

MESHLAB: CÔNG CỤ XỬ LÝ MÔ HÌNH 3D VÀ CÁC PLUGIN BỔ SUNG

● HUỲNH CAO THẾ CƯỜNG - LÊ CÔNG ĐOÀN

TÓM TẮT:

Mô hình 3D được ứng dụng phổ biến trong nhiều lĩnh vực. Do đó, các mô hình 3D cần trải qua các bước tiền xử lý hoặc chỉnh sửa để thu được các thông tin hữu ích. Nhiều công cụ xử lý mô hình 3D đã được đề xuất với các chức năng xem, các bộ lọc và xuất dữ liệu. Bên cạnh các chức năng có sẵn, việc xây dựng thêm các chức năng phục vụ cho một yêu cầu cụ thể là cần thiết. Bài báo tập trung giới thiệu các chức năng chính của MeshLab và cùng với những plugin cho phép xem các định dạng mô hình trong y học, tiền xử lý và các tham số hóa mô hình. Kết quả cho thấy các plugin mới đã bổ sung và cải tiến các chức năng hiện có của MeshLab.

Từ khóa: MeshLab, mô hình 3D, plugin.

1. Đặt vấn đề

Mô hình 3D được ứng dụng trong nhiều lĩnh vực của cuộc sống, như điện ảnh, kiến trúc, giáo dục và y học. Trong điện ảnh, mô hình 3D được sử dụng trong việc tạo nhân vật và các khung cảnh 3D. Trong lĩnh vực điện ảnh, mô hình 3D được sử dụng trong việc tạo các nhân vật và khung cảnh ảo [1]. Mô hình 3D cũng được sử dụng trong việc mô hình các công trình, kiến trúc [2]. Pujol [3] đã đề xuất sử dụng mô hình 3D của các cơ quan để dạy về các khái niệm giải phẫu học. So với mô hình 2D, mô hình 3D cung cấp một cách quan sát đầy đủ hơn về hình ảnh và cấu trúc của đối tượng và đã trở nên phổ biến.

Tuy nhiên mô hình 3D cần được xử lý để có thể thu được những thông tin mong muốn. Chẳng hạn, trong ứng dụng y khoa, ảnh thu được từ máy scan CT (computed tomography) hoặc MRI (Magnetic resonance imaging) của một lá gan có thể được

biểu diễn bởi khoảng 100 đến 500 lát cắt 2D (2D slices). Để có thể mô hình hóa được dưới dạng 3D, sau khi được phân đoạn (segmentation), hình ảnh cần được chuyển sang định dạng 3D và cần được xử lý để trích xuất những thông tin liên quan theo yêu cầu cụ thể. Do đó, việc lựa chọn một công cụ xử lý phù hợp cùng với việc xây dựng các chức năng bổ sung theo mục đích cụ thể là điều cần thiết.

Bài báo giới thiệu một công cụ xử lý mô hình 3D - MeshLab và trình bày cách tạo mới một vài chức năng (plugin) hữu ích, cụ thể chức năng đọc hình ảnh từ dữ liệu ảnh định dạng MetaImage Header (mhd), được dùng phổ biến để lưu hình ảnh y tế; tiền xử lý (đơn giản hóa mô hình, làm mịn và chuẩn hóa); số hóa mô hình 3D (parameterization). Nội dung tiếp theo của bài báo được tổ chức như sau: Phần 2 giới thiệu công cụ MeshLab và cách tạo mới các plugin. Phần 3 thảo luận một số kết quả và thảo luận. Phần 4 trình bày kết luận.

2. MeshLab và các plugin bổ sung

2.1. Giới thiệu các công cụ xử lý phổ biến

Sự phổ biến của việc áp dụng mô hình 3D đã dẫn đến sự ra đời nhiều công cụ xử lý. Phần này chỉ trình bày tóm lược một vài công cụ xử lý được cung cấp dưới dạng mã nguồn mở. Chẳng hạn, Point Cloud Library (PCL) - một công cụ độc lập chuyên cho việc xử lý đám mây điểm (point cloud) với các chức năng như: đăng ký (registration), gom cụm (clustering), phân đoạn (segmentation), tạo mô hình (reconstruction), trực quan hóa (visualization) [4]. Tuy nhiên PCL chưa cung cấp nhiều chức năng trong việc xử lý mô hình 3D dạng lưới (mesh). ParaView (<https://www.paraview.org/download/>) được phát triển từ năm 2000, dựa trên nền tảng thư viện mã nguồn mở Visualization Toolkit (VTK).

