

ĐẠI HỌC THÁI NGUYÊN
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG

PHẠM THỊ LAN ANH

HIỂN THỊ ẢNH DICOM
TRONG Y TẾ THEO THÀNH PHẦN

LUẬN VĂN THẠC SĨ KHOA HỌC MÁY TÍNH

THÁI NGUYÊN, 2017

ĐẠI HỌC THÁI NGUYÊN
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG

PHẠM THỊ LAN ANH

**HIỂN THỊ ẢNH DICOM
TRONG Y TẾ THEO THÀNH PHẦN**

Chuyên ngành: Khoa học máy tính

Mã số: 60 48 01 01

LUẬN VĂN THẠC SĨ KHOA HỌC MÁY TÍNH

Người hướng dẫn khoa học: PGS.TS ĐỖ NĂNG TOÀN

THÁI NGUYÊN, 2017

LỜI CẢM ƠN

Trong quá trình học tập và nghiên cứu tại lớp Cao học khóa 14 chuyên ngành Khoa học máy tính tại Trường Công nghệ thông tin và truyền thông - Đại học Thái Nguyên, em đã nhận được rất nhiều sự giúp đỡ nhiệt tình của các thầy, cô giáo trong Trường Công nghệ thông tin và Viện công nghệ thông tin Việt Nam. Các thầy, cô luôn giúp đỡ, tạo điều kiện cho em trong quá trình học tập. Em xin bày tỏ lời cảm ơn chân thành tới tập thể các thầy, cô giáo trong Trường Công nghệ thông tin và Viện công nghệ thông tin Việt Nam.

Em xin chân thành cảm ơn tới thầy giáo **PGS.TS Đỗ Năng Toàn** đã hướng dẫn và tạo điều kiện giúp đỡ em hoàn thành luận văn tốt nghiệp này.

Xin cảm ơn các đồng nghiệp và người thân đã động viên, giúp đỡ trong quá trình nghiên cứu và thực hiện luận văn này.

Quá trình thực hiện đề tài không tránh khỏi thiếu sót em rất mong nhận được sự đóng góp ý kiến của các thầy, cô giáo và các bạn đồng nghiệp đối với đề tài nghiên cứu của em để đề tài được hoàn thiện hơn.

Em xin trân trọng cảm ơn!

Thái Nguyên, ngày 19 tháng 5 năm 2017

LỜI CAM ĐOAN

Em xin cam đoan, toàn bộ nội dung liên quan tới đề tài được trình bày trong luận văn là bản thân em tự tìm hiểu và nghiên cứu, dưới sự hướng dẫn khoa học của Thầy giáo **PGS. TS ĐỖ NĂNG TOÀN**

Các tài liệu, số liệu tham khảo được trích dẫn đầy đủ nguồn gốc. Em xin chịu trách nhiệm trước pháp luật lời cam đoan của mình.

Thái Nguyên, ngày 19 Tháng 5 năm 2017

Học viên thực hiện

Phạm Thị Lan Anh

MỤC LỤC

MỞ ĐẦU	i
1. Sự cần thiết của đề tài	1
2. Đối tượng và phạm vi nghiên cứu	2
3. Hướng nghiên cứu của đề tài	2
4. Những nội dung nghiên cứu chính	3
5. Phương pháp nghiên cứu	3
6. Ý nghĩa khoa học của đề tài	3
CHƯƠNG 1: TỔNG QUAN VỀ CHUẨN DICOM VÀ BÀI TOÁN HIỂN THỊ ẢNH Y TẾ	4
1.1. Cấu trúc dữ liệu ảnh DICOM	4
1.1.1. Giới thiệu chung	4
1.1.2. Phạm vi và lĩnh vực ứng dụng của DICOM	5
1.1.3. Cấu trúc của chuẩn ảnh DICOM	6
1.2. Bài toán hiển thị ảnh	14
1.2.1. Đặt vấn đề	14
1.2.2. Phương pháp nghiên cứu	17
CHƯƠNG 2: MỘT SỐ KỸ THUẬT HIỂN THỊ ẢNH DICOM	19
2.1. Kỹ thuật nội suy trong xây dựng lát cắt ảnh	19
2.1.1. Kỹ thuật nội suy các điểm ảnh láng giềng gần nhất	19
2.1.2. Kỹ thuật nội suy tuyến tính	23
2.1.3. Kỹ thuật nội suy song khối	25
2.2. Kỹ thuật Volume Rendering	27
2.2.1. Volume Rendering	27
2.2.2. Texture-Based Volume Rendering	30
2.2.3. Chi tiết thực hiện	33
2.2.4. Kỹ thuật nâng cao	39

CHƯƠNG 3: CHƯƠNG TRÌNH THỬ NGHIỆM HIỂN THỊ ẢNH Y TẾ DICOM THEO THÀNH PHẦN.....	43
3.1. Phân tích yêu cầu bài toán.....	43
3.2. Phân tích lựa chọn công cụ	44
3.3. Quy trình thực hiện	49
3.4. Một số kết quả chương trình	50
KẾT LUẬN	52
TÀI LIỆU THAM KHẢO	54

DANH MỤC CÁC HÌNH ẢNH

Hình 1.1. Định dạng file DICOM.....	7
Hình 1.2. Thông tin đính kèm file DICOM.....	8
Hình 1.3. Cấu tạo Data Set	10
Hình 1.4. Lọc nhiễu ảnh y tế DICOM	16
Hình 1.5. Hiện thị 3D hình ảnh y tế bằng kỹ thuật Render Volume	16
Hình 1.6. Hiện thị ảnh DICOM bằng công nghệ hiện thị 2D,3D.....	16
Hình 2.1. Các hiệu ứng dạng Volumetric	19
Hình 2.2. Ví dụ nội suy điểm trên không gian 1 chiều.....	20
Hình 2.3. Kết quả nội suy láng giềng gần nhất trong không gian 1 chiều	21
Hình 2.4. Phóng to ảnh kích thước 2x2 thành ảnh 9x9	21
Hình 2.5. Nội suy điểm $Y(J,K)$ dựa trên các điểm ảnh lân cận.....	22
Hình 2.6. Ứng dụng nội suy láng giềng gần nhất trong thay đổi tỷ lệ ảnh CCD	22
Hình 2.7. Mô tả kỹ thuật nội soi song tuyến trong ứng dụng làm tăng độ phân giải của ảnh.....	24
Hình 2.8. Ứng dụng nội suy tuyến tính trong xoay ảnh MRI.....	25
Hình 2.9. Ứng dụng nội suy tuyến tính sinh lát cắt trung gian	25
Hình 2.10. Mô tả kỹ thuật nội suy song khối	26
Hình 2.11. Kết quả nội suy song khối trong trường hợp biến đổi tỷ lệ ảnh CCD.....	27
Hình 2.12. So sánh 3 kỹ thuật nội suy	27
Hình 2.13. Quá trình Volume Rendering	28
Hình 2.14. Khối mẫu và kết hợp.....	29
Hình 2.15. Các bước của một quá trình Texture-Based Volume Rendering ..	30
Hình 2.16. Kết quả thực hiện của ứng dụng hiện thị hình ảnh sử dụng khối ..	33
Hình 2.17. Lát cắt hai đa giác	36
Hình 2.18. Sự khác biệt giữa các hàm truyền 1D và 2D	38

Hình 2.19. Các ví dụ của kết quả với bóng	41
Hình 3.1. Bảng giá trị mức xám có ý nghĩa trong dữ liệu ảnh DICOM chuyển đổi ...	44
Hình 3.2. Sơ đồ tương tác của VTK với phần cứng	45
Hình 3.3. Mô hình đồ họa của VTK	47
Hình 3.4. Mô hình trực quan hóa của VTK	47
Hình 3.5. Các loại tập dữ liệu của VTK	48
Hình 3.6. Sắp xếp dữ liệu	49
Hình 3.7. Hình ảnh của xương	50
Hình 3.8. Hình ảnh của mạch máu	50
Hình 3.9. Hình ảnh của cơ	51
Hình 3.10. Hình ảnh của da	51

DANH MỤC CHỮ CÁI VIẾT TẮT

Ký hiệu/ Chữ ký viết tắt	Viết đầy đủ	Ý nghĩa
DICOM	Digital Imaging and Communications in Medicine Standars	Tiêu chuẩn số và truyền thông trong y tế
CT	Computed Tomography Scanner	Chụp cắt lớp vi tính
MRI	Magnetic Resonance Imaging	Chụp cộng hưởng từ
VR	Value Representation	Giá trị biểu diễn
2D	Two - dimensional	Không gian 2 chiều
3D	Three - dimensional	Không gian 3 chiều
GPU	Graphic Processing	Bộ xử lý đồ họa
MC	Marching Cubes	Thuật toán Marching Cubes
NEMA	National Electrical Manufacturers Association	Hiệp hội các nhà sản xuất
ACR	American Cllege of Radionlogy	Đại học X quang Hoa Kỳ
VTK	Visualizaton toolket	Bộ công cụ Visualizaton
CS	Conformation Statement	Báo cáo thích nghi
SC	Service Classes	Lớp dịch vụ
IOD	Information Object Desfinition	Đối tượng thông tin

MỞ ĐẦU

1. Sự cần thiết của đề tài

Ngày nay trước sự phát triển của công nghệ thông tin, trong mọi lĩnh vực đặc biệt là y tế đang nắm giữ một vai trò vô cùng quan trọng đối với sự phát triển của con người cũng như toàn xã hội. Với việc dân số không ngừng tăng nhanh kèm theo đó là sự xuất hiện của nhiều loại bệnh mới đòi hỏi ngành y tế cần phải nỗ lực hơn nữa trong việc chăm sóc sức khỏe cho cộng đồng. Việc ứng dụng công nghệ thông tin trong ngành y tế mới chỉ dừng lại ở công tác quản lý hành chính và viện phí mà chưa đáp ứng nhiều trong nhu cầu khám chữa bệnh của người dân.

Hiện nay, trong việc quản lý, thu nhận và xử lý thông tin với khối lượng ngày càng lớn, những phần mềm thủ công không đem lại hiệu quả mong muốn, tốn nhiều thời gian và công sức.

Đất nước ngày càng đi lên công nghệ thông tin cũng ngày càng phát triển. Nhiều bệnh viện đã nhanh chóng lắp đặt các hệ thống hiện đại phục vụ trong công tác quản lý đồng thời không ngừng tăng cường đầu tư máy móc công nghệ hiện đại như máy chụp cắt lớp, máy chụp Xquang, chụp cộng hưởng từ để hỗ trợ trong công tác điều trị, khám chữa bệnh. Việc làm này đã góp phần không nhỏ trong việc tìm ra bệnh và điều trị một cách kịp thời.

Ảnh được chụp từ các loại máy này được lưu trữ phục vụ trong lĩnh vực y tế. Các ảnh này được các bác sỹ sử dụng để đọc thông tin phục vụ cho việc chẩn đoán bệnh hay sử dụng để tái khám lại hoặc xem lại những bệnh cũ của bệnh nhân...

DICOM (Digital Imaging and Communications in Medicine) là tiêu chuẩn để xử lý, lưu trữ, in ấn và thu/nhận hình ảnh trong y tế. Tiêu chuẩn này bao gồm cả việc định nghĩa cấu trúc tập tin và giao thức truyền thông tin. Giao thức truyền thông tin là một giao thức ứng dụng sử dụng nền tảng TCP/IP để giao tiếp lẫn nhau giữa các hệ thống. Các tập tin DICOM có thể

được trao đổi lẫn nhau giữa các hệ thống khi các hệ thống này có khả năng thu nhận hình ảnh và dữ liệu bệnh nhân theo định dạng DICOM.

DICOM cho phép việc tích hợp dễ dàng các máy thu nhận hình ảnh, server, trạm làm việc, máy in và các thiết bị phần cứng khác có nối mạng từ các nhà sản xuất khác nhau trong hệ thống truyền thông (PACS).

Từ những thông tin trên, để hiểu rõ được DICOM có ứng dụng như thế nào trong y tế em đã chọn đề tài **“HIỂN THỊ ẢNH DICOM TRONG Y TẾ THEO THÀNH PHẦN”**

2. Đối tượng và phạm vi nghiên cứu

a. Đối tượng

- Xử lý và hiển thị hình ảnh.
- Hiển thị ảnh DICOM trong y tế theo thành phần.

b. Phạm vi

- Các khái niệm, tính chất, phương pháp phân loại, xử lý ảnh và hiển thị ảnh.
- Các kỹ thuật và phương pháp phát hiện thị ảnh DICOM trong y tế.

3. Hướng nghiên cứu của đề tài

- Tìm kiếm các tài liệu trên Internet, thư viện nhà trường để nghiên cứu các vấn đề liên quan nhằm giải quyết đề tài.
- Liên hệ với thầy hướng dẫn trong suốt quá trình làm đề tài đi đúng hướng và có hiệu quả.
- Tìm hiểu tổng qua về hiển thị ảnh DICOM trong y tế theo thành phần, nghiên cứu một số kỹ thuật hiển thị ảnh.
- Tìm hiểu bài toán và tiến hành phân tích
- Kết hợp nghiên cứu với thử nghiệm nhằm giúp cho việc nghiên cứu đúng hướng, có tính thuyết phục cao.
- Sử dụng các kỹ thuật hiển thị ảnh để làm cơ sở triển khai bài toán hiển thị ảnh DICOM trong y tế theo thành phần.

- Lựa chọn ngôn ngữ lập trình phù hợp để thực hiện.

4. Những nội dung nghiên cứu chính

- Tổng quan về chuẩn DICOM và bài toán hiển thị ảnh trong y tế theo thành phần.
- Nghiên cứu các phương pháp hiển thị hình ảnh DICOM trong y tế.
- Cài đặt và thử nghiệm hệ thống.

Các nội dung *dự kiến* thể hiện như sau:

Nội dung luận văn

Chương 1: Tổng quan về chuẩn DICOM và bài toán hiển thị ảnh y tế

Chương 2: Một số kỹ thuật hiển thị ảnh dicom

Chương 3: Chương trình thử nghiệm hiển thị ảnh y tế dicom theo thành phần

Kết luận:

- Tóm tắt những kết quả đạt được và những hạn chế của luận văn
- Hướng phát triển của luận văn trong tương lai.

5. Phương pháp nghiên cứu

- Nghiên cứu các tài liệu và viết tổng quan
- Phương pháp phân tích và thiết kế đối tượng.
- Nghiên cứu triển khai công nghệ và thử nghiệm hệ thống: xây dựng chương trình cụ thể để thử nghiệm, phân tích, đánh giá kết quả đạt được.

6. Ý nghĩa khoa học của đề tài

- Bản thân hiểu sâu hơn và áp dụng được các ứng dụng của xử lý ảnh vào thực tế.
- Hỗ trợ và nâng cao hiệu quả bài toán nhập liệu tự động, làm cơ sở triển khai các ứng dụng khác.

CHƯƠNG 1: TỔNG QUAN VỀ CHUẨN DICOM VÀ BÀI TOÁN HIỂN THỊ ẢNH Y TẾ

1.1. Cấu trúc dữ liệu ảnh DICOM

1.1.1. Giới thiệu chung

Vào năm 1970, trước sự ra đời của phương pháp chụp ảnh CT (Computed Tomography) cùng với các phương pháp chụp ảnh số dùng trong chẩn đoán y khoa khác, và sự gia tăng nhanh chóng ứng dụng tin học trong các lĩnh vực y khoa lâm sàng, hai tổ chức ACR (American College of Radiology) và NEMA (National Electrical Manufacturers Association) đã nhận ra yêu cầu cần thiết phải có một phương pháp chuẩn dùng trong truyền tải ảnh và thông tin liên quan đến ảnh đó giữa các nhà sản xuất thiết bị y khoa, mặc dù những thiết bị đó lại cho ra các định dạng ảnh khác nhau. Trong năm 1983, ACR và NEMA thành lập một ủy ban chung để phát triển phương pháp chuẩn này với mục đích:

- Tăng cường khả năng giao tiếp thông tin ảnh số của thiết bị y khoa bất chấp thiết bị đó là của nhà sản xuất nào.

- Giúp cho việc phát triển và mở rộng các hệ thống truyền tải và lưu trữ ảnh trở nên dễ dàng hơn, từ đó các hệ thống này sẽ là nơi giao tiếp với các hệ thống thông tin bệnh viện khác.

- Cho phép tạo ra thông tin cơ sở chẩn đoán, từ đó nhiều loại thiết bị chẩn bệnh sẽ sử dụng và tra cứu thông tin này.

ACR-NEMA công bố "ACR-NEMA Standards Publication" phiên bản 1.0 vào năm 1985. Và năm 1988, ủy ban này công bố tiếp "ACR-NEMA Standards Publication" phiên bản 2.0. Tài liệu "ACR-NEMA Standards Publication" đặc tả giao tiếp phần cứng, số lượng tối thiểu các lệnh phân mềm và các định dạng dữ liệu.

Chuẩn DICOM (Digital Imaging and Communications in Medicine) đưa ra nhiều cải tiến qua trọng so với 2 phiên bản của chuẩn ACR-NEMA trước:

Chuẩn DICOM này áp dụng được trong môi trường mạng vì chúng dùng giao thức mạng chuẩn là TCP/IP. Chuẩn ACR-NEMA chỉ có thể áp dụng cho mạng point-to-point.

Chuẩn DICOM áp dụng cho môi trường lưu trữ off-line, DICOM dùng các thiết bị lưu trữ chuẩn như CD-R, MOD và filesystem luận lý như ISO 9660 và FAT16. Chuẩn ACR-NEMA không đặc tả định dạng file, thiết bị lưu trữ vật lý hay filesystem luận lý.

