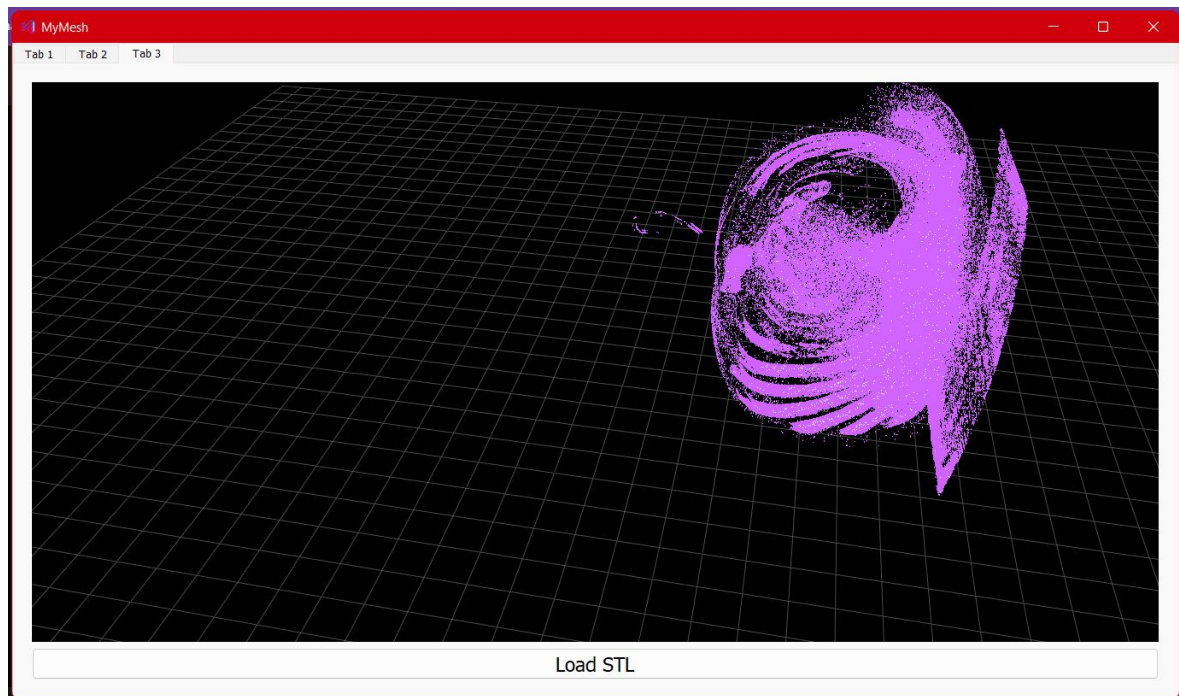
Nhấp vào “Build Model” ở hình 3.7

Sau đó nhấp vào “Display Model” ở hình 3.7

Nhấn qua thẻ 3 bằng nút “tab 3” trên giao diện chính hình 3.1

Đợi một thời gian

Đây sẽ là kết quả



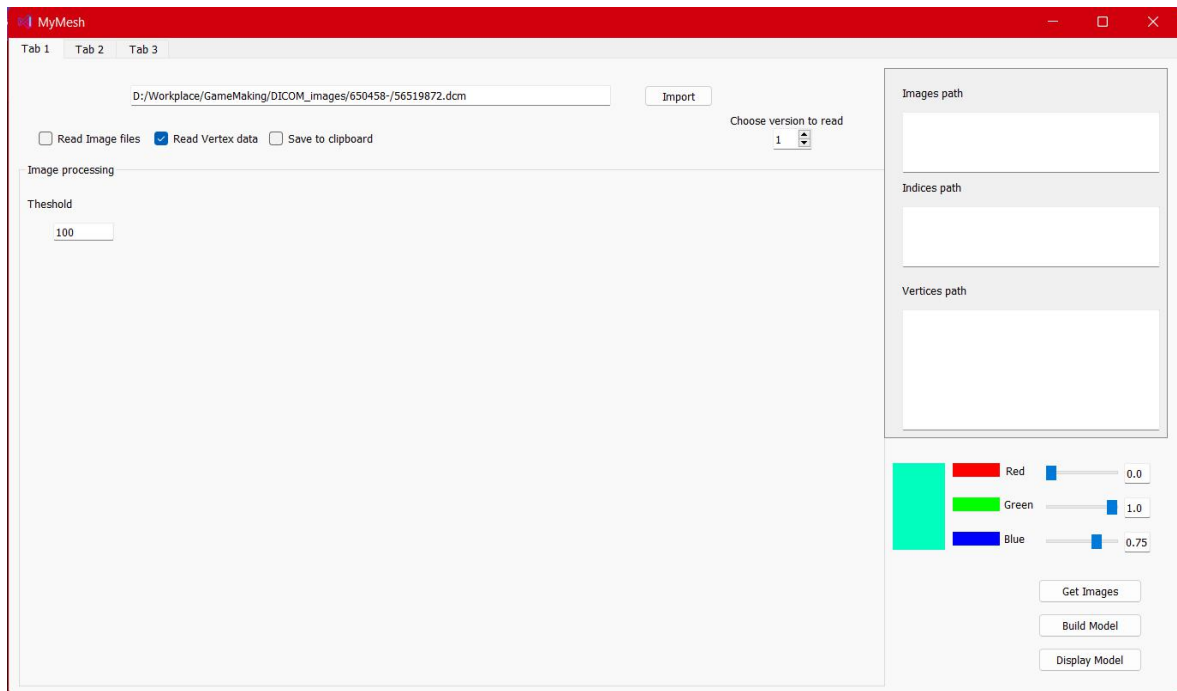
Hình 3.19 Giao diện của thẻ 3 sau kiểm thử 2