ParaView được ứng dụng trong phân tích và trực quan hóa dữ liệu dựa trên kỹ thuật định tính và định lượng. Công cụ này có thể chạy trên máy tính cá nhân hoặc siêu máy tính (supercomputer). Công cụ được đề cập chính trong bài báo này là MeshLab, một công cụ chuyên cho việc xử lý và chỉnh sửa mô hình 3D dạng lưới. MeshLab được phát triển bởi Visual Computing Lab - ISTI-CNR từ năm 2005 và đã được trao giải thưởng Phần mềm Eurographics danh giá vào năm 2017. Đến thời điểm hiện tại, công cụ này đã đạt được 3,000,000 lượt tải với hơn 100,000 người dùng đăng ký, hơn 1,600 trích dẫn. MeshLab dựa trên nền tảng VCGLIB (cũng do Visual Computing Lab phát triển) và sử dụng Qt để cài đặt phần giao diện người dùng.

2.2. Một vài plugin bổ sung

MeshLab hỗ trợ đọc/xuất (input/output) hầu hết các định dạng mô hình 3D phổ biến như: “obj”, “ply”, “stl”, “off”, “3ds”, “wrl”, “pdb”. Công cụ này cung cấp rất nhiều tính năng cho việc xử lý mô hình 3D dạng lưới, như: đăng ký, tái tạo, ánh xạ màu và kết cấu (color mapping and texture), so sánh (compare), tinh chỉnh và cấu trúc lại mô hình (refine and remeshing), trực quan hóa, cùng nhiều tính năng khác [5]. Mặc dù đã hỗ trợ rất nhiều chức năng liên quan cho việc xử lý một mô hình 3D, nhưng một vài tính năng vẫn chưa được tối ưu cũng như chưa có sẵn trong MeshLab.

Phần tiếp theo sẽ trình bày kiến trúc cơ bản của MeshLab và các tính năng bổ sung. Cụ thể, tính năng đọc tập tin “mhd” và “vtk”, được sử dụng rất phổ biến cho việc lưu trữ hình ảnh y khoa như CT, MRI, tính năng tiền xử lý và tham số hóa mô hình trong không gian cầu 3D.

2.2.1. Kiến trúc cơ bản của một plugin trong MeshLab

MeshLab plugin là một tập tin thư viện (*.so trong Linux và MacOS hay *.dll trong Windows). Mỗi plugin cung cấp một tính năng cụ thể, độc lập với kết xuất đồ họa và giao diện người dùng, ngoại trừ một số loại plugin đặc biệt. Bao gồm các loại sau:

Filterplugin: đây là plugin phổ biến nhất và dễ xây dựng. Cho phép tạo và xử lý một hay nhiều lưới mô hình trong MeshLab.

IOplugin: hỗ trợ MeshLab khả năng xử lý một định dạng tập tin mô hình.

Editplugin: cung cấp các công cụ đồ họa giúp căn chỉnh lưới mô hình.

Decorateplugin: cung cấp các tính năng trang trí mô hình.