Chuẩn DICOM đặc tả các thiết bị y khoa cần tuân theo chuẩn DICOM sẽ phải đáp ứng lệnh và dữ liệu như thế nào. Chuẩn ACR-NEMA bị giới hạn về truyền tải dữ liệu, DICOM dùng khái niệm Service Classes để mô tả ngữ nghĩa lệnh và dữ liệu đi kèm.

DICOM có kèm đặc tả về yêu cầu, quy tắc cho các nhà sản xuất thiết bị y khoa sản xuất sản phẩm tuân theo chuẩn DICOM. Chuẩn ACR-NEMA đặc tả rất ít về điều này.

Hướng phát triển hiện thời: chuẩn DICOM luôn phát triển và do Procedures of the DICOM Standards Committee quản lý. Đề nghị nâng cấp trong tương lai của các thành viên trong ủy ban DICOM dựa trên thông tin từ những người đã dung qua chuẩn DICOM. Các ý kiến được xem xét để đưa vào phiên bản tiếp theo của DICOM và các thay đổi của DICOM phải đảm bảo tương thích tốt với phiên bản trước.

1.1.2. Phạm vi và lĩnh vực ứng dụng của DICOM

Chuẩn DICOM gắn liền với thông tin y tế. Với lĩnh vực này, nó định ra sự trao đổi thông tin số giữa các thiết bị tạo ảnh và hệ thống mạng thông tin. Do các thiết bị tạo ảnh có thể hoạt động tương tác với các thiết bị y tế khác, phạm vi của chuẩn cần thiết phải chồng lên các khu vực khác trong thông tin y tế.

Chuẩn tăng cường khả năng hoạt động tương tác của các thiết bị tạo ảnh y tế bằng cách định ra:

- Với việc truyền thông tin qua mạng, chuẩn đưa ra một bộ giao thức được tuân theo bởi các thiết bị thích nghi chuẩn.

- Cú pháp và ngữ nghĩa của lệnh và các thông tin liên quan được trao đổi sử dụng các giao thức này.

- Với việc truyền tin bằng phương tiện trung gian, chuẩn đưa ra một bộ các dịch vụ lưu trữ trung gian, cũng như định dạng file và cấu trúc thư mục y tế, tạo điều kiện cho việc truy nhập thông tin lưu trữ trên phương tiện trung gian. Thông tin được sử dụng trong ứng dụng tuân theo chuẩn.

1.1.3. Cấu trúc của chuẩn ảnh DIOCM

**** / Các thành phần định dạng của ảnh DICOM***

Cấu trúc của DIOCM gồm các thành phần sau:

- Thích nghi: Định nghĩa các nguyên tắc thực thi chuẩn gồm các yêu cầu thích nghi và báo cáo thích nghi CS (Conformation Statement)

- Định nghĩa đối tượng thông tin IOD (Information Object Definition)

- Định nghĩa lớp dịch vụ SC (Service Classes)

- Ngữ nghĩa và cấu trúc dữ liệu

- Từ điển dữ liệu

- Trao đổi bản tin

- Hỗ trợ truyền thông mạng cho việc trao đổi bản tin

- Định dạng file và lưu trữ trung gian

- Sơ lược ứng dụng lưu trữ trung gian

- Chức năng lưu trữ và định dạng trung gian cho trao đổi dữ liệu

- Chức năng hiển thị chuẩn mức xám

- Sơ lược an toàn

- Nguồn ánh xạ nội dung.

	APPLICATION	File Transire, Email, HTTP
Upper Layers	PRESENTATION	Data Formatting, Compression,
	SESSION	synchronization, Communication,
	TRANSPORT	End to End communication
Lower Layers	NETWORK	Internetworking
	DATA LINK	LLC, MAC Ethernet, FDDI,...
	PHYSICAL	Fiber, Coax, UTP

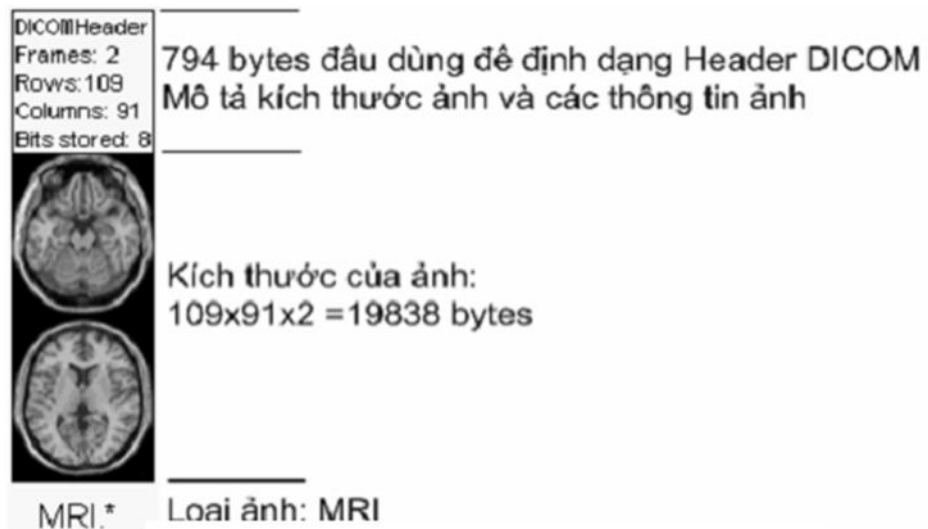
Bảng 1. DICOM và mô hình tham chiếu OSI

****/ Định dạng file DICOM***

Gồm 2 thành phần là header và dữ liệu ảnh:

+) Header

- Tên và ID của bệnh nhân
- Loại ảnh y khoa (CT, MR, Audio Recording,...)
- Kích thước ảnh, máy sinh ảnh,...



Hình 1.1. Định dạng file DICOM

Hình 1.1 chỉ ra rằng: 794 bytes đầu dùng để định dạng Header DICOM, mô tả kích thước ảnh và các thông tin ảnh. Để biết được kích thước ảnh ta dựa vào thông tin của Frames, Rows và Columns trong phần Header. Hình 1.2 là ví dụ về một ảnh MRI với số Frames, Rows, Columns tương ứng được chụp: $109 \times 91 \times 2 = 19838$ bytes. Như vậy ta sẽ tính được kích thước của ảnh


```

0002,0000,File Meta Elements Group Len: 132
0002,0001,File Meta Info Version: 256
0002,0010,Transfer Syntax UID: 1.2.840.10008.1.2.1.
0008,0000,Identifying Group Length: 152
0008,0060,Modality: MR
0008,0070,Manufacturer: MRlcro
0018,0000,Acquisition Group Length: 28
0018,0050,Slice Thickness: 2.00
0018,1020,Software Version: 46\64\37
0028,0000,Image Presentation Group Length: 148
0028,0002,Samples Per Pixel: 1
0028,0004,Photometric Interpretation: MONOCHROME2.
0028,0008,Number of Frames: 2
0028,0010,Rows: 109
0028,0011,Columns: 91
0028,0030,Pixel Spacing: 2.00\2.00
0028,0100,Bits Allocated: 8
0028,0101,Bits Stored: 8
0028,0102,High Bit: 7
0028,0103,Pixel Representation: 0
0028,1052,Rescale Intercept: 0.00
0028,1053,Rescale Slope: 0.00392157
7FE0,0000,Pixel Data Group Length: 19850
7FE0,0010,Pixel Data: 19838

```

Hình 1.2. Thông tin đính kèm file DICOM

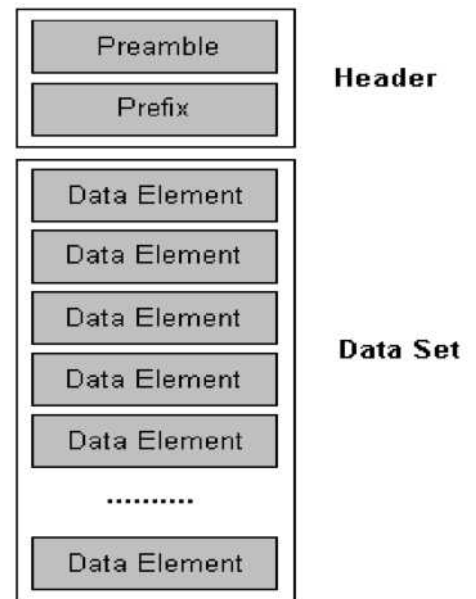
+) Dữ liệu ảnh

- Ảnh nén (bitmap) hoặc ảnh chưa nén từ (jpeg, gif, ...).
- Định nghĩa đối tượng thông tin IOD (Information Object Definition).
- Định nghĩa lớp dịch vụ SC (Service Classes).
- Ngữ nghĩa và cấu trúc dữ liệu.
- Từ điển dữ liệu.
- Trao đổi bản tin.
- Hỗ trợ truyền thông mạng cho việc trao đổi bản tin.
- Khuôn dạng file và lưu trữ trung gian.
- Sơ lược ứng dụng lưu trữ trung gian.
- Chức năng lưu trữ và khuôn dạng trung gian cho trao đổi dữ liệu.
- Chức năng hiển thị chuẩn mức xám.
- Sơ lược an toàn.
- Nguồn ảnh xạ nội dung

****/ Khuôn dạng file DICOM***

Thông tin đầu file (Header): Bao gồm các định danh bộ dữ liệu được đưa vào file. Nó bắt đầu bởi 128 byte file Preamble (tất cả được đưa về 00H), sau đó 4 byte kí tự “DICM”, tiếp theo là các thành phần dữ liệu đầu file. Các thành phần dữ liệu đầu file này là bắt buộc đối với mọi file DICOM. Các thành phần dữ liệu này có nhãn dạng (0002, xxxx), được mã hóa theo cú pháp chuyển đổi VR ẩn và Little Endian.

Bộ dữ liệu: Mỗi file chỉ chứa một bộ dữ liệu thể hiện một SOP cụ thể và duy nhất liên quan đến một lớp SOP đơn và IOD tương ứng. Một file có thể chứa nhiều hình ảnh khi các IOD được xác định mang nhiều khung. Cú pháp chuyển đổi được sử dụng để mã hóa bộ dữ liệu được xác định duy nhất thông qua UID cú pháp chuyển đổi trong thông tin đầu file DICOM.



Thông tin quản lý file: Khuôn dạng file DICOM không bao gồm thông tin quản lý file để tránh sự trùng lặp với chức năng liên quan ở lớp khuôn dạng trung gian. Nếu cần thiết với một sơ lược ứng dụng DICOM cho trước, các thông tin sau sẽ được đưa ra bởi một lớp khuôn dạng trung gian :

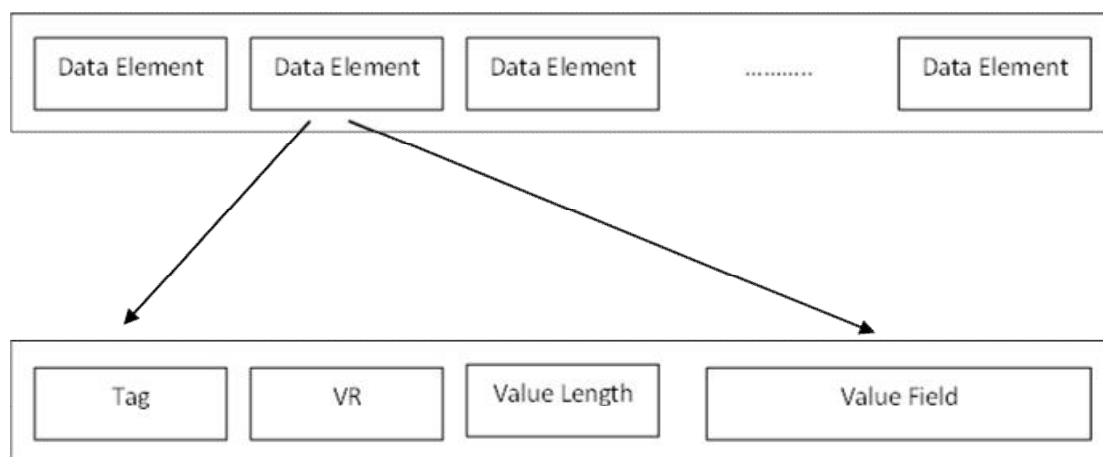
- Định danh sở hữu nội dung file.
- Thông tin truy cập (ngày giờ tạo).
- Điều khiển truy cập file ứng dụng.
- Điều khiển truy cập phương tiện trung gian vật lý (bảo vệ ghi ...)

Khuôn dạng file DICOM an toàn: Một file DICOM an toàn là một file DICOM được mã hóa với một cú pháp bản tin mật mã được định nghĩa trong

RFC2630. Phụ thuộc vào thuật toán mật mã sử dụng, một file DICOM an toàn có thể có các thuộc tính an toàn sau :

- Bảo mật dữ liệu.
- Xác nhận nguồn gốc dữ liệu.
- Tính toàn vẹn dữ liệu

Cấu trúc căn bản của file DICOM là Data Set



Hình 1.3. Cấu tạo Data Set

Một File sẽ chứa một Data Set biểu diễn cho một thực thể SOP liên quan tới lớp SOP (tương ứng với IOD). Tập dữ liệu bao gồm một tập hợp các phần tử dữ liệu (Data Element -DE). DE chứa giá trị thuộc tính được mã hoá theo cú pháp chuyển đổi (TrS-Transfer Syntax). Dựa vào TrS ta có thể xác định xem dữ liệu trong file được mã hoá theo kiểu gì.

Phần tử dữ liệu - Data Element

Một phần tử dữ liệu được xác định duy nhất bởi một thẻ. Nó có thể là một trong ba cấu trúc:

Hai trong số đó có trường VR (cho biết giá trị của *Data Element* được biểu diễn ở dạng dữ liệu nào, độ dài bao nhiêu, cố định hay thay đổi được) nhưng khác về cách biểu diễn độ dài, còn một cấu trúc thì không có trường

VR. Cả ba cấu trúc đều có trường *Data Element Tag*, *value Length*, giá trị của *Data Element*.

Thẻ Data Element Tag: bao gồm hai số nguyên không dấu 16 bit biểu diễn Group Number và Element Number. Mỗi loại thông tin được đặc trưng bởi thẻ này.

Giá trị biểu diễn – VR (Value Representation): một xâu hai ký tự biểu diễn tương ứng theo thẻ. VR sẽ được mã hoá bằng cách dùng tập ký tự mặc định của DICOM .

Chiều dài giá trị -Value Length: có hai trường hợp

Số nguyên không dấu 16 hoặc 32 bit tùy thuộc vào VR được biểu diễn theo kiểu tường minh hay không tường minh, mô tả số byte độ dài của trường Value.

Một trường 32 bit được đặt là FFFFFFFFH, chỉ một chiều dài không xác định. Giá trị này có thể được sử dụng khi Data Element có trường VR là SQ(Sequence of Item), UN(Unknown). Nó cũng có thể được sử dụng với VR là OB, OW phụ thuộc vào Transfer Syntax.

Trường giá trị - Value Field: chứa giá trị của *Data Element*.

****/ Các khái niệm trong DICOM***

Khái niệm	Ý nghĩa
Data Set	Là tập hợp nhiều Data Element trong một file DICOM.
Data Element	Là một đơn vị thông tin trong DICOM file. Data Element chứa một thông tin đầy đủ. Các field trong Data Element có nhiệm vụ đặc tả đầy đủ một thông tin, đặc tả bao gồm: ý nghĩa, giá trị, chiều dài của tin và định dạng dữ liệu của tin.
Tag	Là 2 số nguyên không dấu, mỗi số 16 bit. Cặp số nguyên này xác định ý nghĩa của Data Element như tên bệnh nhân, chiều cao của ảnh, số bit màu,

	<p>... Một số xác định Group Number và số kia xác định Element Number.</p> <p>Giá trị của Group Number và Element Number cho biết Data Element nói lên thông tin nào. Các thông tin (Data Element) cùng liên quan đến một nhóm ngữ nghĩa sẽ có chung số Group Number.</p>
VR(Value Representation)	<p>Đây là field tùy chọn, tùy vào giá trị của Transfer Syntax mà VR có mặt trong Data Element hay không.</p> <p>Giá trị của VR cho biết kiểu dữ liệu và định dạng giá trị của Data Element.</p>
VM (Value Multiplicity)	<p>Cho biết số lượng Value của Value Field nếu Value Field có nhiều giá trị.</p> <p>Nếu số lượng Value không xác định, VM sẽ có dạng “a-b” với a số giá trị Value nhỏ nhất và b là số Value lớn nhất có thể có của Data Element.</p> <p>VD: VM = “6-10” : Value Field có ít nhất là 6 giá trị và nhiều nhất là 10 giá trị.</p> <p>Data Element với Value Field có nhiều giá trị sẽ</p> <p>Với chuỗi kí tự, dùng kí tự 5Ch (‘\’) làm kí tự phân cách.</p> <p>Với giá trị nhị phân, không có kí tự phân cách.</p>
Value Length	<p>Là một số nguyên không dấu, có độ dài là 16 hay 32 bit. Giá trị của Value Length cho biết độ lớn (tính theo byte) của field Value Field (không phải là độ lớn của toàn bộ Data Element).</p> <p>Giá trị của Value Length là FFFFFFFFh (32 bit) hàm ý không xác định được chiều dài (Undefined Length).</p>
Value Field	Là nội dung thông tin (Data Element). Kiểu dữ

		liệu của field này do VR quy định và độ lớn (tính theo byte) nằm trong Value Length.
Transfer Syntax		Transfer Syntax là các quy ước định dạng dữ liệu. Giá trị của Transfer Syntax cho biết cách dữ liệu được định dạng và mã hóa trong DICOM đồng thời cũng cho biết VR sẽ có tồn tại trong Data Element hay không. Mặc định ban đầu, Transfer Syntax của file DICOM là Explicit VR Little Endian Transfer Syntax.
Information Object Definition (IOD)	Object	IOD đại diện cho một đối tượng chứa thông tin và đối tượng này có tồn tại trong thế giới thực. Thông tin của đối tượng IOD là thông tin của đối tượng trong thế giới thực. Có 2 loại IOD: Composite IOD: là IOD đại diện cho những phần khác nhau của các đối tượng khác nhau trong thế giới thực. Normalized IOD: là IOD cho duy nhất một đối tượng trong thế giới thực.
Lớp Service-Object Pair (SOP)		Lớp SOP được tạo ra khi ghép một IOD với DIMSE Service dành cho IOD đó. Có 2 loại lớp SOP: Lớp Normalized SOP: được tạo ra khi ghép Normalized IOD với các dịch vụ DIMSE- N. Lớp Composite SOP: được tạo ra khi ghép Composite IOD với các dịch vụ DIMSE- C.