1.2.3 Kịch bản 3: Có sẵn tệp tin chứa dữ liệu điểm ảnh đã qua xử lý bằng thuật toán Marching cubes

Chọn “Read Vertex data” và chọn phiên bản đã lưu là 1 ở hình 3.3

Chọn mức ngưỡng là 100 ở hình 3.5

Kéo thanh màu red = 0.0, green = 1.0, blue = 0.75 ở hình 3.6



Hình 3.20 Giao diện của thẻ 1 sau kiểm thử 3

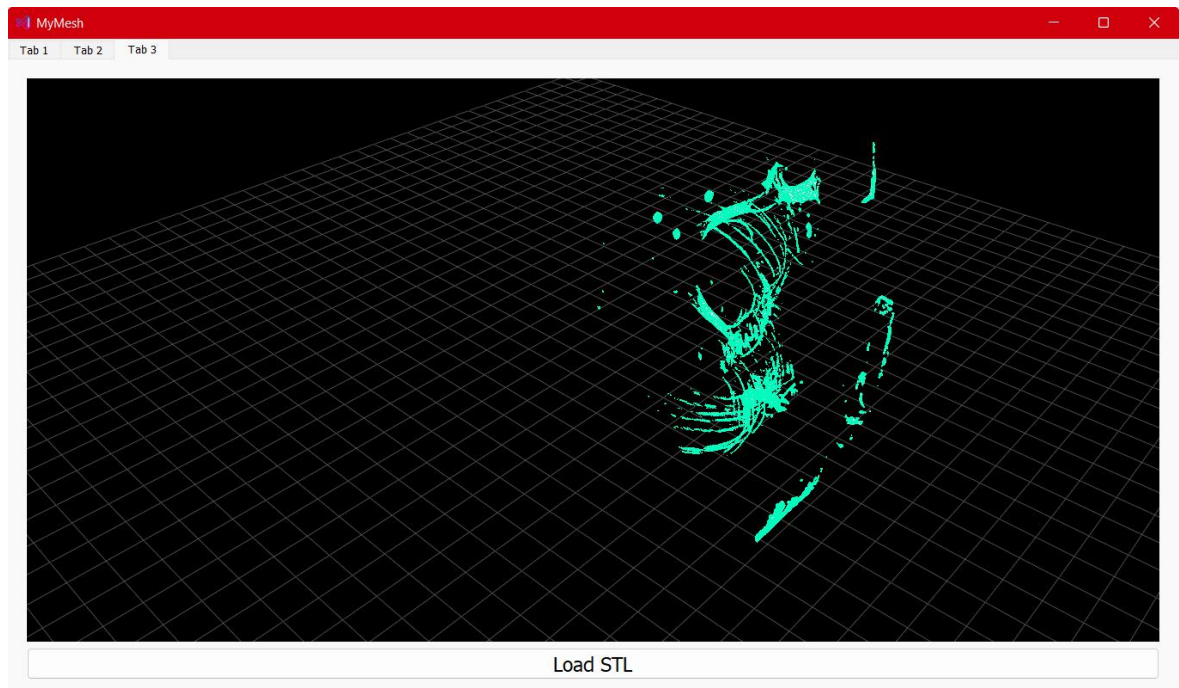
Nhấp vào “Build Model” ở hình 3.7

Sau đó nhấp vào “Display Model” ở hình 3.7

Nhấn qua thẻ 3 bằng nút “tab 3” trên giao diện chính hình 3.1

Đợi một thời gian

Đây sẽ là kết quả



Hình 3.21 Giao diện của thẻ 3 sau kiểm thử 3

1.3 Phân tích các bộ ảnh DICOM đã qua quan sát và kiểm thử

Các mẫu tệp tin DICOM được dùng để nghiên cứu có thể được dùng tải và truy cập tại kho dữ liệu công khai của Harvard

Đối tượng	Số lượng ảnh	Bộ phận được chụp	Ưu điểm	Nhược điểm	Nhận xét kết quả
Đối tượng 1	427	Ngực	Đầy đủ lát cắt của ngực	Có sự lặp lại làm số lượng tăng gấp nhiều lần so với số lát cắt ảnh thật	Ở số ảnh lặp lần đầu có xuất hiện nhiều điểm nhiễu khiến mô hình sinh ra không chính xác
Đối tượng 2	758	Ngực	Đầy đủ lát cắt của ngực	Có sự lặp lại làm số lượng tăng gấp nhiều lần so với số lát cắt ảnh thật	Mô hình được tạo ra hiển thị rõ các bộ phận như xương, phổi
Đối tượng 3	426	Ngực	Đầy đủ lát cắt của ngực	Có sự lặp lại làm số lượng tăng gấp nhiều lần so với số lát cắt ảnh thật	Mô hình được tạo ra hiển thị rõ các bộ phận như xương, phổi

Bảng 1.2 Bảng so sánh các mẫu DICOM được dùng (8)

PHẦN KẾT LUẬN

1. Kết quả đạt được