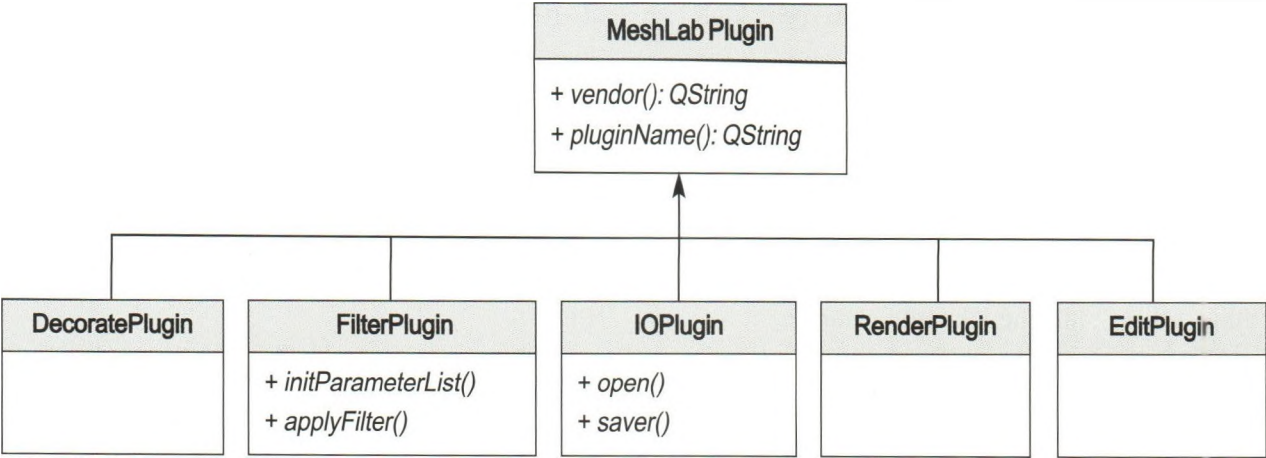
Renderplugin: cho phép tùy biến tiến trình kết xuất đồ họa của lưới mô hình.

Một MeshLab plugin cơ bản là một lớp Qt Plugin. Để tạo một plugin mới, người dùng cần định nghĩa một lớp mới thực thi một trong các lớp ảo sau: “DecoratePlugin”, “FilterPlugin”, “IOPlugin”, “RenderPlugin”, “EditPlugin” và cài đặt những hàm ảo bắt buộc, chẳng hạn, phương thức “open” và “save” với “IOPlugin” và “initParameterList” và “applyFilter” với “FilterPlugin”, như trình bày trong Hình 1.

2.2.2. Plugin đọc tập tin mhd và vtk: io_vtk

Một dữ liệu hình ảnh định dạng “mhd” được lưu trữ thành 2 tập tin riêng biệt, mhd và raw. Trong đó, tập tin “mhd” chứa thông tin của hình ảnh như mô tả trong Bảng 1, tập tin định dạng “raw” chứa dữ liệu. Cụ thể, dữ liệu chứa trong tập tin “raw” là các giá trị vô hướng, Hounsfield (HU), là một thang đo định tính mô tả mật độ bức xạ. Mỗi cơ quan trong cơ thể sẽ tương ứng với một miền giá trị của HU tương đối. Giá trị đặc trưng cho các bộ phận mềm thay đổi trong khoảng -700 đến +255, trong khi của xương là

Hình 1: Sơ đồ kiến trúc cơ bản của plugin trong MeshLab



Bảng 1. Một số thuộc tính chính trong tập tin “mhd”

Thuộc tính	NDims	ElementSpacing	DimSize	ElementType	ElementDataFile
Ý nghĩa	Số chiều của ảnh	Khoảng cách các điểm ảnh	Độ phân giải của ảnh	Kiểu dữ liệu được lưu trữ trong tập tin “raw”	Đường dẫn đến tập tin “raw”

từ +200 đến +3000. Dựa vào đặc tính này, các nhà nghiên cứu đã có thể phân đoạn (segment) các bộ phận để mô hình hóa riêng biệt.

Hình ảnh định dạng “mhd” có thể được tạo và đọc bởi thư viện VTK. Cho trước một ảnh dạng “mhd”, cần định nghĩa một ngưỡng để trích xuất các vùng tương ứng với các cơ quan. Một thuật toán phổ biến được sử dụng để trích xuất một lưới đa từ dữ liệu vô hướng rời rạc (discrete scalar field) là Marching Cubes [6], được cài đặt trong thư viện VTK, thông qua lớp “vtkMarchingCubes”. Cuối cùng, dữ liệu thu được cần chuyển sang cấu trúc dữ liệu của MeshLab để có thể trình bày trên giao diện.