Bảng 1. DICOM và mô hình tham chiếu OSI

Thứ tự của chuỗi byte: một giá trị sẽ được lưu thành một hay nhiều byte trong file. Có 2 quy ước quy định thứ tự xuất hiện của các byte của một giá trị nào đó trong file DICOM.

Little Endian	<p>Đối với số nhị phân gồm nhiều byte thì byte có trọng số thấp nhất (Least Significant Byte) sẽ nằm trước, những byte còn lại có trọng số tăng dần nằm tiếp sau đó.</p> <p>Đối với chuỗi kí tự, các kí tự sẽ nằm theo thứ tự xuất hiện trong chuỗi (từ trái sang phải).</p>
Big Endian	<p>Đối với số nhị phân gồm nhiều byte thì byte có trọng số lớn nhất (Most Significant Byte) sẽ nằm trước, những byte còn lại có trọng số giảm dần nằm tiếp sau đó.</p> <p>Đối với chuỗi kí tự, các kí tự sẽ nằm theo thứ tự xuất hiện</p>

Bảng 2. Quy ước trong file DICOM

1.2. Bài toán hiển thị ảnh

1.2.1. Đặt vấn đề

Việc hỗ trợ khám chữa bệnh liên tuyến giữa các bệnh viện là điều rất khó khăn, đặc biệt trong lĩnh vực giải phẫu, phẫu thuật chỉnh hình. Hiện đại hóa nền y học nước nhà để tạo ra những thay đổi mới mẻ, thúc đẩy sự phát triển và đem lại nhiều lợi ích thiết thực trong đời sống xã hội là vấn đề cần thiết và cấp bách.

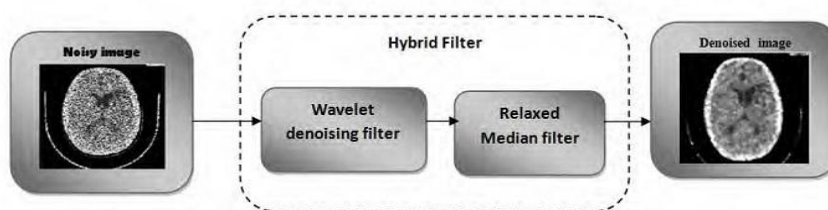
Thấy được tầm quan trọng đó, hầu hết các bệnh viện đã có khoa chẩn đoán hình ảnh riêng biệt và trong y tế cũng đã hình thành nên một chuẩn trao đổi dữ liệu hình ảnh DICOM (Digital Imaging and Communications in Medicine). Việc quản lý hệ thống thông tin hình ảnh cũng như việc khai thác nó trong các hệ thống thông tin bệnh viện trở thành một việc làm cần thiết và không thể thiếu, nhất là trong điều kiện cơ sở hạ tầng mạng phát triển và

chúng ta cũng quen dần với các khái niệm: Bệnh viện số, bệnh viện di động, chẩn đoán bệnh từ xa v.v...

Ảnh y tế DICOM được chụp từ các phương tiện kỹ thuật tiên tiến như: chụp cộng hưởng từ (Magnetic Resonance Imaging – MRI), chụp hình ảnh cắt lớp (Computed Tomography - CT), chụp X – quang hay siêu âm được ứng dụng rộng rãi trong cho chẩn đoán cận lâm sàng để phát hiện bệnh lý từ sọ não, đầu mặt cổ, tim, ngực, bụng, chậu, xương, mô mềm cho đến bệnh lý mạch máu não, cổ, mạch máu chi và các mạch máu tạng khác. Ảnh được chụp từ thiết bị chụp cắt lớp CT còn được dùng để hướng dẫn phẫu thuật xạ trị, theo dõi sau phẫu thuật

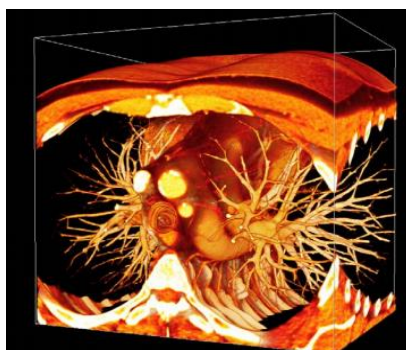
Kể từ khi ra đời vào năm 1993 cho đến nay, chuẩn DICOM đã được đầu tư nghiên cứu và được ứng dụng rất rộng rãi trong ngành y tế trên toàn thế giới. Có thể nói đây là chuẩn ảnh phổ biến nhất hiện nay. Khi nghiên cứu về DICOM vấn đề đang được quan tâm nhất hiện nay đó là kỹ thuật nhằm nâng cao chất lượng hình ảnh, cũng như các ứng dụng của DICOM trên nghiên cứu lâm sàng. Nhằm đưa chuẩn ảnh này đến một tầm cao mới của sự phát triển, không chỉ là ứng dụng trong chẩn đoán bệnh mà còn ứng dụng trong việc nghiên cứu các loại bệnh cũng như đào tạo y khoa.

Khi nghiên cứu về chuẩn DICOM, người ta quan tâm tới hai khía cạnh chủ yếu: thứ nhất là nâng cao chất lượng hiển thị ảnh DICOM, thứ hai là hiển thị các thông tin đi kèm một cách đầy đủ và khoa học giúp quá trình chẩn đoán được thuận lợi, chính xác và khách quan hơn. Liên quan đến kỹ thuật nhằm nâng cao chất lượng ảnh y tế, đã có nhiều công trình nghiên cứu của nhiều nhà khoa học trên thế giới được công bố. Trong nghiên cứu của mình năm 2011[3] Umamaheswari, J. và Radhamani, G. đã sử dụng bộ lọc dựa trên wavelet nhằm làm giảm nhiễu trên hình ảnh thu nhận được từ máy CT. Quá trình lọc nhiễu của nhóm tác giả được mô tả trong hình dưới đây.



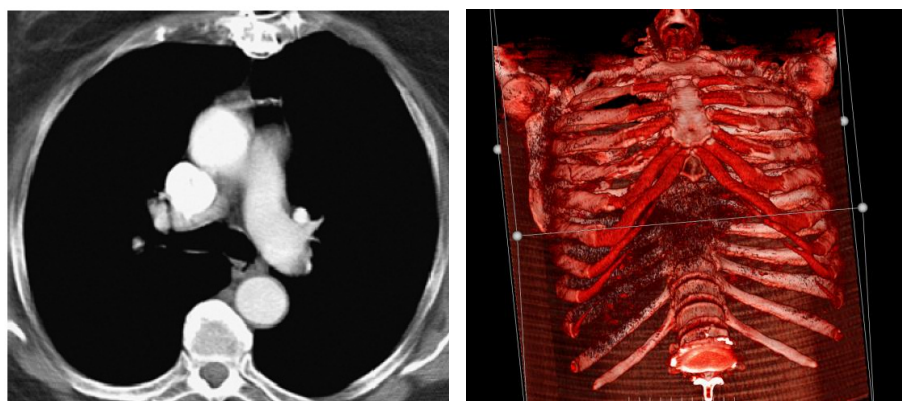
Hình 1.4. Lọc nhiễu ảnh y tế DICOM

Hình ảnh phân tích y tế đòi hỏi độ chính xác cao, các thuật toán dựng hình hiệu suất cao. Trong nghiên cứu của nhóm năm 2009 [4] Mikhail Smelyanskiy và các cộng sự đã đề xuất các kỹ thuật tính toán song song dựa trên GPU kết hợp với các kỹ thuật hiển thị hình ảnh 3D như render volume, nhằm làm tăng độ chính xác, chất lượng cũng như tốc độ hiển thị hình ảnh y tế.



Hình 1.5. Hiển thị 3D hình ảnh y tế bằng kỹ thuật Render Volume

Hiển thị ảnh DICOM bằng công nghệ hiển thị 2D, 3D



- a) Hình ảnh hiển thị sử dụng thông tin 2D b) Hình ảnh hiển thị sử dụng thông tin 3D

Hình 1.6. Hiển thị ảnh DICOM bằng công nghệ hiển thị 2D,3D

Tuy nhiên các kết quả đạt được mới chỉ dừng lại ở mức độ hiển thị 2D , chưa tích hợp các kỹ thuật xử lý ảnh liên quan đến chú giải các vùng ảnh và công nghệ hiển thị hình ảnh 3D. Trong khi dữ liệu ảnh DICOM bản chất là dữ liệu hình ảnh 3 chiều, trong khi đó lâu nay chúng ta vẫn sử dụng nó như hình ảnh 2 chiều rời rạc, chưa tận dụng được các thông tin từ dữ liệu hình ảnh này một cách tối đa. Với sự phát triển vượt bậc của các kỹ thuật cũng như công nghệ hiển thị, cho phép chúng ta có thể hiển thị các thông tin liên quan, giải thích cụ thể và chi tiết các vùng tổn thương. Nói cách khác, nghiên cứu nâng cao chất lượng hình ảnh trong y tế sử dụng công nghệ 3D ở Việt Nam hiện nay vẫn chưa được quan tâm và phát triển.

Chuyên đề này sẽ tập trung vào việc đề xuất một số giải pháp trong xây dựng lát cắt ảnh và kỹ thuật xây dựng mô hình 3D từ ảnh chụp cắt lớp nhằm phục vụ trong quá trình quản lý và chẩn đoán hình ảnh y tế.

1.2.2. Phương pháp nghiên cứu

Trên cơ sở thực trạng trong việc quản lý khám chữa bệnh tại bệnh viện, việc thực hiện xây dựng hệ thống phần mềm quản lý và khai thác dữ liệu chẩn đoán hình ảnh đồng thời xây dựng mô đun phần mềm kết nối vào hệ thống phần mềm quản lý thông tin bệnh viện là vấn đề hết sức quan trọng và cần thiết. Đó đó, khi nghiên cứu đề tài xây dựng hệ thống phần mềm quản lý và khai thác dữ liệu chẩn đoán hình ảnh tại bệnh viện trong đó bao gồm nghiên cứu xây dựng hệ cơ sở dữ liệu đa phương tiện dựa trên chuẩn ảnh y tế DICOM và phát triển hệ thống phần mềm quản lý và truyền thông dữ liệu hình ảnh DICOM đến các trạm xử lý khác trong cùng bệnh viện hoặc giữa các bệnh viện khác, đồng thời xây dựng các mô đun chức năng đáp ứng cần thiết cho việc quản lý và khai thác dữ liệu một cách hiệu quả và chính xác.

Do các vấn đề có liên quan là những vấn đề trên thế giới đang nghiên cứu và phát triển nên công việc đầu tiên cần phải tiến hành là thu thập và hệ thống hóa các tài liệu chuyên môn liên quan đến những kết quả gần đây nhất

về nhận dạng, xử lý ảnh và công nghệ đa phương tiện trên nền hệ thống nhúng. Các bước tiến hành sẽ tuân theo quy trình: khảo sát điều tra thực trạng, nghiên cứu, phân tích, thiết kế giải pháp, cài đặt và triển khai ứng dụng thử nghiệm hệ thống. Việc hợp tác với các tập thể và cá nhân trong và ngoài nước sẽ được đặc biệt quan tâm, tận dụng tối đa chất xám và công nghệ mới, nâng cao chất lượng và hiệu quả thực hiện đề tài.

Phương pháp nghiên cứu của đề tài được lựa chọn là lý thuyết kết hợp với thực nghiệm. Các vấn đề cần giải quyết liên quan đến các giải thuật và lý thuyết truyền thông mạng, xử lý ảnh và nhận dạng được thực hiện trên phần mềm máy tính với đầu vào là các thông tin thu nhận được từ thực tế. Quá trình được tiến hành dựa trên việc tìm hiểu tài liệu, cài đặt thử nghiệm sau đó đánh giá kết quả thực nghiệm và cải tiến nhằm nâng cao chất lượng.

CHƯƠNG 2: MỘT SỐ KỸ THUẬT HIỂN THỊ ẢNH DICOM

Nhiều hiệu ứng hình ảnh trong tự nhiên có hình dạng kiểu volumetric. Chất lỏng, mây, lửa, khói, sương mù, và bụi rất khó có thể mô hình hóa với các đối tượng hình học nguyên thủy. Do đó, sử dụng mô hình volumetric là phù hợp cho việc tạo hiệu ứng như vậy. Những mô hình này giả định rằng ánh sáng được phát ra, hấp thụ và phân tán của một số lượng lớn các hạt trong một vùng thể tích. Cho hai ví dụ.



Hình 2.1. Các hiệu ứng dạng Volumetric

Ngoài việc mô hình hóa và hiển thị các hiện tượng, Volume rendering là cần thiết cho các ứng dụng khoa học và kỹ thuật đòi hỏi phải hình dung của bộ dữ liệu ba chiều. Ví dụ như hình dung của dữ liệu thu được bởi các thiết bị hình ảnh y tế hoặc do tính toán mô phỏng động lực học. Người sử dụng các ứng dụng Volume rendering dựng hình tương tác dựa trên hiệu suất của máy gia tốc đồ họa hiện đại cho thăm dò dữ liệu hiệu quả và khám phá tương lai.

Chuyên đề này trình bày chi tiết kỹ thuật Volume rendering. Mặc dù nó có thể thực hiện các thuật toán Volume rendering phổ biến khác trên GPU, như ray casting (Roettger et al. 2003, Krüger và Westermann 2003).

2.1. Kỹ thuật nội suy trong xây dựng lát cắt ảnh

2.1.1. Kỹ thuật nội suy các điểm ảnh láng giềng gần nhất

**/ Giới thiệu*

Trong kỹ thuật nội suy nhằm tăng cường độ phân giải khi phóng to ảnh thì kỹ thuật ra đời đầu tiên (khoảng những năm 1940) và đơn giản nhất.

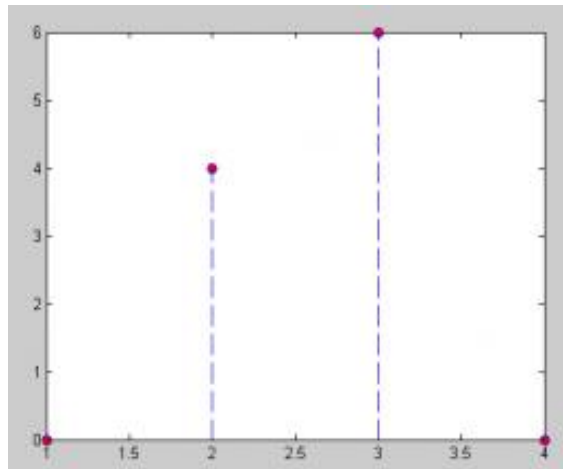
Tư tưởng của kỹ thuật này chỉ đơn giản là xác định điểm ảnh láng giềng gần nhất với điểm cần nội suy và lấy giá trị cường độ của điểm ảnh gần nhất đó mà không cần phải tính toán giá trị trung bình theo một tiêu chí hay quy tắc phức tạp nào khác. Điểm mấu chốt của kỹ thuật này là xác định điểm láng giềng gần nhất bằng khoảng cách Euclid hoặc khoảng cách Mahanttan với $k=1$ hoặc khoảng cách Minkowski với $k=2$.

****/ Phương pháp***

Kỹ thuật nội suy láng giềng gần nhất là cách đơn giản nhất để lấy xấp xỉ hàm sinc được coi là hàm đánh trọng số cho các điểm lân cận như sau [5]:

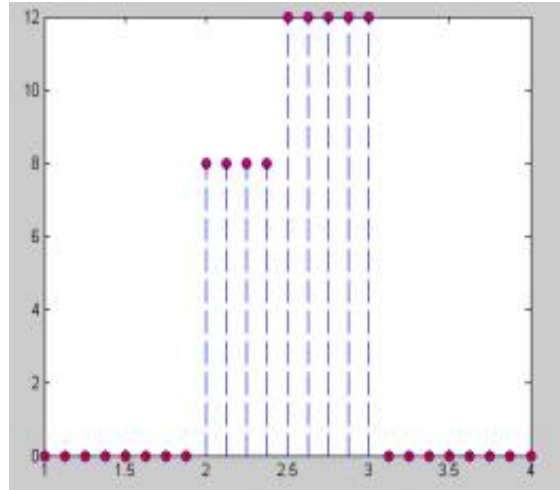
$$h(x) = \begin{cases} 1, & |x| < 1/2 \\ 0, & \text{trường hợp khác} \end{cases} \quad (1)$$

Giả sử đầu tiên áp dụng kỹ thuật này trong không gian 1 chiều như hình 2.2 dưới đây:



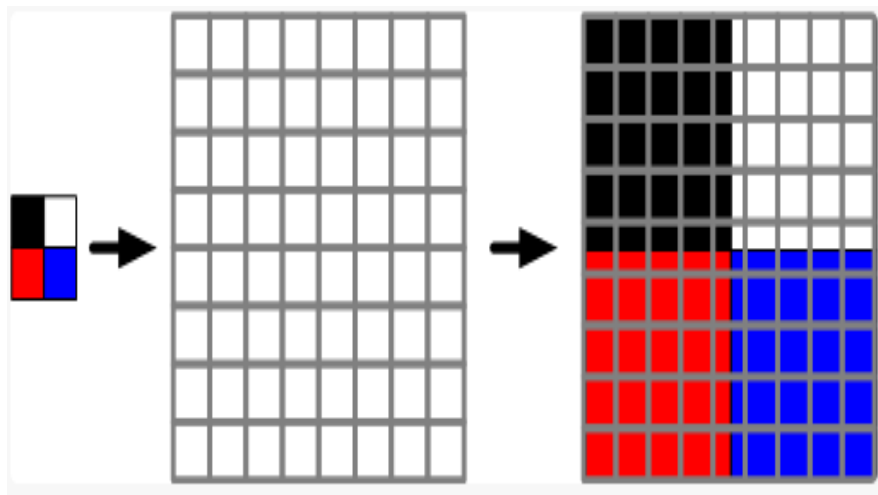
Hình 2.2. Ví dụ nội suy điểm trên không gian 1 chiều

Lúc này, cần chèn thêm các giá trị ở giữa $x_1 = 2$ và $x_2 = 3$, giá trị gần đúng của hàm $y=f(x)$, trong khoảng $f(x_1) = 4$ và $f(x_2) = 6$. Sử dụng nội suy láng giềng gần nhất được kết quả như hình dưới đây.