Phần mềm đọc ảnh y tế với định dạng DICOM được xây dựng với ngôn ngữ python. Ảnh định dạng DICOM đã có thể được đọc và tái cấu trúc thành dạng khung. Với dạng khung này dữ liệu có thể tạo cái tệp định dạng STL và hiển thị trực tiếp trên không gian ba chiều có các tương tác cơ bản như xoay và thu phóng.

2. Hướng phát triển

Nguyên cứu thêm về các thuật toán tái tạo bề mặt khác để so sánh và cải thiện tốc độ tái tạo mô hình ba chiều. Thêm vào các phương pháp hậu xử lý mô hình giúp hiển thị một cách phân biệt và rõ nét hơn ở từng cơ quan, bộ phận của ảnh. Áp dụng máy học cho phép học hỏi nhận diện vật thể lạ, phát hiện tế bào ung thư, chuẩn đoán bằng trí tuệ nhân tạo.

TÀI LIỆU THAM KHẢO

1. Aziz Fajar, Riyanarto Sarno, Chastine Fatichah và Achmad Fahmi (2020), Reconstructing and resizing 3D images from DICOM files
2. Chak Chi Sio, Muhua Ngan, Jin Yi, Xuxin Chen (2010), Volume Rendering with Marching Cube Algorithm
3. K.Tzevanidis, X. Zabulis, T. Sarmis, P. Koutlemanis, N. Kyriazis, and A. Argyros (2010), From multiple views to textured 3D meshes: a GPU-powered approach
4. T.Senthil Kumara , Anupa Vijai, et al. (2011), 3D Reconstruction of Face from 2D CT Scan Images
5. Yvette Brazier (2018), How does a CT or CAT scan work? article on Medical News Today
6. Adrian Schneider (2017), <https://github.com/AOT-AG/DicomToMesh>
7. Singular Health Pty Ltd (2020), 3Dicom Viewer is a product/service developed and delivered by Singular Health Pty Ltd, part of the Singular Health Group
8. Mostafavi SM. COVID19-CT-dataset: an open-access chest CT image repository of 1000+ patients with confirmed covid-19 diagnosis. Harv Dataverse. 2021. <https://doi.org/10.7910/DVN/6ACUZJ>.