2.2.3. Tiền xử lý và tham số hóa: filter_fix_param

Mô hình 3D cần được xử lý để thu được những thông tin hữu ích. Mô hình 3D với độ phân giải cao cho phép đối tượng được thể hiện chi tiết, song điều này dẫn đến tốc độ xử lý chậm và tốn tài nguyên. Một số mô hình có thể chứa các đỉnh (vertice) hoặc các mặt (face) trùng lặp (duplicate), cũng như những lỗ trống (hole), điều này có thể dẫn đến lỗi trong quá trình xử lý, chẳng hạn trong các yêu cầu có tham chiếu đến diện tích mỗi mặt trong bài toán

làm mịn (smooth) hoặc bài toán tham số hóa mô hình (parameterization).

Attene [7] đã đề xuất một tập các phương pháp cho việc xử lý mô hình, với các thao tác cơ bản như: đơn giản hóa (simplification), xóa bỏ đỉnh và mặt trùng lặp, làm mịn,... Tác giả đã cung cấp mã nguồn bằng ngôn ngữ C/C++ độc lập dưới dạng dòng lệnh, tuy nhiên chỉ giới hạn cho việc xử lý một vài định dạng. Do đó, việc liên kết các tính năng này vào MeshLab vừa giúp thao tác được nhiều định dạng tập tin, vừa đảm bảo tính liên tục và hiệu quả trong quá trình xử lý. Sau khi trải qua thao tác tiền xử lý, mô hình 3D có thể được áp dụng với các thuật toán theo yêu cầu cụ thể, như tính độ cong (curvature), hoặc tham số hóa. Bài toán tính độ cong đã được cài đặt hiệu quả trong MeshLab, trong khi tính năng tham số hóa mô hình vào không gian hình cầu chưa có sẵn.

Việc tham số hóa mô hình rất hữu ích trong việc phân tích bề mặt đối tượng. Tham số hóa định nghĩa như việc xác định một ánh xạ (mapping) 1:1 từ hệ tọa độ Euclid vào hệ tọa độ đơn giản hơn, như hệ tọa độ cầu. [8] và [9] đã sử dụng tham số hóa của vỏ não để phân tích các rãnh trên não người. Trong bài báo này, chúng tôi sử dụng thuật toán đề xuất

bởi [8] để xây dựng plugin có tính năng ánh xạ một mô hình 3D vào không gian hình cầu có bán kính 1.

3. Kết quả và thảo luận

Chúng tôi đã biên dịch MeshLab phiên bản 2020.02 và các plugin bổ sung với hệ điều hành Ubuntu 20.04 LTS, máy tính cá nhân. Các tính năng bổ sung được thể hiện dưới dạng thư viện liên kết động (“.so”), gồm “libfilter_fix_param.so” và “libio_vtk.so”, như thể hiện trong Hình 2.

Để minh họa các chức năng được bổ sung, chúng sử dụng một số hình ảnh từ cơ sở dữ liệu ảnh CT của gan, MICCAI 2007 [10]. Dữ liệu từ MICCAI 2007 gồm 20 ảnh CT vùng bụng, và 20 ảnh CT đã được phân đoạn vùng gan. Giá trị chứa