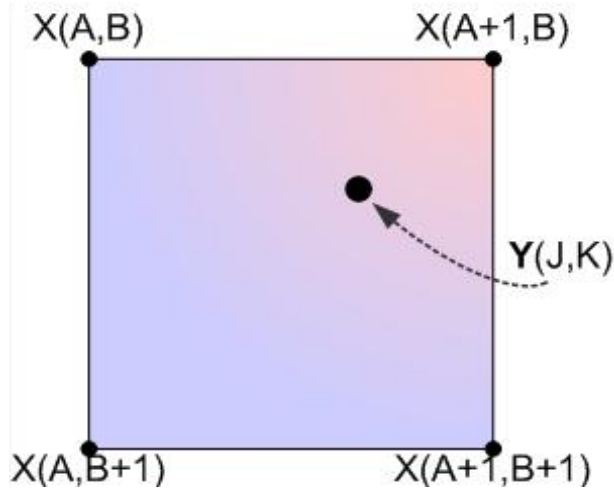


Hình 2.3. Kết quả nội suy láng giềng gần nhất trong không gian 1 chiều

Có thể thấy rằng với mỗi điểm x_i giữa hai điểm dữ liệu gốc ban đầu x_1 , x_2 , một giá trị $f(x_i)$ được tính dựa trên điểm gần nhất với điểm gốc trên trục Ox . Ví dụ tiếp theo được đưa ra với ảnh 2D kích cỡ 2×2 cần phóng to đạt được kích thước 9×9 và kỹ thuật nội suy láng giềng gần nhất được sử dụng để lấp đầy khoảng trống khi phóng to ảnh.



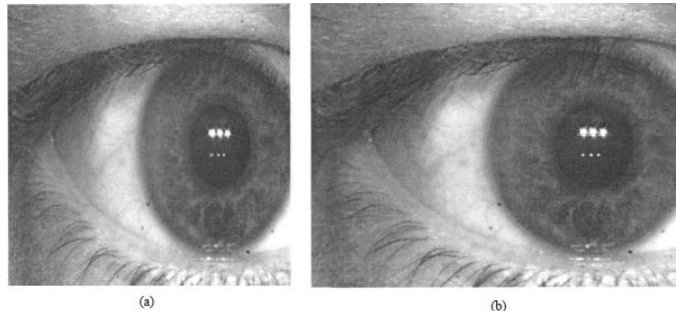
Hình 2.4. Phóng to ảnh kích thước 2×2 thành ảnh 9×9



Hình 2.5. Nội suy điểm $Y(J,K)$ dựa trên các điểm ảnh lân cận

***/ Nhận xét**

- +) Ưu điểm: Kỹ thuật nội suy láng giềng gần nhất là kỹ thuật có tư tưởng đơn giản, không cần tính toán nhiều do đó có tốc độ nhanh.
- +) Nhược điểm: Kỹ thuật này có khuynh hướng để lại nhiều hình khối trong ảnh được nội suy và xảy ra hiện tượng “răng cưa” khi phóng ảnh với kích thước lớn hơn nhiều lần ảnh gốc.
- +) Ứng dụng:



Hình 2.6. Ứng dụng nội suy láng giềng gần nhất trong thay đổi tỷ lệ ảnh CCD

Ví dụ trong hình 2.6 (a) cho hình ảnh kỹ thuật số mắt của con người. Vị trí của các phản xạ Purkinje của đồng tử (con người) được sử dụng để chẩn đoán hiện tượng lác. Hình ảnh được sử dụng trong chẩn đoán phải có độ sắc nét phù hợp. Tuy nhiên, để hiển thị được các phản xạ Purkinje yêu cầu phải phóng to theo tỷ lệ $4/3$ tức là điều chỉnh tỷ lệ theo trục x gấp 1.3 lần (như hình

2.6 (b) nhưng những điểm ảnh nội suy phải được lấy từ dữ liệu gốc ban đầu mà không được sửa đổi.

Kỹ thuật nội suy láng giềng gần nhất được sử dụng và cho kết quả chính xác nhất trong trường hợp cụ thể này.

2.1.2. Kỹ thuật nội suy tuyến tính

**/ Giới thiệu*

Kỹ thuật nội suy tuyến tính được phát triển cùng giai đoạn với nội suy láng giềng gần nhất. Kỹ thuật này được sử dụng trong cả hai hướng tiếp cận trong xử lý ảnh y tế đó là ứng dụng trong thay đổi kích thước ảnh và sinh lát cắt trung gian. Nội suy tuyến tính có nhiều phương thức thực thi phụ thuộc vào bậc của hàm nội suy. Trong luận văn này chỉ trình bày nội suy dựa trên hàm bậc hai hay còn gọi là nội suy song tuyến.

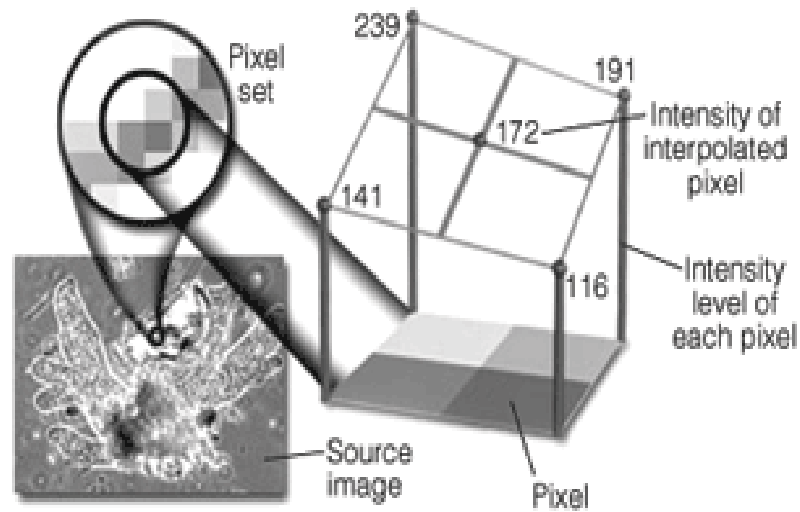
**/ Phương pháp*

+) Nội suy song tuyến trong thay đổi kích thước ảnh

Giả sử ảnh đầu vào v có kích thước $m \times n$. Mục tiêu của nội suy là biến đổi ảnh v trở thành ảnh có kích thước mới $m' \times n'$ bằng cách tính các giá trị điểm nội suy $u(x,y)$. Tư tưởng của kỹ thuật nội suy song tuyến làm tăng độ phân giải của ảnh đó là giá trị cường độ của điểm ảnh nội suy được tính bằng trung bình cường độ của 2 điểm láng giềng lân cận với nó. Các điểm láng giềng được xác định bằng phép đo khoảng cách cụ thể (phép đo Euclid hoặc Mahanttan).

Đối với nội suy song tuyến, các giá trị của các láng giềng trực tiếp được tính bằng khoảng cách của chúng tới điểm nội suy. Do đó, hàm sin được tính xấp xỉ như sau:

$$h(x) = \begin{cases} 1 - |x|, & 0 \leq |x| < 1 \\ 0, & \text{trường hợp khác} \end{cases} \quad (3)$$



Hình 2.7. Mô tả kỹ thuật nội suy tuyến trong ứng dụng làm tăng độ phân giải của ảnh

+) Nội suy tuyến tính sinh lát cắt trung gian

Điểm ảnh trên lát cắt nội suy bằng kỹ thuật nội suy tuyến tính được ước tính bằng trung bình giá trị cường độ của điểm ảnh trên lát cắt tham chiếu và lát cắt mục tiêu. Giả sử kích thước của mỗi lát cắt là $m \times n$.

Trường hợp lát cắt nội suy nằm giữa hai lát cắt đầu vào, cường độ điểm ảnh của lát cắt nội suy được ước tính theo công thức:

$$I_{int}(i, j) = \frac{I_{ref}(i, j) + I_{tar}(i, j)}{2} \quad (4)$$

Trong đó: I_{int} , I_{ref} , I_{tar} lần lượt là cường độ của lát cắt nội suy, lát cắt tham chiếu và lát cắt mục tiêu; $1 \leq i \leq n$, $1 \leq j \leq m$.

***/ Nhận xét**

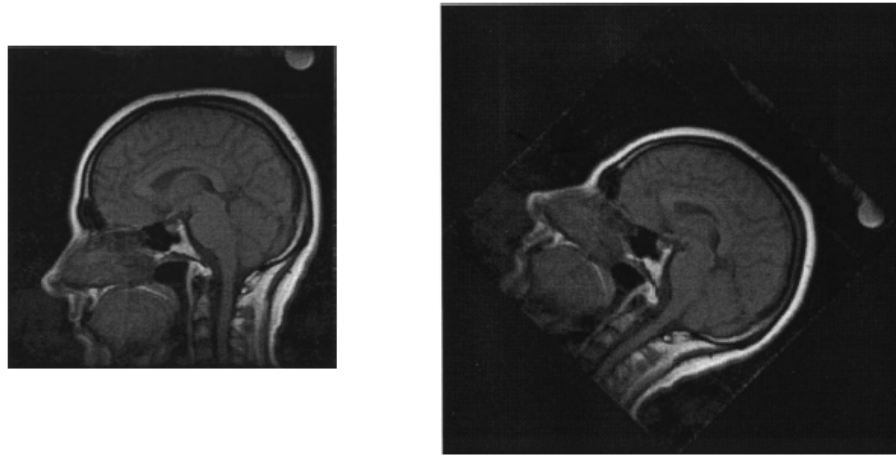
+) Ưu điểm: kỹ thuật nội suy tuyến tính được coi là kỹ thuật đơn giản, thời gian thực thi nhanh, chi phí thấp.

+) Nhược điểm:

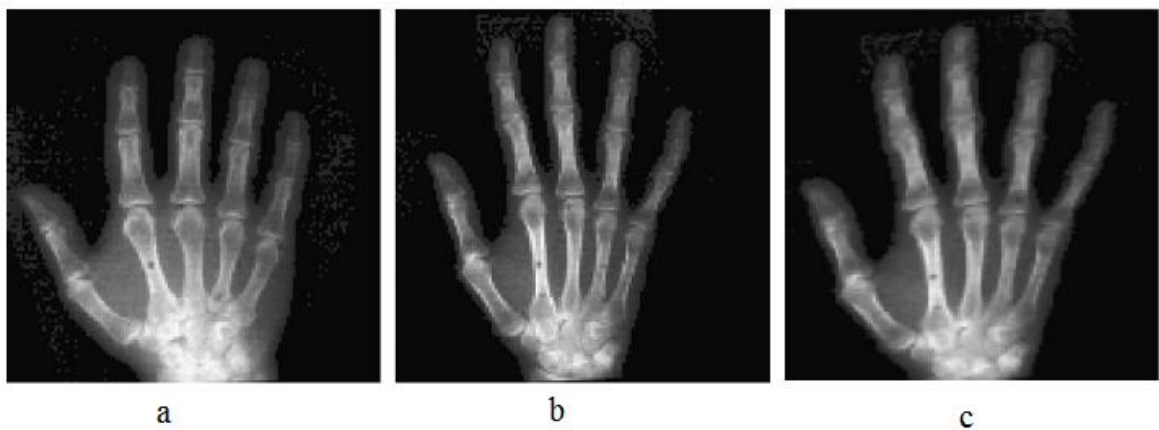
- Kỹ thuật nội suy tuyến tính áp dụng cho việc thay đổi kích thước ảnh vẫn còn hiện tượng “răng cưa”, ảnh nội suy chưa có được sắc nét như ảnh ban đầu.

- Trong nội suy lát cắt trung gian, kỹ thuật nội suy tuyến tính sinh ra hình ảnh còn vết mờ trên đường biên ảnh.

****/ Ứng dụng***



Hình 2.8. Ứng dụng nội suy tuyến tính trong xoay ảnh MRI



Hình 2.9. Ứng dụng nội suy tuyến tính sinh lát cắt trung gian

(a) lát cắt tham chiếu, (b) lát cắt mục tiêu, (c) lát cắt nội suy

2.1.3. Kỹ thuật nội suy song khối

****/ Giới thiệu***

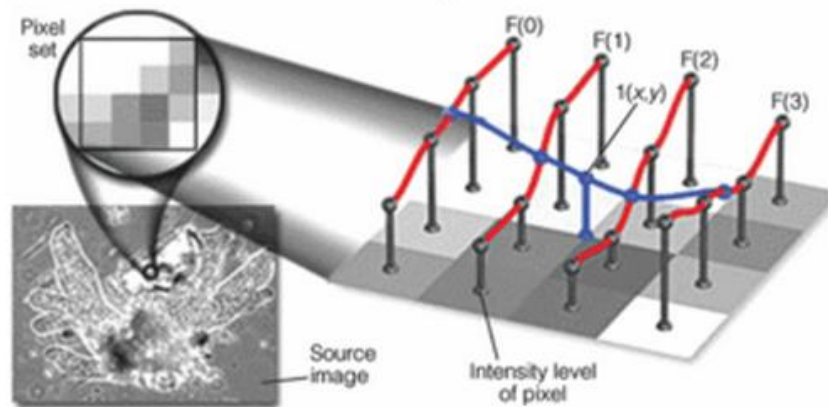
Kỹ thuật nội suy song khối là kỹ thuật cho kết quả khá tốt và được sử dụng nhiều trong các phần mềm xử lý ảnh số nói chung và ảnh y tế nói riêng. Đây cũng là kỹ thuật được áp dụng cho cả hai hướng tiếp cận nội suy trong việc thay đổi kích thước ảnh và nội suy lát cắt trung gian. Phần tiếp theo sẽ

trình bày phương pháp thực hiện để thay đổi kích thước ảnh bằng kỹ thuật nội suy song khối.

****/ Phương pháp***

Tư tưởng của kỹ thuật này để tính toán điểm ảnh mới được tạo ra sẽ có giá trị trung bình của 16 điểm ảnh gốc gần nhất (4×4).

Trong toán học, phép nội suy song khối (Bicubic Interpolation) là mở rộng của nội suy khối cho nội suy điểm dữ liệu trên lưới hai chiều. Nội suy song khối có thể sử dụng đa thức Lagrange, thuật toán cubic splines (đường bậc ba) hoặc thuật toán cubic convolution (khối cuộn).



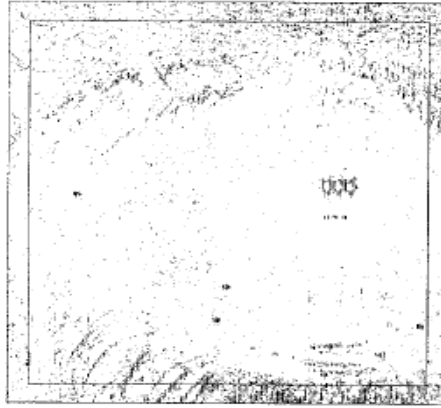
Hình 2.10. Mô tả kỹ thuật nội suy song khối

****/ Nhận xét***

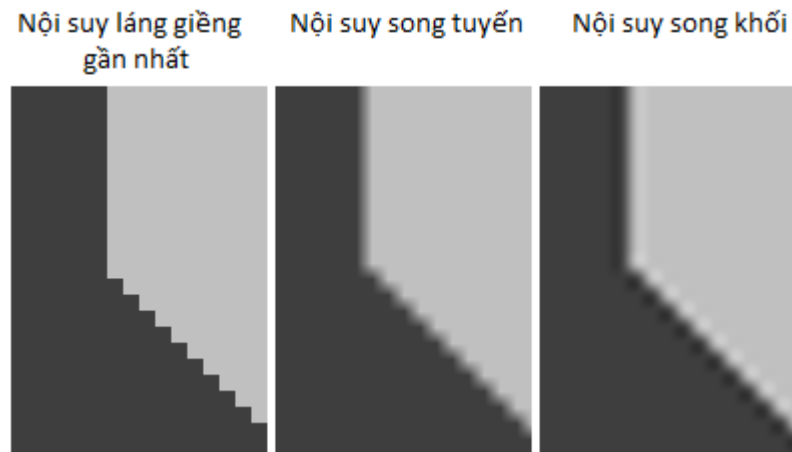
- +) Ưu điểm: kỹ thuật nội suy song khối cho ảnh nội suy khá sắc nét, hạn chế hiện tượng răng cưa do số điểm ảnh đưa vào nội suy nhiều hơn.
- +) Nhược điểm: chính vì mỗi điểm ảnh nội suy được tính toán từ 16 điểm ảnh gốc lân cận nên thời gian tính toán nhiều hơn so với 2 kỹ thuật đã trình bày.