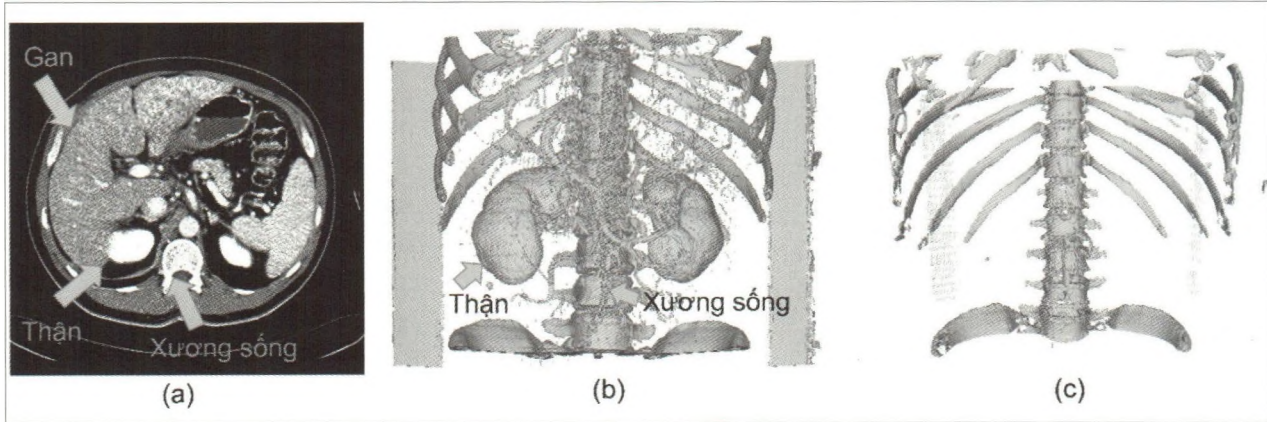
trong tập tin dữ liệu (raw) của tập thứ nhất dao động từ -1000 đến +1000, của tập thứ hai là từ 0 đến 1, trong đó giá trị 0 thể hiện vùng nền, giá trị 1 thể hiện vùng gan. Tất cả hình ảnh được lưu trữ dạng “mhd”. Đối với ảnh CT vùng bụng, ngưỡng được minh họa với giá trị 160 và 300, với ảnh CT vùng gan, ngưỡng được thiết lập với giá trị 0.5. Kết quả đọc tập tin định dạng “mhd” được thể hiện trong Hình 3.

Như thể hiện trong Hình 3, giá trị ngưỡng càng thấp thì mô hình càng thể hiện được nhiều chi tiết của các cơ quan (mạch máu và thận), với giá trị ngưỡng cao (300) thì chỉ một vài bộ phận được thể hiện (xương). Đối với ảnh đã được phân đoạn

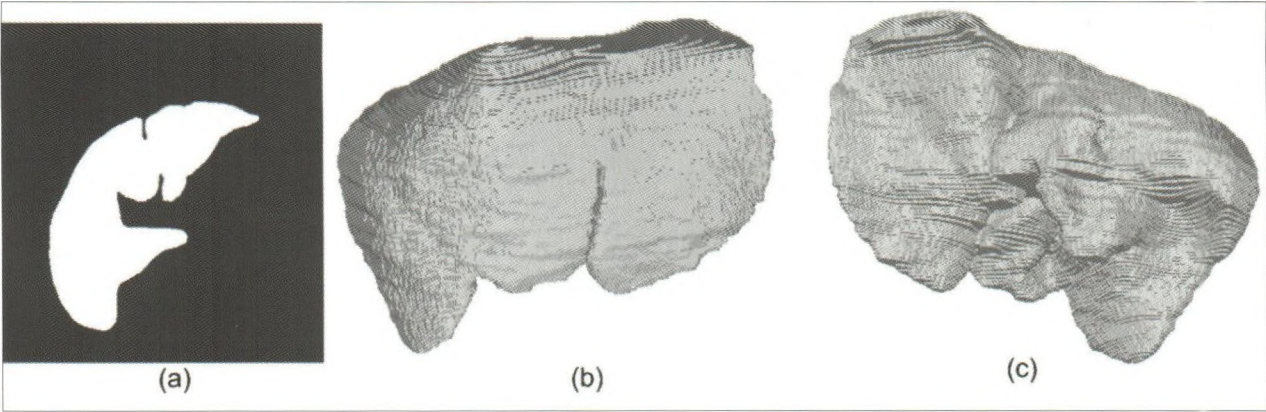
Hình 2: Các plugin mới được tích hợp vào trong MeshLab

Plugin Information					
Number of plugin loaded: 77					
Plugins	Enabled	Type	Vendor	File	Uninstall
FilterColorP...	☒	Filter	CNR-ISTI VCLab	libfilter_colorproc.so	X
FilterCreate	☒	Filter	CNR-ISTI VCLab	libfilter_create.so	X
FilterCreatel...	☒	Filter	CNR-ISTI VCLab	libfilter_createiso.so	X
FilterCSG	☒	Filter	CNR-ISTI VCLab	libfilter_csg.so	X
FilterDirt	☒	Filter	CNR-ISTI VCLab	libfilter_dirt.so	X
Fix_Param	☒	Filter	LCD,HCTC AGU	libfilter_fix_param.so	X
FilterFractal	☒	Filter	CNR-ISTI VCLab	libfilter_fractal.so	X
IOU3D	☒	IO	CNR-ISTI VCLab	libio_u3d.so	X
IOVTK	☒	IO	LCD,HCTC AGU	libio_vtk.so	X
IOX3D	☒	IO	CNR-ISTI VCLab	libio_x3d.so	X

Hình 3: Kết quả đọc ảnh gốc (a) với các giá trị ngưỡng 160 (b) mất 5.4 giây và 300 (c) mất 3.0 giây



Hình 4: Kết quả đọc ảnh đã phân đoạn (a), dạng mhd được thể hiện ở mặt trước (b) và mặt sau (c)



(segment), giá trị ngưỡng thường được thiết lập để tách biệt giữa vùng nền và vùng chứa đối tượng, như được thể hiện trong Hình 4.

Mô hình 3D sau khi được đọc bởi MeshLab cần qua thao tác tiền xử lý. Chúng tôi đã thực sự so sánh kết quả giữa việc sử dụng chức năng hiện có trên MeshLab với plugin mới cài đặt dựa vào phương pháp của Attene [7]. Mô hình ban đầu chứa 210,124 đỉnh và 420,248 mặt. Kết quả được thể hiện như trong Hình 5.

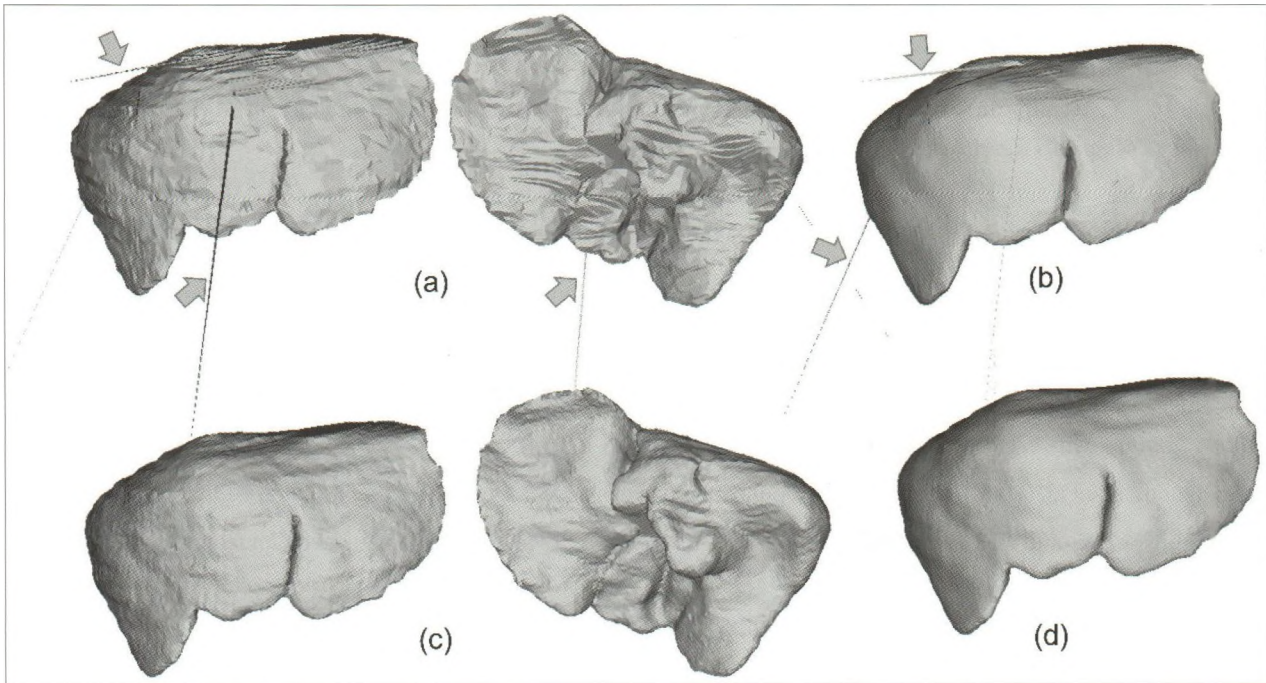
Dòng thứ nhất sử dụng chức năng có sẵn của MeshLab, ở mặt trước và sau (a) (b) và làm mịn