****/ Ứng dụng***

Kỹ thuật nội suy song khối trong trường hợp thay đổi tỷ lệ kích thước của ảnh CCD được cho như trong hình 2.6 (a). Kết quả ảnh nội suy bằng kỹ thuật nội suy song khối được so sánh với kết quả nội suy láng giềng gần nhất (kỹ thuật này được xác định là chính xác nhất trong trường hợp cụ thể này). Ở đây các điểm ảnh có lỗi >1.0 được hiển thị màu đen trên hình.



Hình 2.11. Kết quả nội suy song khối trong trường hợp biến đổi tỷ lệ ảnh CCD */Đánh giá 3 kỹ thuật nội suy láng giềng gần nhất, nội suy song tuyến và nội suy song khối.



Hình 2.12. So sánh 3 kỹ thuật nội suy

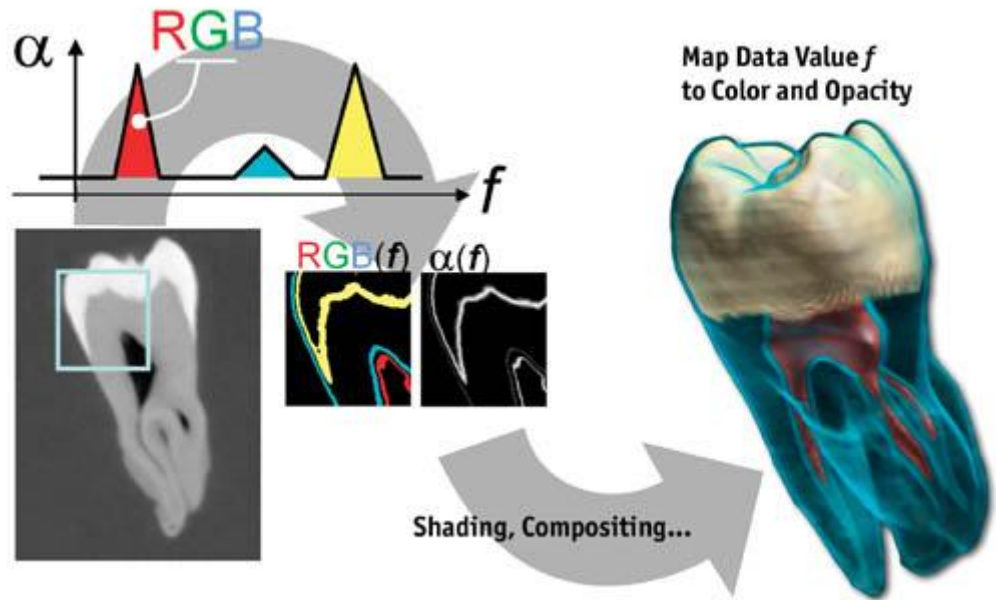
Hình trên cho thấy bề mặt đã nội suy bằng kỹ thuật song khối trông nhìn hơn bề mặt tương ứng đạt được của nội suy song tuyến tính (bilinear interpolation) hoặc nội suy các điểm gần nhất (Nearest Neighbor Interpolation).

2.2. Kỹ thuật Volume Rendering

2.2.1. Volume Rendering

Phương pháp Volume rendering trực tiếp tạo ra hình ảnh của một dữ liệu dạng khối 3D mà không cần khai báo một cách rõ ràng đặt bề mặt hình học từ dữ liệu. Những kỹ thuật này sử dụng một mô hình quang để ánh xạ giá trị dữ liệu và tính chất quang học, chẳng hạn như màu sắc và độ mờ đục. Trong khi

hiển thị, tính chất quang học được tích lũy theo từng tia để tạo thành một hình ảnh của dữ liệu.



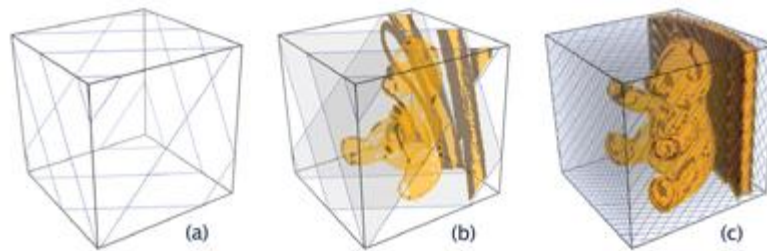
Hình 2.13. Quá trình Volume Rendering

Mặc dù tập dữ liệu được hiểu như là một hàm liên tục trong không gian, trên thực tế nó được đại diện bởi một mảng 3D đồng nhất của mẫu. Trong bộ nhớ đồ họa, khối dữ liệu được lưu giữ như một ngăn xếp của các lát cắt 2D hoặc như là một đối tượng 3D duy nhất. *Voxel* cho Volume rendering biểu thị một "khối phần tử," tương tự như *pixel* cho ảnh 2D. Mỗi voxel tương ứng với một vị trí trong không gian dữ liệu và có một hoặc nhiều giá trị dữ liệu liên kết với nó. Giá trị tại các vị trí trung gian thu được bằng cách nội suy dữ liệu láng giềng. Quá trình này được gọi là *khôi phục lại* và đóng một vai trò quan trọng trong các ứng dụng Volume rendering.

Về bản chất, vai trò của mô hình quang học là để mô tả cách thức các hạt trong khối tương tác với ánh sáng. Ví dụ, các mô hình phổ biến nhất được sử dụng cho rằng khối bao gồm các hạt mà đồng thời phát ra và hấp thụ ánh sáng. Mô hình phức tạp hơn kết hợp chiếu sáng và bóng địa phương. Thông số quang học được quy định bởi các giá trị dữ liệu trực tiếp, hoặc chúng được tính toán từ việc áp dụng một hoặc nhiều *chức năng truyền* dữ liệu. Mục đích

của chức năng truyền trong các ứng dụng trực quan là để nhấn mạnh hoặc *phân lớp* các tính năng quan tâm trong dữ liệu. Thông thường, chức năng truyền được thực hiện bởi *các bảng tra cứu kết cấu*, mặc dù chức năng đơn giản cũng có thể được tính trong đánh bóng mảnh. Ví dụ, hình 2.17 minh họa việc sử dụng một chức năng truyền để làm rõ ranh giới vật chất từ một ảnh chụp cắt lớp CT của một chiếc răng.

Hình ảnh được tạo ra bằng cách lấy mẫu khối lượng cùng tất cả các tia nhìn và tích lũy các tính chất quang học. Đối với mô hình phát xạ hấp thụ, màu sắc và độ mờ đục tích lũy được tính theo phương trình 1, trong đó C_i và A_i là những màu sắc và độ mờ đục của các hàm truyền cho các giá trị dữ liệu ở mẫu i .



Hình 2.14. Khối mẫu và kết hợp

Phương trình (1) rời rạc Volume Rendering

$$C = \sum_{i=1}^n C_i \prod_{j=1}^{i-1} (1 - A_j)$$

$$A = 1 - \prod_{j=1}^n (1 - A_j)$$

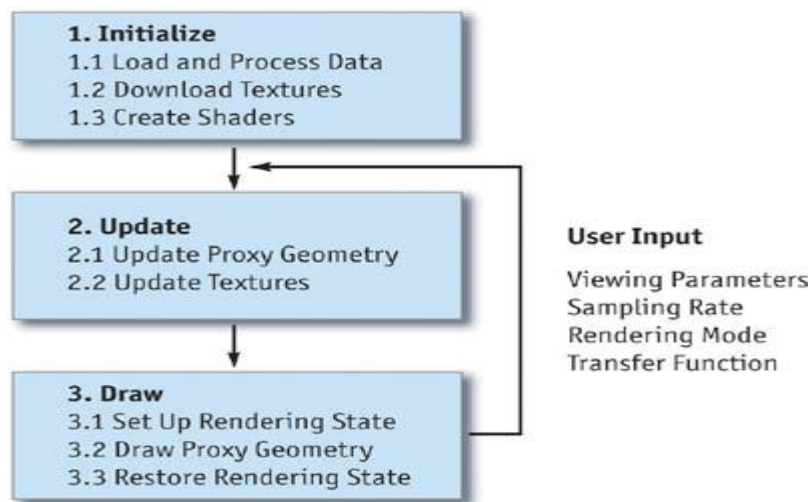
Độ trong suốt A_i xấp xỉ với sự hấp thụ, và độ mờ đục màu C_i xấp xỉ phát và hấp thụ cùng tia giữa các mẫu i và $i + 1$. Đối với các thành phần màu sắc, các tích trong các tổng đại diện cho số lượng mà ánh sáng phát ra tại mẫu i đang suy yếu trước khi đến mắt. Công thức này được đánh giá có hiệu quả bằng cách phân loại các mẫu vật dọc theo tia nhìn và tính toán tích lũy màu C và độ mờ đục A lặp đi lặp lại. Mô tả cách các bước *sắp xếp bố cục* có

thể được thực hiện thông qua trộn kênh trong suốt. Bởi vì phương trình 1 là một xấp xỉ số cho mô hình quang liên tục, các tỷ lệ lấy mẫu s , đó là tỷ lệ nghịch với khoảng cách giữa các mẫu l , rất ảnh hưởng đến độ chính xác của xấp xỉ và chất lượng của việc hiển thị.

Kỹ thuật Texture-based volume rendering dựa trên thực hiện các bước lấy mẫu và hợp lại bằng cách làm cho một tập hợp các hình học 2D nguyên thủy bên trong khối, như thể hiện trong. Mỗi nguyên thủy được gián tọa độ texture cho lấy mẫu các texture. Kỹ thuật chiếu sáng có thể thay đổi màu sắc kết quả trước khi nó được gửi đến các giai đoạn hợp lại.

2.2.2. Texture-Based Volume Rendering

Texture-based volume rendering dựa trên thuật toán dựng hình có thể được chia thành ba giai đoạn: (1) Khởi tạo, (2) Cập nhật, và (3) Vẽ. Giai đoạn khởi tạo thường được thực hiện chỉ một lần. Update và vẽ là giai đoạn được thực hiện bất cứ khi nào các ứng dụng nhận đầu vào cho người dùng ví dụ, khi xem hoặc làm thay đổi các thông số.



Hình 2.15. Các bước của một quá trình Texture-Based Volume Rendering

Khi khởi động ứng dụng, khối dữ liệu được nạp vào bộ nhớ CPU. Trong một số trường hợp, các bộ dữ liệu cũng cần phải được xử lý trước khi đóng gói và tải chúng vào bộ nhớ texture. Ví dụ, người ta có thể lựa chọn để tính

toán độ dốc hoặc xuống lấy mẫu dữ liệu ở giai đoạn này. Một số các hoạt động xử lý dữ liệu cũng có thể được thực hiện bên ngoài các ứng dụng.

Sau khi khởi tạo và mỗi lần thay đổi thông số nhìn, thay cho hình học sẽ được tính toán và lưu trữ trong mảng vertex. Khi bộ dữ liệu được lưu trữ như là một đối tượng 3D, hình học được thay bởi một tập hợp các đa giác, cắt qua khối vuông góc với hướng nhìn. Lát cắt đa giác được tính bằng cách lấy giao giữa những mặt phẳng cắt với các cạnh của hộp bao khối và sau đó phân loại các đỉnh kết quả trong một chiều kim đồng hồ hoặc ngược chiều kim đồng hồ xung quanh tâm. Đối với mỗi đỉnh, hình ảnh 3D tương ứng tọa độ được tính toán trên CPU, trong một quá trình vertex, hoặc thông qua tự động tạo ra texture-coordinate.

Khi một tập hợp dữ liệu được lưu trữ như là một tập hợp các lát 2D, các đa giác thay thế chỉ đơn giản là hình chữ nhật phù hợp với những lát. Mặc dù là nhanh hơn, phương pháp này có một số nhược điểm. Đầu tiên, nó đòi hỏi nhiều bộ nhớ hơn ba lần, vì lát cắt dữ liệu cần phải được nhân rộng dọc theo mỗi hướng chính. Bản sao dữ liệu có thể tránh được với một số trên không hiệu quả bằng cách xây dựng lại lát cắt. Thứ hai, tỷ lệ lấy mẫu phụ thuộc vào độ phân giải của khối. Vấn đề này có thể được giải quyết bằng cách thêm lát trung gian và thực hiện nội suy tuyến tính với một đánh bóng. Thứ ba, thay đổi khoảng cách lấy mẫu với các điểm nhìn, dẫn đến các biến cường độ khi di chuyển camera và hiện vật hình ảnh đầy ấn tượng khi chuyển đổi từ một tập hợp các lát khác.

Trong giai đoạn cập nhật, các texture được làm mới nếu chế độ hiển thị hoặc các thông số chức năng chuyển đổi. Ngoài ra, hiệu chỉnh độ mờ đục của các hàm truyền texture được thực hiện nếu tỷ lệ lấy mẫu đã thay đổi.

Trước khi các lát cắt của các đa giác được vẽ và lấy ra trong thứ tự sắp xếp, trạng thái hiển thị được thiết lập phù hợp. Bước này thường bao gồm việc vô hiệu hóa ánh sáng và suy hao, và thiết lập trộn kênh trong suốt. Để

trộn trong hình học không rõ ràng, kiểm tra độ sâu đã được kích hoạt, và ghi vào bộ đệm chiều sâu phải được vô hiệu hóa. Khối lượng và chức năng chuyển giao các kết cấu phải được ràng buộc với các đơn vị kết cấu, mà các đánh bóng sử dụng cho đầu vào. Tại thời điểm này, các thông số đầu vào đánh bóng được quy định, và mảng đỉnh được thiết lập để hiển thị. Cuối cùng, sau khi lát cắt được lấy ra trong thứ tự sắp xếp, trạng thái hiển thị được phục hồi, vì vậy mà các thuật toán không ảnh hưởng đến việc hiển thị các đối tượng khác trong cảnh.

Thuật toán 1: các bước của ứng dụng hiển thị hình ảnh sử dụng khối

- 1) Tạo và tải về các dữ liệu thiết lập như là một cấu trúc trong suốt 3D.
 - 2) Tải các chương trình con.
 - 3) Nạp góc nhìn và ma trận phép chiếu.
 - 4) Kích hoạt tính năng trộn kênh trong suốt bằng cách sử dụng một đoạn mã cho nguồn và một đoạn mã cho đích.
 - 5) Vô hiệu hóa ánh sáng và kiểm tra độ sâu.
 - 6) Ràng buộc texture khởi tạo là 0.
 - 7) Kích hoạt tính năng và ràng buộc các chương trình con và xác định đầu vào của nó.
 - 8) Vẽ textured dọc theo trục z. Các tọa độ xy đỉnh là (-1 -1), (1 - 1), (1 1), (-1 1). Các tọa độ texture xy tương ứng là (0, 0), (1, 0), (1, 1), (0, 1). Các tọa độ vertex và texture theo trục z tăng thống nhất từ -1 đến 1 và 0 -1, tương ứng.
- Ví dụ: hình 2.16 cho thấy các hình ảnh được tạo ra bởi volume renderer.



Hình 2.16. Kết quả thực hiện của ứng dụng hiển thị hình ảnh sử dụng khối

2.2.3. Chi tiết thực hiện

Phần này trình bày tổng quan về các thành phần thường được sử dụng trong các ứng dụng dựng texture-based nêu ở trên. Mục đích là để cung cấp đủ chi tiết để làm cho nó dễ dàng hơn để hiểu được hiện thực tiêu biểu của volume renderers mà sử dụng hiện nay thế hệ phần cứng đồ họa của người tiêu dùng, chẳng hạn như họ GeForce FX.

* *Dữ liệu mô tả và xử lý*

Bộ dữ liệu thử nghiệm gồm nhiều kích cỡ, chủng loại. Để hiển thị khối, dữ liệu được lưu trữ trong bộ nhớ trong một định dạng phù hợp, do đó nó có thể được dễ dàng tải về bộ nhớ đồ họa như kết cấu. Thông thường, khi khối lượng được lưu trữ trong một mảng 3D duy nhất. Tùy thuộc vào loại hình học thay thế sử dụng, hoặc một đối tượng kết cấu 3D duy nhất hoặc 1-3 bộ kết cấu lát cắt 2D được tạo ra. Các nhà phát triển cũng đã chọn sẵn định dạng kết cấu sử dụng để render. Ví dụ, kết cấu quyền lực của hai kích thước thường được sử dụng để tối đa hóa hiệu suất hiển thị. Thường xuyên, tập dữ liệu không phải là ở dạng đúng và không đúng kích cỡ, và nó có thể không phù hợp với các kết cấu có sẵn bộ nhớ trên GPU. Trong việc triển khai đơn giản, xử lý dữ liệu được thực hiện trong một bước riêng biệt bên ngoài các hiển thị. Trong tình huống phức tạp, xử lý dữ liệu sẽ trở thành một phần không thể thiếu của

các ứng dụng, chẳng hạn như khi các giá trị dữ liệu được tạo ra trên bay hoặc khi hình ảnh được tạo ra trực tiếp từ dữ liệu thô.

Để thay đổi kích thước của một tập hợp dữ liệu, người ta có thể đổi mẫu nó thành một dạng thô hoặc dạng lưới mịn hơn, hoặc đệm nó ở ranh giới. Việc tạo đệm được thực hiện bằng cách đặt các dữ liệu vào một khối lớn hơn và làm đầy các vùng trống, với các giá trị. Sự lấy mẫu lại yêu cầu thăm dò khối (có nghĩa là, tính toán nội suy các giá trị dữ liệu tại một vị trí từ các láng giềng voxel). Mặc dù kỹ thuật nội suy tam giác thường được sử dụng là dễ thực hiện, nó không phải luôn luôn là sự lựa chọn tốt nhất cho sự lấy mẫu lại, bởi vì nó có thể giới thiệu hiện vật trực quan. Nếu chất lượng tái tạo cũng là rất quan trọng cho các ứng dụng, chức năng nội suy phức tạp hơn được yêu cầu, chẳng hạn như đa thức từng phần theo khối. May mắn thay, các hoạt động đó được dễ dàng thực hiện với sự giúp đỡ của các bộ công cụ có sẵn công khai. Ví dụ, các bộ công cụ Teem bao gồm một loạt các công cụ xử lý dữ liệu có thể truy cập trực tiếp từ dòng lệnh, phơi bày các chức năng của thư viện cơ bản mà không cần phải viết bất kỳ mã. Ví dụ về cách sử dụng Teem cho khối lượng xử lý dữ liệu có trên đĩa CD và Web site của cuốn sách. Xử lý dữ liệu cao cấp có thể được thực hiện trên GPU, ví dụ, để tạo ra hình ảnh chất lượng cao.

Kỹ thuật chiếu sáng địa phương và chức năng chuyển đa chiều sử dụng thông tin gốc trong khi hiển thị. Hầu hết các trường sử dụng sự khác biệt trung tâm để có được các vector gradient ở mỗi voxel. Các phương pháp khác biệt xấp xỉ toàn cục gradient là chênh lệch giữa giá trị dữ liệu của hai láng giềng voxel dọc theo một trục tọa độ, chia cho khoảng cách vật lý. Ví dụ, công thức sau đây tính toán các thành phần x của vector gradient ở vị trí voxel $\vec{P}_{(i,j,k)}$

Phương trình (2) Gradient Computation Sử dụng khác nhau Trung

$$g_x(\vec{P}_{(i,j,k)}) = \frac{v(\vec{P}_{(i+1,j,k)}) - v(\vec{P}_{(i-1,j,k)})}{2h}$$

trong đó h là khoảng cách giữa các voxel dọc theo trục x . Để có được gradient ở ranh giới dữ liệu, khối lượng được đệm bằng cách lặp lại các giá trị biên. Hình ảnh nhiễu gây ra bởi sự khác biệt trung ương tương tự như kết quả từ resampling với Trilinear suy. Nếu chất lượng hình ảnh là quan tâm, chức năng phát sinh phức tạp hơn là cần thiết, chẳng hạn như những cái mà Teem cung cấp. Tùy thuộc vào định dạng kết cấu được sử dụng, các gradient tính có thể cần phải được lượng tử hóa, thu nhỏ và thiên vị để phù hợp với phạm vi có sẵn của các giá trị.

Chức năng truyền khu vực trong khối bằng cách gián màu sắc và độ mờ đục để giá trị dữ liệu. Biểu đồ rất hữu ích cho việc phân tích mà phạm vi của các giá trị quan trọng trong dữ liệu. Nói chung, biểu đồ thể hiện sự phân bố của các giá trị dữ liệu và các biện pháp dữ liệu khác có liên quan. Một biểu đồ 1D được tạo ra bằng cách phân chia các dải giá trị vào một số thùng. Mỗi khối chứa số voxels trong giới hạn trên và dưới được giao vào thùng. Bằng cách kiểm tra các biểu đồ, ta có thể thấy những giá trị thường xuyên trong các dữ liệu. Tuy nhiên biểu đồ không thể hiện sự phân bố không gian của các mẫu trong các khối lượng.

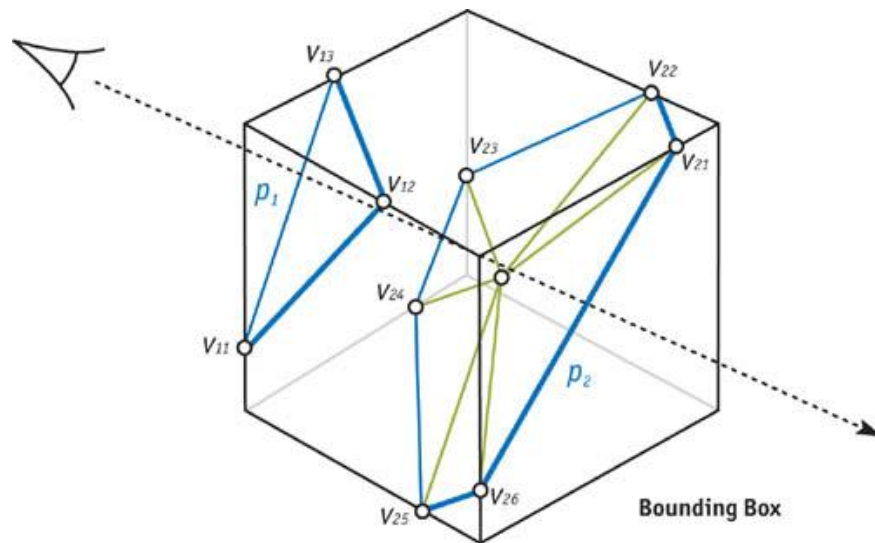
Các đầu ra của bước xử lý dữ liệu là một tập hợp các kết cấu được tải về cho GPU trong giai đoạn sau. Đó là đôi khi hiệu quả hơn để kết hợp một số kết cấu thành một kết cấu đơn. Ví dụ, để giảm chi phí tra cứu kết cấu và nội suy, các giá trị và kết cấu dốc bình thường thường được lưu trữ và sử dụng cùng nhau trong một kết cấu RGBA duy nhất.

* *Hình học thay thế*

Trong giai đoạn dựng hình, hình ảnh của khối được tạo ra bằng cách vẽ hình học thay thế trong thứ tự sắp xếp. Khi bộ dữ liệu được lưu trữ trong một

kết cấu 3D, mặt phẳng xem liên kết được sử dụng để cắt khối bao, kết quả là một tập hợp các đa giác cho lấy mẫu khối. Thuật toán 2 tính hình học thay thế trong khung nhìn không gian bằng cách sử dụng ma trận nhìn cho chuyển vertex giữa các đối tượng và xem các hệ tọa độ. Đa giác thay thế được chia thành hình tam giác, và các vertex kết quả được lưu trữ trong một mảng vertex cho dựng hình hiệu quả hơn.

Hình 2.16 minh họa thuật toán 2 với lát cắt hai đa giác. Các đa giác đầu tiên chứa ba đỉnh, thứ hai gồm có sáu đỉnh và được chia thành sáu tam giác.



Hình 2.17. Lát cắt hai đa giác

Có một số cách để tạo ra tọa độ texture cho các đỉnh đa giác. Ví dụ, tọa độ texture có thể được tính trên CPU trong bước 3 (c) của thuật toán 2 từ vị trí vertex tính và hộp khối bao. Trong trường hợp này các tọa độ được gửi xuống cho bộ nhớ GPU trong một mảng vertex riêng hoặc xen kẽ với các dữ liệu đỉnh. Có nhiều phương pháp khác nhau để tính toán tọa độ texture trên GPU, bao gồm cả kết cấu tự động phối hợp thể hệ, các ma trận kết cấu hoặc với một chương trình vertex.

Các thuật toán tiên tiến, như được mô tả có thể sử dụng một trục cắt khác nhau hơn so với hướng nhìn. Trong trường hợp này, các thuật toán hoạt động

theo cách tương tự, nhưng các ma trận modelview cần phải được sửa đổi cho phù hợp.

Thuật toán 2: biến đổi khối dữ liệu sắp xếp hình thành 3D: Quy trình chuyển đổi khối dữ liệu thành hình ảnh thường gồm các bước sau:

1) Xây dựng khối tương ứng với các pixel trên tập ảnh

2) Khởi tạo vùng hiển thị $M = \text{null}$;

3) Với mỗi khối con chúng ta tiến hành kiểm tra:

+) Nếu tất cả các đỉnh của khối con đều có giá trị $>$ ngưỡng thì các đỉnh của khối con này không được chọn.

+) Nếu tất cả các đỉnh của khối con đều có giá trị $<$ ngưỡng thì các đỉnh của khối con này cũng không được chọn

+) Nếu tồn tại đỉnh của khối con có giá trị $<$ ngưỡng và đồng thời cũng tồn tại đỉnh có giá trị $>$ ngưỡng thì lựa chọn khối con và xác định các đỉnh của khối con được đưa vào tập đỉnh của mô hình. $M = M \cup T$, với T là đa giác được tạo bởi các đỉnh của khối con.

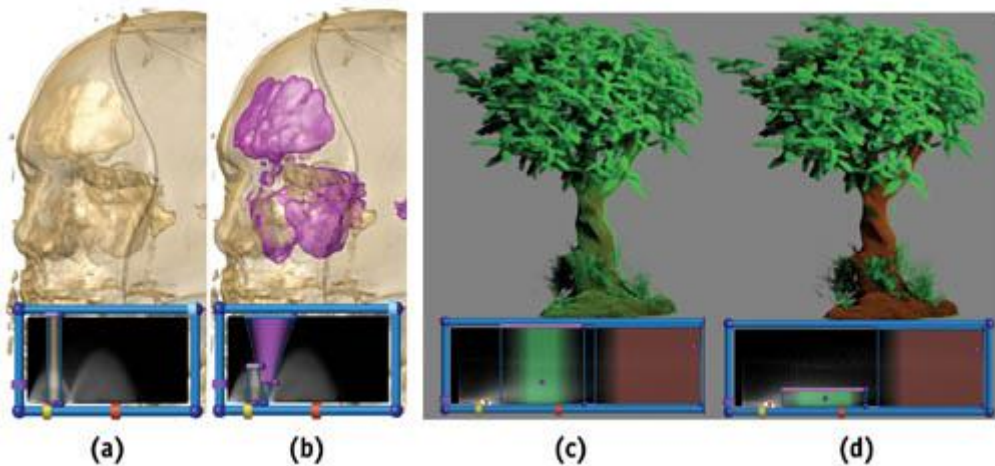
3) Chuẩn hóa vùng hiển thị M

* **Rendering**

Hàm truyền

Vai trò của hàm truyền là để nhấn mạnh các tính năng trong các dữ liệu của giá trị bản đồ và các biện pháp khác để dữ liệu thuộc tính quang học. Cách đơn giản nhất và được sử dụng rộng rãi nhất các hàm truyền là một chiều và họ lập bản đồ các dãy các giá trị dữ liệu đến màu sắc và độ mờ đục. Thông thường, các hàm truyền được thực hiện với 1D bảng tra cứu kết cấu. Khi các bảng tra cứu được xây dựng, màu sắc và độ mờ đục thường được gián một cách riêng biệt bởi các chức năng chuyên giao. Để vẽ chính xác, các thành phần màu cần phải được nhân với opacity, vì màu sắc xấp xỉ cả các phát xạ và hấp thụ trong một phân đoạn tia (*màu trong suốt*).

Sử dụng giá trị dữ liệu là biện pháp duy nhất để kiểm soát sự phân công của màu sắc và độ mờ đục có thể hạn chế hiệu quả của tính năng phân loại trong các dữ liệu. Kết hợp các biện pháp dữ liệu khác vào các hàm truyền, chẳng hạn như độ lớn gradient, cho phép kiểm soát tốt hơn và trực quan phức tạp hơn. Ví dụ, xem hình 2.17 để minh họa sự khác biệt giữa việc sử dụng một và hàm truyền hai chiều dựa trên các giá trị dữ liệu và độ lớn gradient.



Hình 2.18. Sự khác biệt giữa các hàm truyền 1D và 2D

Thiết kế hàm truyền là một thủ tục lặp đi lặp lại khó mà đòi hỏi cái nhìn sâu sắc đáng kể vào các thiết lập dữ liệu cơ bản. Một số thông tin được cung cấp bởi các biểu đồ của giá trị dữ liệu, cho thấy trong đó khoảng các giá trị cần được nhấn mạnh. Giao diện người dùng là một thành phần quan trọng của quá trình thiết kế tương tác. Thông thường, giao diện bao gồm một trình biên tập đường cong 1D để xác định chức năng chuyển giao thông qua một tập hợp các điểm kiểm soát. Một phương pháp khác là sử dụng các vật dụng thao tác trực tiếp với sơn trực tiếp vào kết cấu chức năng chuyển giao. Các vật dụng cung cấp một cái nhìn về sự phân bố chung của các giá trị dữ liệu, đại diện bởi các trục ngang, và độ lớn gradient đại diện bởi các trục thẳng đứng. Nguyên tử trong giá trị và phân phối cường độ dốc cho thấy sự hiện diện của ranh giới vật chất. Một bộ bàn chải được cung cấp cho sơn vào các chức năng phụ thuộc kết

cầu chuyển 2D, trong đó giao cho màu sắc và độ mờ đục để Voxels với phạm vi tương ứng của các giá trị dữ liệu và độ lớn gradient.

Độ trong suốt giao cũng phụ thuộc vào tỷ lệ lấy mẫu. Ví dụ khi sử dụng lát cắt ít hơn, độ mờ đục đã được nhân rộng, vì vậy mà cường độ tổng thể của hình ảnh vẫn giữ nguyên.

Chiếu sáng

Mô hình chiếu sáng được sử dụng để cải thiện sự xuất hiện hình ảnh của các đối tượng. Mô hình đơn giản tại địa phương gần đúng cường độ ánh sáng phản xạ từ bề mặt của một đối tượng. Cường độ tính toán được sử dụng để điều chỉnh các thành phần màu sắc từ các chức năng chuyển giao.

Cường độ tính toán được sử dụng để điều chỉnh các thành phần màu sắc từ các chức năng chuyển giao. Trong khối lượng render các ứng dụng, các vector gradient bình thường được sử dụng như các bề mặt bình thường. Thật không may, các gradient là không được xác định rõ trong các vùng đồng nhất của khối lượng. Đối với khối lượng vẽ mô hình Blinn-Phong thường xuyên được sửa đổi, do đó chỉ những vùng có độ lớn dốc cao được tô.

2.2.4. Kỹ thuật nâng cao

Phần này mô tả các kỹ thuật để nâng cao chất lượng dựng hình và tạo các hiệu ứng đặc biệt volumetric.

* *Chiếu sáng volumetric*

Các mô hình chiếu sáng địa phương đã trình bày trong phần trước cho biết thêm tín hiệu thị giác quan trọng để các rendering. Như một mô hình đơn giản là không thực tế, tuy nhiên, vì nó cho rằng ánh sáng tới một mẫu mà không tương tác với phần còn lại của khối lượng. Hơn nữa, loại ánh sáng giả định một mô hình bề mặt dựa trên, đó là không phù hợp cho các vật liệu tích. Một cách để kết hợp hiệu ứng ánh sáng phức tạp, chẳng hạn như bóng tích, là để precompute một khối lượng bóng tối để lưu trữ số lượng ánh sáng đến ở mỗi mẫu sau khi bị suy yếu bởi các vật liệu can thiệp. Trong thời gian

render, các giá trị nội suy từ bản đồ bóng tích này được nhân với màu sắc từ các chức năng chuyển giao. Nhưng ngoài việc sử dụng bộ nhớ thêm, bản đồ bóng tích dẫn đến hiện vật trực quan như bóng mờ và hình ảnh tối.

Một lựa chọn tốt hơn là sử dụng một bộ đệm pixel để tích lũy số lượng ánh sáng yếu đi từ quan điểm của ánh sáng của view. Để làm điều này một cách hiệu quả, các trục cắt được đặt nằm giữa view và hướng ánh sáng. Điều này cho phép cùng một lát để được trả lại từ cả hai mắt và những điểm sáng. Lượng ánh sáng khi đến một phần cụ thể là bằng 1 trừ đi opacity tích lũy của những lát trả trước. Mỗi slice là lần đầu tiên kết xuất từ quan điểm của mắt nhìn, sử dụng kết quả của vượt qua trước kết xuất từ quan điểm của ánh sáng nhìn, được sử dụng để điều chỉnh độ sáng của mẫu trong miếng hiện nay. Cùng lát sau đó được kết xuất từ quan điểm của ánh sáng nhìn để tính toán cường độ của ánh sáng tới được lát tiếp theo.

Thuật toán 3: Two-Pass Volume Rendering với bóng

1. Xóa bộ đệm mắt và khởi tạo bộ đệm ánh sáng với màu sắc ánh sáng C_L . Một bản đồ kết cấu có thể được sử dụng để khởi tạo các bộ đệm ánh sáng để tạo hiệu ứng đặc biệt, chẳng hạn như đèn sân khấu.
2. Tính toán hình học thay thế trong không gian đối tượng sử dụng thuật toán 1. Khi chấm sản phẩm của ánh sáng và hướng dẫn xem là tích cực, thiết lập các hướng slice đến nửa đường giữa ánh sáng và hướng nhìn. Trong trường hợp này, khối lượng là ra phía trước để sao cho mắt bằng cách sử dụng các nhà điều hành bên dưới. Khi sản phẩm dot là tiêu cực, cắt dọc theo nửa chùng vector giữa ánh sáng và hướng nhìn ngược, và làm cho khối lượng trở về trước cho mắt bằng cách sử dụng các nhà điều hành. Trong cả hai trường hợp, làm cho phía trước khối lượng để sao cho ánh sáng bằng cách sử dụng các nhà điều hành.
3. Đối với mỗi lát cắt:
 - a) Hiện thị và pha trộn các lát cắt trong bộ đệm.

+) Chiếu đỉnh lát vào bộ đệm bằng cách sử dụng ánh sáng và chiếu modelview ma trận của ánh sáng. Chuyển đổi các vị trí đỉnh dẫn đến tọa độ texture 2D dựa trên kích thước của khung nhìn của ánh sáng và bộ đệm ánh sáng.

+) Ràng buộc đệm nhẹ như một kết cấu cho một đơn vị kết cấu có sẵn và sử dụng các tọa độ texture tính ở bước 3 (a). Nhớ lại rằng một tập hợp các tọa độ 3D kết cấu cũng là cần thiết cho việc tra cứu kết cấu dữ liệu.