(c). Dòng thứ hai sử dụng plugin bổ sung, và làm mịn. Mô hình đơn giản hóa chứa 5,000 đỉnh và 10,000 mặt

Có thể thấy, việc sử dụng chức năng có sẵn có thể cho ra các kết quả không như mong muốn, mô hình chứa nhiều mặt có cạnh dài vô tận. Điều này gây khó khăn trong việc trực quan hóa và phân tích, in ấn mô hình.

Sau khi được xử lý đơn giản hóa và làm mịn, mô hình được tham số hóa vào trong không gian hình cầu. Để thấy rõ sự tương ứng giữa cấu trúc bề mặt mô hình và bề mặt được tham số hóa, chúng tôi sử

Hình 5: So sánh giữa chức năng của MeshLab và plugin bổ sung



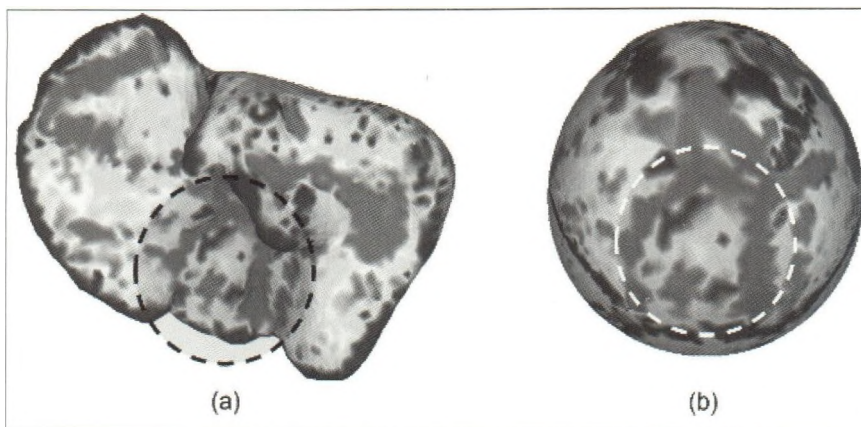
dụng chức năng tính độ cong trung bình trên bề mặt trước và mô phỏng bằng hệ màu RGB, sau đó ánh xạ mô hình sang không gian hình cầu đơn vị, như thể hiện trong Hình 6.

Tính chất rất quan trọng của ánh xạ chính là bảo toàn các góc (angle preserving) của đa giác (tam giác), do đó hạn chế độ biến dạng của bề mặt. Có thể thấy trong Hình 6, hình dạng của các rãnh trên gan được bảo toàn.

4. Kết luận

Mô hình 3D cung cấp thông tin chi tiết và đầy đủ về đối tượng, do đó việc xử lý và chỉnh sửa mô hình là điều cần thiết do sự đa dạng hình ảnh đầu vào. Nhiều công cụ đã được thiết kế với nhiều thuật toán xử lý. MeshLab được xem là một trong những công cụ mã nguồn mở được sử dụng rất phổ biến và nhận được nhiều sự quan tâm từ cộng đồng nghiên cứu, do sự đơn giản, hiệu quả. Tuy nhiên, một vài chức năng trên MeshLab vẫn chưa hoàn thiện, hoặc chưa