+) Trong shader mảnh, đánh giá các chức năng chuyển giao cho màu C phản xạ và opacity A . Tiếp theo, nhân C bởi các màu sắc từ ánh sáng đệm C_L , và pha trộn C và A vào bộ đệm mắt bằng cách sử dụng các nhà điều hành thích hợp cho việc định hướng lát hiện nay.

b) Hiển thị và pha trộn các miếng đệm vào ánh sáng bằng cách sử dụng các nhà điều hành. Trong đánh bóng đánh giá các hàm truyền cho các thành phần alpha và thiết lập màu fragment 0.

Bóng khối cải thiện đáng kể tính hiện thực của cảnh trả lại, như thể hiện trong hình 2.18. Lưu ý rằng khi góc giữa người quan sát và hướng thay đổi ánh sáng, khoảng cách lát cần được điều chỉnh để duy trì một tỷ lệ lấy mẫu liên tục.



Hình 2.19. Các ví dụ của kết quả với bóng

Tạo ra những hình ảnh đen tối và không thực tế, bởi vì nó bỏ qua sự đóng góp của tán xạ ánh sáng bên trong khối lượng. Hiệu ứng tán xạ có thể được chụp hoàn toàn thông qua các mô hình chiếu sáng khối lượng dựa trên vật lý, đó là quá phức tạp để dựng hình tương tác.

Một trong những hiện tượng mờ, đó là kết quả của ánh sáng truyền vào và phân tán trong chất. Trong khi tính toán phân tán nói chung xem xét ánh sáng đến từ mọi hướng, cho mờ nó là đủ để bao gồm ánh sáng đến trong một hình nón theo hướng của nguồn sáng duy nhất. Kết quả của việc đơn giản hóa này là sự đóng góp gián tiếp tán xạ tại một mẫu cụ thể phụ thuộc vào một khu phố địa phương của mẫu tính trong phiên trước đó. Do đó, hiệu ứng mờ là có thể kết hợp bằng cách truyền bá và làm mờ các thành phần ánh sáng gián tiếp từ lát cắt lát ở khối.

*** Phương pháp Rendering**

Một nhược điểm của rendering khối lượng là các chi tiết tần số cao nhỏ không thể được đại diện trong khối lượng độ phân giải thấp. Chi tiết tần số cao là rất cần thiết cho việc chụp những đặc điểm của nhiều hiện tượng tích như mây, khói, cây xanh, tóc và lông thú. Mô phỏng tiếng ồn thủ tục là một kỹ thuật mạnh mẽ để thêm chi tiết dung lượng sử dụng độ phân giải thấp. Các cách tiếp cận chung sử dụng một mô hình thô cho macrostructure và tiếng ồn thủ tục cho các cấu trúc vi mô. Mô tả tiếp theo là hai cách để thêm tiếng ồn thủ tục để khối lượng kết xuất họa dựa trên. Các phương pháp tiếp cận đầu tiên perturbs tính chất quang học trong giai đoạn đánh bóng phương pháp thứ hai làm nhiễu chính đối tượng.

Cả hai phương pháp sử dụng một khối nhiễu nhỏ. Trong tập này, mỗi voxel được khởi tạo tới bốn con số ngẫu nhiên, được lưu trữ như các thành phần RGBA, mờ hơi để ẩn nhiễu gây ra bởi nội suy tam giác. Nhiều bản sao của các kết cấu tiếng ồn được sử dụng cho từng slice tại quy mô khác nhau trong quá trình dựng hình. Mỗi pixel nhiễu loạn được tính bằng tổng trọng số của các thành phần nhiễu cá nhân. Để animate nhiễu loạn, một vector bù đắp khác nhau có thể được thêm vào các tọa độ texture nhiễu trong đánh bóng.

Kỹ thuật đầu tiên sử dụng bốn thành phần nhiễu trên một điểm ảnh để thay đổi tính chất quang học của các khối lượng sau khi chuyển giao chức năng đã được đánh giá. Điều này dẫn đến những vật liệu hiện có sai phạm. Lựa chọn mà tính chất quang học để sửa đổi, hiệu ứng khác nhau đã đạt được.

CHƯƠNG 3: CHƯƠNG TRÌNH THỬ NGHIỆM HIỂN THỊ ẢNH Y TẾ DICOM THEO THÀNH PHẦN

3.1. Phân tích yêu cầu bài toán

Ảnh y tế có rất nhiều các chuẩn khác nhau, trong luận văn này sử dụng chuẩn ảnh DICOM. Để lấy được dữ liệu từ chuẩn DICOM thì ta phải xây dựng một bộ phân tích cú pháp tập tin DICOM. Dữ liệu của một tập tin DICOM gồm hai phần chính. Phần Header là phần dữ liệu chứa các thông tin chung và phần Data Set phần dữ liệu về hình ảnh, đây là phần dữ liệu chúng ta cần phân tích và xử lý.

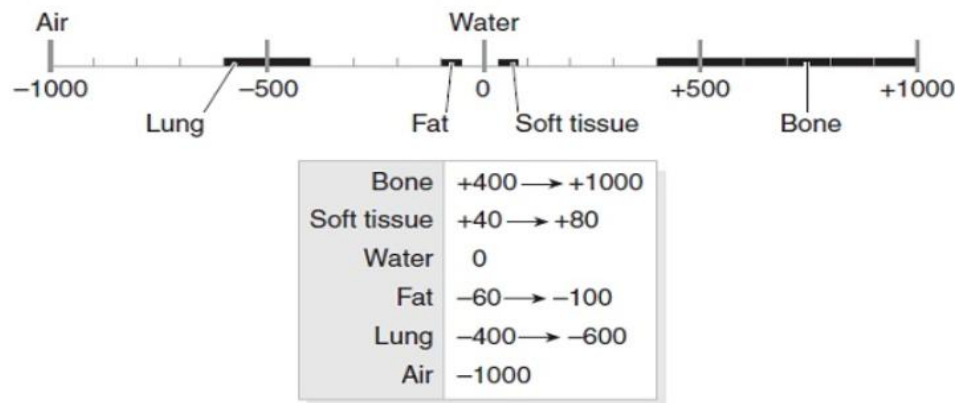
Vấn đề lớn nhất trong xử lý ảnh DICOM là làm thế nào chúng ta có thể phân tích, bóc tách được đâu là vùng dữ liệu của các thành phần như: xương, mô mềm, nước, mỡ, không khí... để thực hiện được điều này ta phải tiến hành xây dựng một công cụ chuyển dữ liệu của Data set thành dữ liệu ảnh mức xám theo hai bước.

Bước 1: Chuyển các giá trị điểm ảnh từ dữ liệu data set thành giá trị Hounsfield. Đối với ảnh CT, mối quan hệ giữa các giá trị được lưu trữ (SV) và các giá trị Hounsfield được xác định theo công thức sau:

$$HU = SV * RescaleSlope + RescaleIntercept$$

Trong đó: RescaleSlope và RescaleIntercept được lấy dữ liệu từ phần meta của tập tin DICOM.

Bước 2: Chuyển đổi các giá trị Hounsfield (HU) vào các giá trị mức xám. Khi các giá trị lưu trữ (SV) được quy đổi ra giá trị Hounsfield, các giá trị Hounsfield phải được chuyển đổi tiếp tục sang các giá trị mức xám có ý nghĩa. Các giá trị Hounsfield khoảng từ -1000 đến +1000 như hình dưới đây:



Hình 3.1. Bảng giá trị mức xám có ý nghĩa trong dữ liệu ảnh DICOM chuyển đổi

Sau khi đã chuyển đổi được dữ liệu data set sang dữ liệu mức xám chúng ta có thể dễ dàng lấy ra các vùng dữ liệu theo mong muốn dựa vào ảnh các giá trị mức xám như: vùng là xương có giá trị xấp xỉ từ +400 đến +1000, vùng mô mềm có giá trị mức xám xấp xỉ từ +40 đến +80.

Sau khi đã tiến hành phân tích bóc tách dữ liệu data set tiếp theo tiến hành xây dựng mô hình 3D từ tập ảnh CT theo phương pháp Volume rendering như sau:

- *) Đầu vào: Tập ảnh (w,h,n)
- *) Phương pháp: Volume rendering
- *) Đầu ra: Mô hình 3D (Volume rendering)

3.2. Phân tích lựa chọn công cụ

Môi trường cài đặt trên hệ điều hành window 10 có cài đặt phần mềm hỗ trợ Microsoft .Net Framework 3.5

Ngôn ngữ lập trình C++ với sự hỗ trợ của thư viện đồ họa máy tính VTK. Máy tính Dell Vostr 5560 core I5 thế hệ 3, Ram 4GB.

Trên cơ sở tìm hiểu nhiều bộ công cụ hỗ trợ lập trình đồ họa người viết đã chọn bộ công cụ VTK để xây dựng phần mềm của mình vì những lí do:

+) VTK là hai trong những bộ công cụ đ ược sử dụng nhiều nhất trên thế giới. Hầu hết các bài giảng về xử lí ảnh, đồ họa 3D và trực quan hóa của các

trường đại học lớn trên thế giới đều lấy VTK làm ví dụ minh họa, trên thế giới cũng có nhiều phần mềm hay được phát triển từ VTK.

+) VTK được thiết kế rất chuyên nghiệp: hai bộ toolkit này được viết bằng C++ theo kiểu hướng đối tượng, có thể sử dụng trong nhiều ngôn ngữ khác như Java, Tcl, Python, .NET, ...; hai bộ toolkit này có mã nguồn mở và liên tục được nâng cấp.

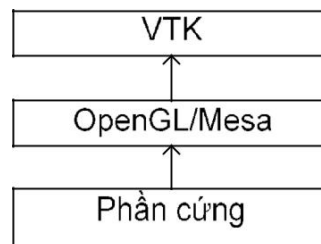
+) VTK có nhiều hỗ trợ cho xử lý ảnh y tế.

+) VTK là một bộ toolkit có mã nguồn mở được xuất bản lần đầu vào năm 1994, hiện nay VTK đang được quản lý và phát triển bởi **Kitware Inc.** VTK có thể ứng dụng vào đồ họa vi tính 3D và trực quan hóa. VTK cung cấp hàng trăm thuật toán xử lý ảnh và trực quan hóa. VTK được thiết kế theo kiểu hướng đối tượng trên nền ngôn ngữ C++ do đó nó có thể được sử dụng được cho các ngôn ngữ khác như Tcl/Tk, Python, Java. VTK được thiết kế để có thể chạy trên các hệ máy Unix, PC (Windows 9x/2000/NT), Mac OSX Jaguar.

Phiên bản VTK mới nhất 5.0 có khoảng 600 lớp với xấp xỉ 325 000 dòng mã. Việc sử dụng VTK tuân theo các quy định chung của thế giới về sử dụng mã nguồn mở.

****/Cấu trúc một ứng dụng VTK:***

VTK tương tác với phần cứng thông qua OpenGL/Mesa



Hình 3.2. Sơ đồ tương tác của VTK với phần cứng

Sơ đồ trên giới thiệu cấu trúc của một chương trình ứng dụng VTK. Nếu chỉ muốn kế thừa các lớp từ VTK chúng ta có thể xây dựng phần mềm theo kiểu “binary”. Ở hướng khác chúng ta sẽ sửa mã trong VTK theo ý muốn.

****/ Mô hình đối tượng (object model)***

Mô hình đối tượng của VTK gồm hai thành phần: mô hình đồ họa (graphics model) và mô hình trực quan hóa (visualization model)

+Tcl/Tk shell	+Tcl/Tk source
+Java interpreter	+Java JDK
+Python interpreter	+Python source
Nhân (core) C++	
File thư viện và các file tiêu đề (.dll file và .h file)	Tất cả các lớp trong mã nguồn (Khối lượng sẽ rất lớn)
Xây dựng theo kiểu binary	Xây dựng theo kiểu cài đặt mã nguồn

Bảng 3. Cấu trúc chương trình ứng dụng VTK

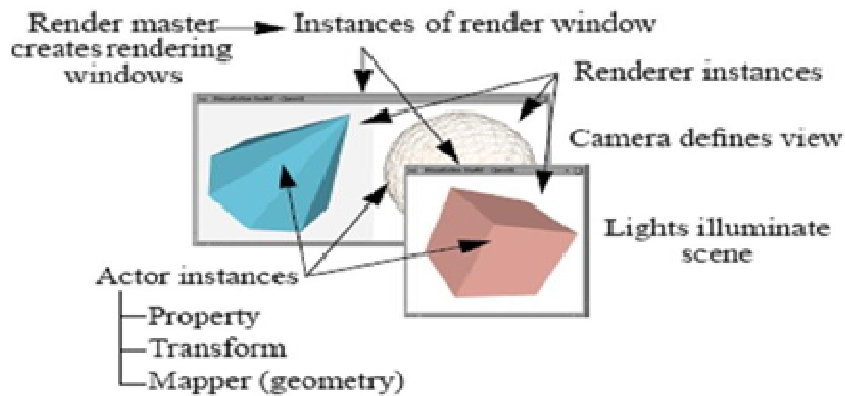
Mô hình đồ họa là mô hình cho đồ họa 3D. Mô hình này dựa trên các hệ thống đồ họa đã được sử dụng rộng rãi trong công nghiệp. Có chín đối tượng cơ bản trong mô hình đồ họa của VTK:

1. Render Master: định vị thiết bị và tạo render window. Nó độc lập với phương pháp.

2. Render Window: quản lý cửa sổ trên thiết bị hiển thị (display device). Một hoặc nhiều renderer cùng vẽ trong một render window để tạo nên một cảnh.

3. Renderer: xác định cách biểu diễn của light, camera và actor

4. Light: chiếu sáng actor trong cảnh.



Hình 3.3. Mô hình đồ họa của VTK

5. Camera: xác định vị trí nhìn, tiêu điểm và các thông số khác của camera.

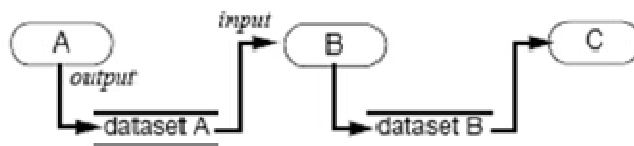
6. Actor: Actor xác định thời hạn hoạt động của đối tượng mapper, property và transform.

7. Property: xác định các thuộc tính của Actor như màu sắc, ánh sáng, texture map, kiểu vẽ,...

8. Mapper: xác định dạng hình học của actor

9. Transform : đối tượng này gồm một ma trận biến đổi 4x4 cùng các phương thức để thay đổi ma trận. Nó xác định vị trí và định hướng của actor, camera và light.

Mô hình trực quan hóa là mô hình dòng chảy dữ liệu (data flow, pipeline) trong quy trình trực quan hóa. Mô hình trực quan hóa của VTK dựa trên những mô hình đã được sử dụng trong các hệ thống thương mại.



Hình 3.4. Mô hình trực quan hóa của VTK

Các đối tượng xử lý (process objects) A, B, C nhận hoặc xuất ra một hay nhiều đối tượng dữ liệu (data objects).

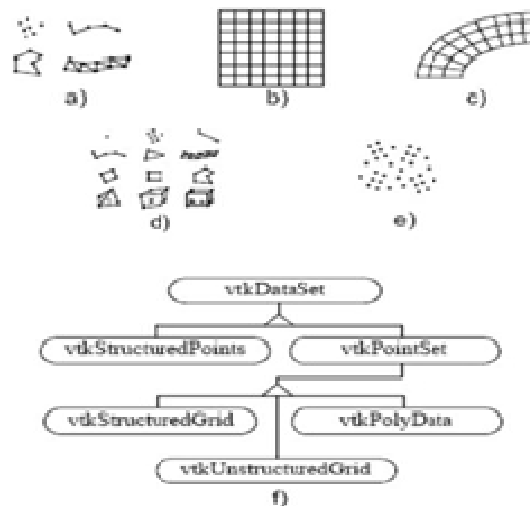
Đối tượng dữ liệu biểu diễn và cung cấp các phương thức truy cập dữ liệu; đối tượng xử lý thực hiện trên dữ liệu

Các đối tượng A, B, and C theo thứ tự là source, filter, và mapper objects

Trong mô hình trực quan hóa có hai loại đối tượng cơ bản là đối tượng xử lý (process object) và đối tượng dữ liệu (data object) (như ở hình 3.4).

+) Đối tượng xử lý: là các module hoặc các thuật toán. Đối tượng xử lý có thể chia làm 3 loại: sources, filter và mapper.

+) Đối tượng dữ liệu: là các tập dữ liệu (dataset). Trong VTK có 5 loại dữ liệu (hình 3.5)



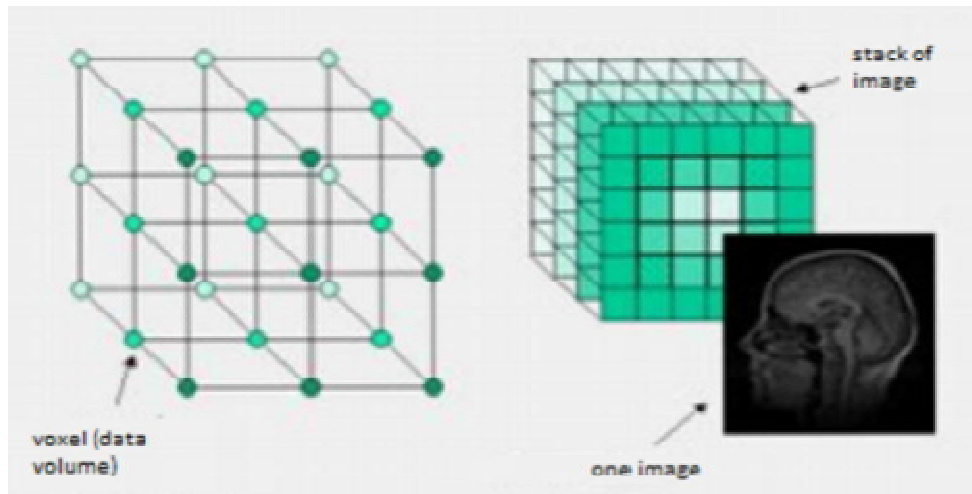
Hình 3.5. Các loại tập dữ liệu của VTK

a) polygonal data, b) structured points, c) structured grid, d) unstructured grid, e) unstructured points, f) sơ đồ các đối tượng theo sơ đồ OMT

3.3. Quy trình thực hiện

Bước 1:

Sắp xếp dữ liệu: Dữ liệu vào là các lát cắt song song, ta cần sắp xếp dữ liệu 2D này theo đúng thứ tự của chúng trong không gian thành khối dữ liệu (data volume).



Hình 3.6. Sắp xếp dữ liệu

Bước 2:

Biểu diễn: chuyển khối dữ liệu song song thành dữ liệu hình ảnh. Áp dụng phương pháp volume render.

Bước 3:

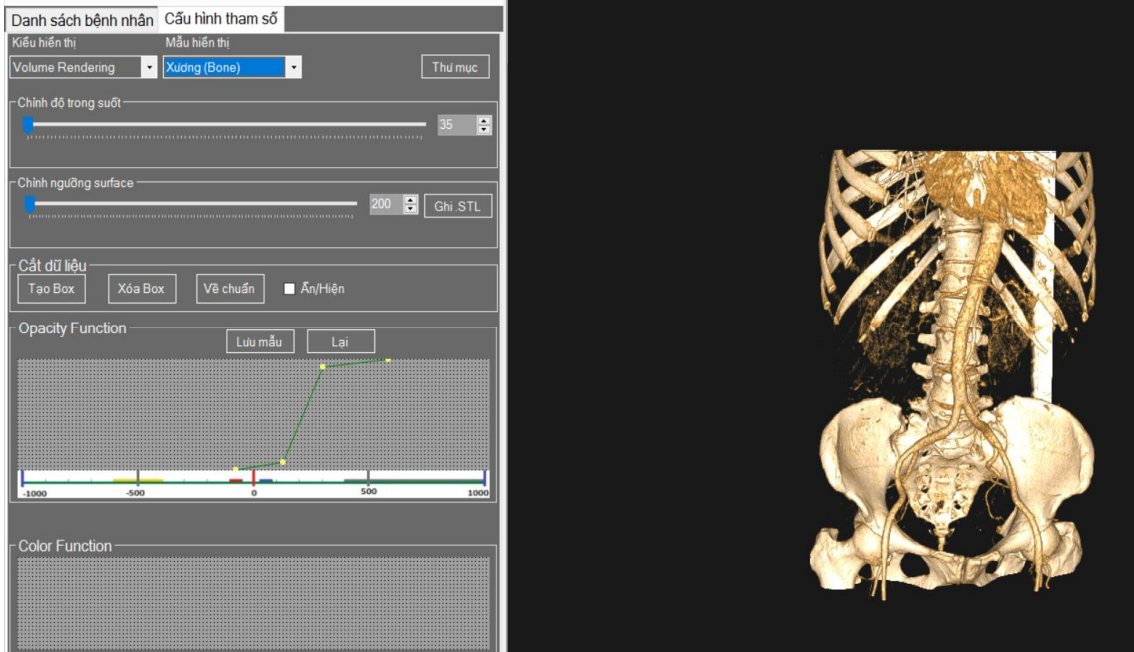
Biến đổi khối dữ liệu sắp xếp thành hình ảnh 3D: Quy trình chuyển đổi khối dữ liệu thành hình ảnh thường gồm các bước sau:

- 1) Sắp xếp, đọc ảnh ->(đầu ra: dữ liệu ảnh)
- 2) Tạo volume
- 3) Tính toán:
 - +) Tạo Texture – Base (1)
 - +) Tạo hình học đối tượng (2)
 - +) Kết hợp (1) và (2)
- 4) Rendering

3.4. Một số kết quả chương trình

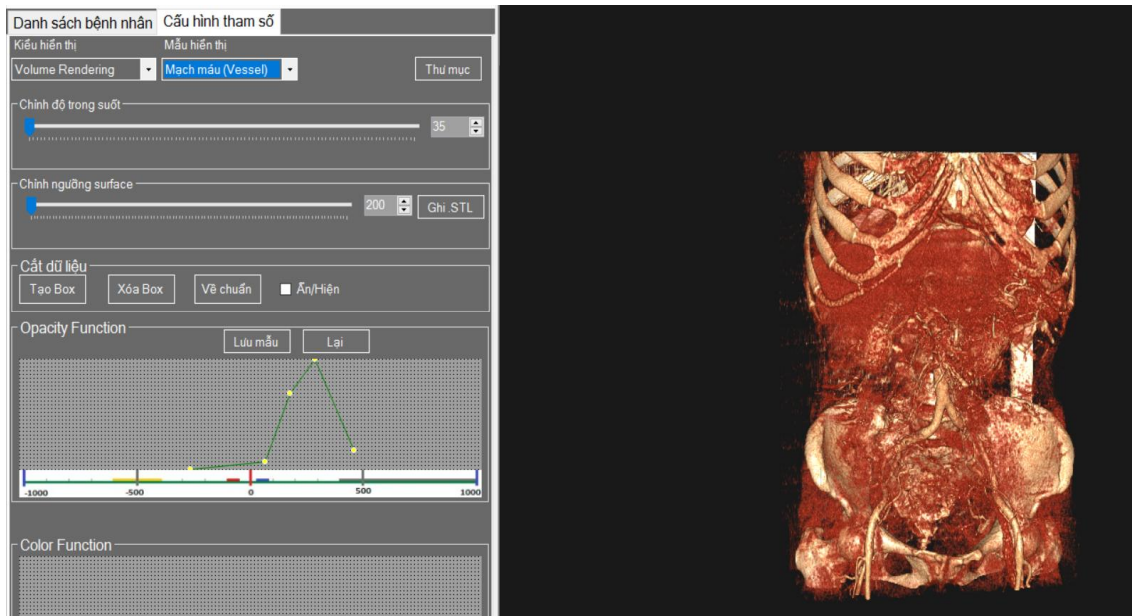
Các hình ảnh sau đây là kết quả thử nghiệm với dữ liệu chương trình demo.

- Xương: Có giá trị mức xám xấp xỉ từ -100 đến +600



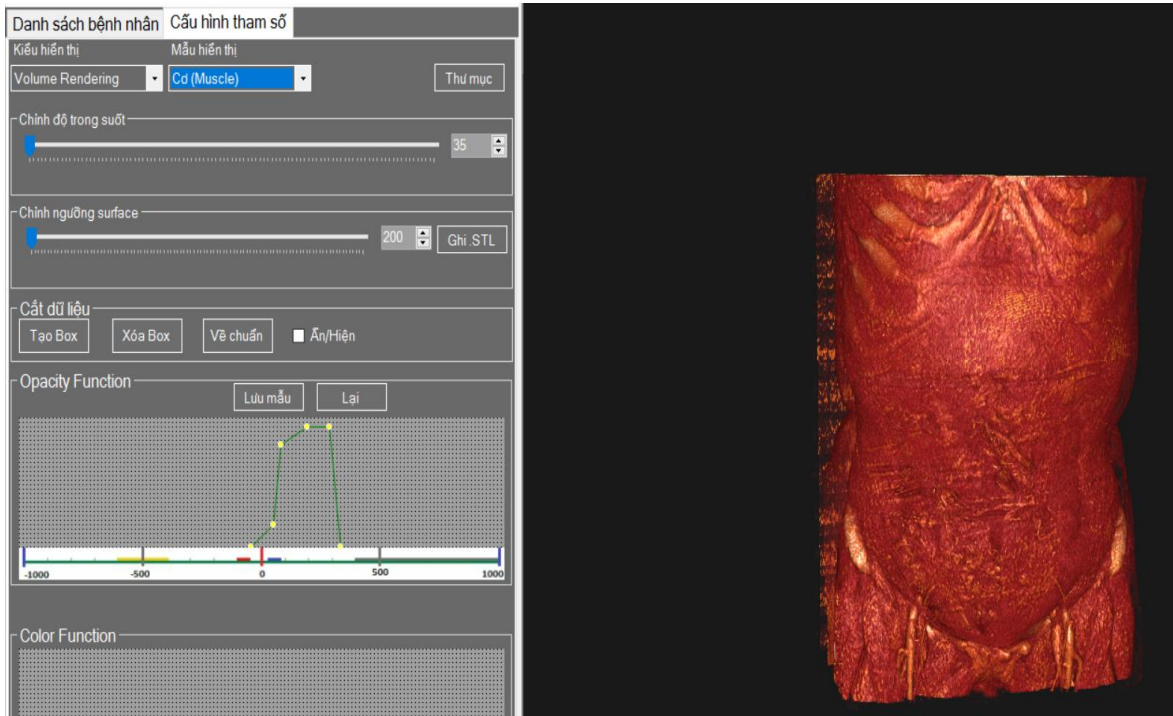
Hình 3.7. Hình ảnh của xương

- Mạch máu: Có giá trị mức xám xấp xỉ từ -300 đến +500



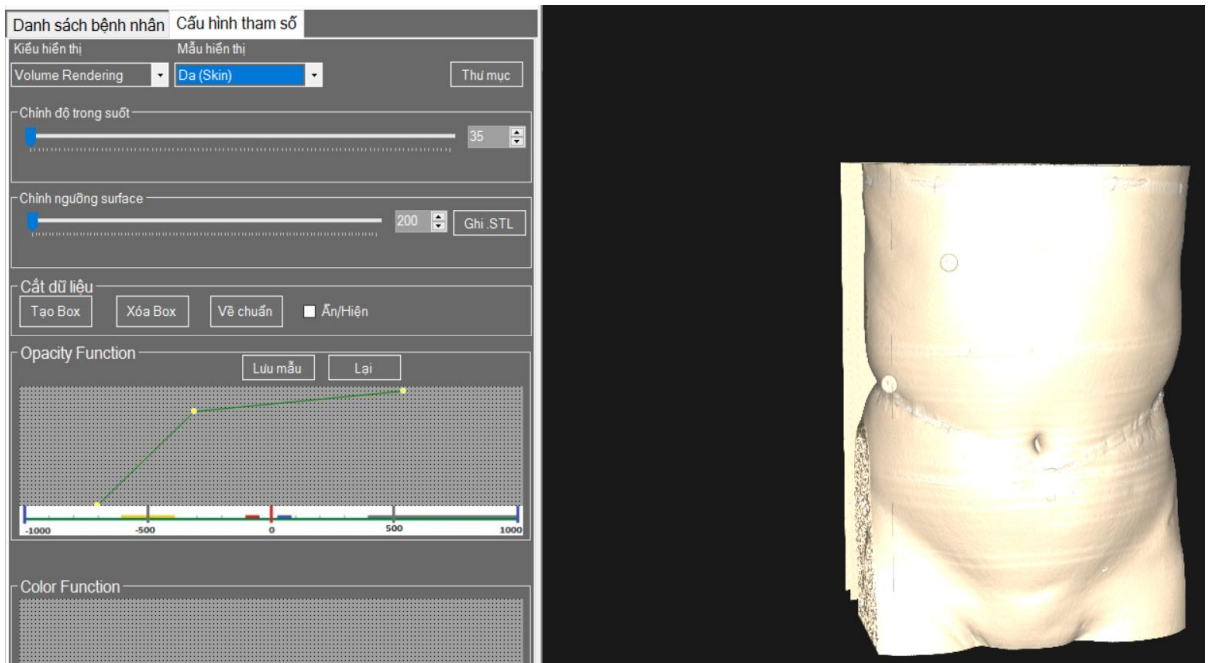
Hình 3.8. Hình ảnh của mạch máu

- Cơ: Có giá trị mức xám xấp xỉ từ -50 đến +300



Hình 3.9. Hình ảnh của cơ

- Da: Có giá trị mức xám xấp xỉ từ -700 đến +500



Hình 3.10. Hình ảnh của da

KẾT LUẬN

Có thể nói hình ảnh 3D ngày càng chiếm vai trò quan trọng trong việc chẩn đoán và điều trị, khi nghiên cứu về hình ảnh y tế là một lĩnh vực của ngành kỹ thuật y sinh. Việc nghiên cứu các giải pháp và giải thuật trong việc tái dựng hình ảnh 3 chiều trong y khoa để từ đó có thể xây dựng một lý thuyết và ứng dụng phục vụ trong công tác điều trị bệnh được tốt hơn.

Trên thế giới hiển thị hình ảnh 3D trong y tế đã có những bước tiến rất xa do đã được phát triển trong khoảng vài chục năm gần đây, có khác nhiều phần mềm thương mại tái tạo ảnh 3D từ các hình ảnh lát cắt và giá thành không hề rẻ.

Tại Việt Nam lĩnh vực phần mềm và xử lý ảnh y tế còn khá mới mẻ đối với người tiếp cận và người sử dụng. Đây là một lĩnh vực quan trọng mà nước ta đang dần dần tiến đến.

Cùng với xu thế đó dựa trên nhu cầu thực tế luận văn này mong muốn nêu ra những lý thuyết cũng như các kỹ thuật trong xử lý ảnh DICOM là làm thế nào để có thể phân tích, bóc tách được đâu là vùng dữ liệu cần lấy ra để từ đó đưa ra được những hình ảnh cắt lớp nói chung và từ lát cắt song song nói riêng.

Luận văn đã hoàn thành và đưa ra một số vấn đề như sau:

➤ Về lý thuyết

- +) Tổng quan về chuẩn DICOM và bài toán hiển thị ảnh
- +) Một số kỹ thuật hiển thị ảnh dicom
- +) Kỹ thuật Volume rendering
- +) Xây dựng chương trình thử nghiệm dựa trên kỹ thuật đã được hệ thống hóa.

➤ Về thực nghiệm

+) Đưa ra giải pháp phù hợp và xây dựng ứng dụng minh họa hiển thị ảnh y tế dựa trên kỹ thuật volume rendering và thuật toán Marching Cubes

+) Nghiên cứu và kế thừa các chương trình mã nguồn mở VTK phục vụ cho chương trình.

➤ Kết quả chưa làm được

Chương trình cài đặt thử nghiệm tuy cho chất lượng ảnh khá tốt nhưng với những bộ dữ liệu có số liệu ảnh ít thì chương trình cho chất lượng ảnh tương đối thấp. Bên cạnh đó với những bộ dữ liệu lớn thời gian chạy chương trình còn khá chậm.

➤ Hướng phát triển

Tiếp tục nghiên cứu thêm các phương pháp để đưa ra những hình ảnh cắt lớp tốt hơn. Thay vì hình ảnh sẽ tạo ra các mẫu

Tăng tốc độ tính toán về nội suy.

Tiếp tục đưa ra được chất lượng ảnh tốt hơn.

TÀI LIỆU THAM KHẢO

Tiếng Việt

- [1]. Dương Anh Đức, Lê Đình Huy, “Giới thiệu về đồ họa 3 chiều”
- [2]. PGS.TS Đỗ Năng Toàn, TS. Phạm Việt Bình (2007), “Giáo trình Xử lý ảnh” – Đại học Thái Nguyên.
- [3]. PGS. TS Nguyễn Quang Hoan (2006), *Xử lý ảnh*, Học viện Công nghệ Bưu chính Viễn thông.

Tiếng anh

- [1]. Digital Imaging and Communications in Medicine (DICOM) Published by National Electrical manufacturers Association 1300 N. 17th Street Rosslyn, Virginia 22209 USA.
- [2]. H.K.Huang (2010), “PACS and Imaging Informatics”, *Nhà xuất bản John Wiley & Sons*, Feb 12, 2010.
- [3]. Umamaheswari, J.; Radhamani, G.(2011), “A HYBRID APPROACH FOR DENOISING DICOM IMAGES”, *International Journal of Engineering Science & Technology*, Vol. 3 Issue 12, p7953.
- [4]. Smelyanskiy, M.; Holmes, D.; Chhugani, J.; Larson, A.; Carmean, D. M.; Hanson, D.; Dubey, P.; Augustine, K.; Kim, D.; Kyker, A.; Lee, V. W.; Nguyen, A. D.; Seiler, L.; Robb, R. (2009). "Mapping High-Fidelity Volume Rendering for Medical Imaging to CPU, GPU and Many-Core Architectures", *IEEE Transactions on Visualization and Computer Graphics* 15 (6): 1563–1570.
- [5]. Thomas M. Lehmann, Claudia Gonner, Klaus Spitzer, November 1999, “Survey: Interpolation Methods in Medical Image Processing”, *IEEE Transactions on Medical Imaging*, vol. 18, no. 1

- [6]. Lorensen U E, Cline H E. Marching cubes: A high resolution 3D surface construction algorithm. *ACMSIGGRAPH Computer Graphics*. 1987;21(4)
- [7]. Ebert, D., F. Musgrave, D. Peachey, K. Perlin, S. Worley, W. Mark, and J. Hart. 2002. *Texturing and Modeling: A Procedural Approach*. Morgan Kaufmann.
- [8]. William E. Lorensen, Harvey E. Cline, "Marching Cubes: A high Resolution 3D Surface Construction Algorithm"
- [9]. Kindlmann, G. 1999. "Semi-Automatic Generation of Transfer Functions for Direct Volume Rendering." Master's Thesis, Department of Computer Science, Cornell University.
- [10]. Engel, K., M. Kraus, and T. Ertl. 2001. "High-Quality Pre-Integrated Volume Rendering Using Hardware-Accelerated Pixel Shading." In *Proceedings of the SIGGRAPH/Eurographics Workshop on Graphics Hardware 2001*, pp. 9–16.