Hình 6: Tham số hóa của mô hình (a) vào không gian hình cầu (b)



được hỗ trợ. Bài báo này trình bày cách để tạo mới các plugin trên MeshLab, bao gồm tính năng đọc dữ liệu y tế định dạng “mhd”, tiền xử lý và tham số hóa mô hình vào không gian hình cầu. Các thực nghiệm trực quan cho thấy, các tính năng mới đã góp phần khắc phục và bổ sung, làm cho việc xử lý mô hình tiện lợi và hiệu quả hơn. Trong thời gian tới, chúng tôi sẽ bổ sung thêm nhiều chức năng khác, chẳng hạn, cài đặt thuật toán FLASH [9], nhanh hơn 5.000 lần so với thuật toán đề xuất trong [8], và những chức năng theo nhu cầu cụ thể ■

TÀI LIỆU THAM KHẢO:

1. Li, M., X. Huang, and T. Zheng. (2011). Application of 3D Animation Technology in Movie Art Design. *Advanced Materials Research*, 267, 164-169.
2. Boeykens, S. and H. Neuckermans. (2009). Architectural design analysis, historical reconstruction and structured archival using 3D Models. [Online] Available at https://www.researchgate.net/publication/258241690_Architectural_design_analysis_historical_reconstruction_and_structured_archival_using_3D_Models
3. Pujol, S., et al. (2016). Using 3D Modeling Techniques to Enhance Teaching of Difficult Anatomical Concepts. *Academic radiology*, 23(4), 507-516.
4. Rusu, R.B. and S. Cousins (2011). 3D is here: Point Cloud Library (PCL). [Online] Available at https://pointclouds.org/assets/pdf/pcl_icra2011.pdf
5. Cignoni, P., et al. (2008). MeshLab: an Open-Source Mesh Processing Tool. *Computing*, 1, 129-136.
6. Lorensen, W. and H. Cline (1987). Marching Cubes: A High Resolution 3D Surface Construction Algorithm. *ACM SIGGRAPH Computer Graphics*, 21, 163.
7. Attene, M. (2010). A lightweight approach to repairing digitized polygon meshes. *The Visual Computer*, 26, 1393-1406.

8. Xianfeng, G., et al. (2004). Genus zero surface conformal mapping and its application to brain surface mapping. *IEEE Transactions on Medical Imaging*, 23(8), 949-958.
9. Lam, K.C., et al. (2014). FLASH: Fast Landmark Aligned Spherical Harmonic Parameterization for Genus-0 Closed Brain Surfaces. *SIAM Journal on Imaging Sciences*, 8(1), 67-94.
10. Heimann, T., et al. (2009). Comparison and Evaluation of Methods for Liver Segmentation From CT Datasets. *IEEE transactions on medical imaging*, 28, 1251-65.

Ngày nhận bài: 12/5/2022

Ngày phản biện đánh giá và sửa chữa: 26/5/2022

Ngày chấp nhận đăng bài: 19/6/2022

Thông tin tác giả:

1. ThS. HUỖNH CAO THẾ CƯỜNG

2. TS. LÊ CÔNG ĐOÀN

Trường Đại học An Giang, Đại học Quốc gia Hồ Chí Minh

MESHLAB: A 3D MESH PROCESSING AND ITS ADDITIONAL PLUGINS

● Master. **HUỖNH CAO THẾ CƯỜNG**¹

● PhD. **LE CONG DOAN**¹

¹An Giang University

Vietnam National University - Ho Chi Minh City

ABSTRACT:

3D model has been widely used in many fields. Therefore, 3D models should be pre-processed and edited to obtain the useful information. Several 3D model processing tools have been employed with functions of presenting, filtering, and exporting data. Besides available functions of these tools, it is crucial to develop additional plugins for processing specified tasks. This paper introduces MeshLab's functions and additional plugins for importing the MetaImage, preprocessing, mesh parameterization. The experiments shows that the proposed plugins have been efficiently implemented and improved the current functions of MeshLab.

Keywords: MeshLab, 3D model, plugin.