

DICOM 3D数字化建模

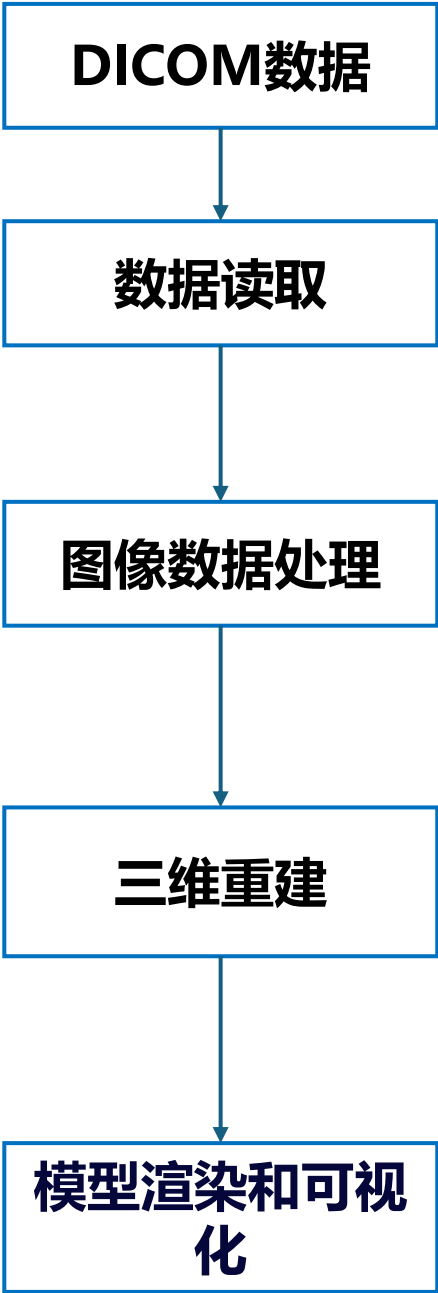
2024年3月

方案概述

项目需求：

将DICOM数据，进行3D数字化建模，实现在PC上通过鼠标点击查看，包括旋转、放大、缩小等交互，将功能封装制作成具有便捷性的软件可以随时使用。

实现方案：



将DICOM格式的CT数据根据DICOM标准数据格式协议进行识别、读取，开发读取算法，可批量读取DICOM系列文件。

研发滤波算法，将DICOM图像中的伪影去除，再使用图像识别和计算机视觉CV技术，将CT图像中的不同组织结构或对象边界分离开来。自动分割则主要利用阈值分割、区域生长、边缘检测等算法来识别并分离结构。

将分割结果数据，转换为三维模型：使用VTK工具库以及计算机视觉、三维重建算法，将处理完毕的DICOM数据图像进行关联、计算，构建出三维立体的模型数据。

将三维重建的模型数据在软件界面上进行渲染，渲染后的模型可以通过三维可视化工具进行查看和分析，可以自由旋转、缩放查看模型的细节。

1. DICOM数据读取

DICOM格式是国际通用的CT影响数据格式，具有通用的格式标准和编码标准，从<https://www.dicomstandard.org/>官网我们可以查看标准的编码格式从而进行数据解析和读取。DICOM数据在电脑上的格式为.dcm后缀的文件，一个CT扫描结果可能具有多个文件，我们需要识别并从单幅多帧的dcm文件中读取数据，如下为DICOM数据文件：

★	📄	I1800001	2019/8/15 9:43	文件	532 KB
★	📄	I1810000	2019/8/15 9:43	文件	532 KB
★	📄	I1820000	2019/8/15 9:43	文件	532 KB
★	📄	I1830000	2019/8/15 9:43	文件	532 KB
	📄	I1840000	2019/8/15 9:43	文件	532 KB
	📄	I1850000	2019/8/15 9:43	文件	532 KB
	📄	I1860000	2019/8/15 9:43	文件	532 KB
	📄	I1870000	2019/8/15 9:43	文件	532 KB
	📄	I1880000	2019/8/15 9:43	文件	532 KB
	📄	I1890000	2019/8/15 9:43	文件	532 KB
	📄	I1900000	2019/8/15 9:43	文件	532 KB
	📄	I1900001	2019/8/15 9:43	文件	532 KB
	📄	I1910000	2019/8/15 9:43	文件	532 KB
	📄	I1920000	2019/8/15 9:43	文件	532 KB
	📄	I1930000	2019/8/15 9:43	文件	532 KB
	📄	I1940000	2019/8/15 9:43	文件	532 KB
	📄	I1950000	2019/8/15 9:43	文件	532 KB
	📄	I1960000	2019/8/15 9:43	文件	532 KB
	📄	I1970000	2019/8/15 9:43	文件	532 KB
	📄	I1980000	2019/8/15 9:43	文件	532 KB

我们根据DICOM文件的格式编码可以解析出其数据格式，并采用C++代码编写读取程序，程序开发的逻辑框架如下

- 1.在程序代码中首先建立DcmFileFormat 和DcmObject 类，是处理DICOM 文件格式的类(可以识别头文件标签)
- 2.从.dcm文件中读取DICOM格式的数据，并将其解析为DcmDataset对象，将数据填充在DcmDataset对象的内存缓存中， DcmDataset类可以采用PyDicom开源库来建立，用于表示一个DICOM数据集。
- 3.将DcmDataset存储的CT数据集，采用算法进行分类并转成图像数据格式，建立DicomImage类用于读取和处理 DICOM 图像文件，可以处理图像的具体数据比如像素值、像素尺寸、像素类型等。

2. 图像数据处理

在上一步解析完成DICOM数据并转换成图像数据的基础上，进行图像处理，目的是处理成可用于3D模型渲染的数据格式：

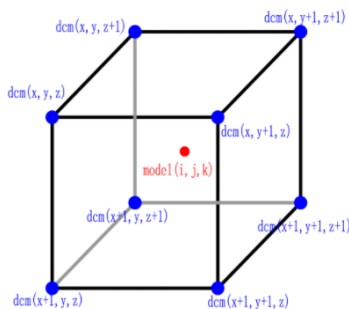


图像数据处理的程序开发逻辑如下：

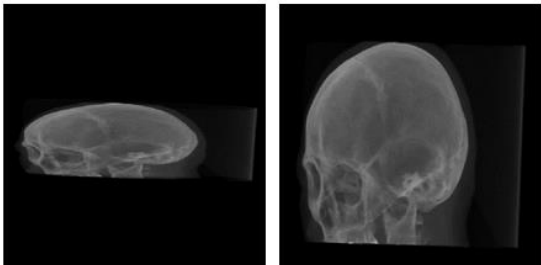
- 1.采用OpenCV计算机视觉库，对图像进行过滤，将图像中可以进行3D重建的部分进行识别，一般利用阈值分割、区域生长、边缘检测等算法来识别并分离结构。
- 2.将图像中不用于3D重建的部分进行数据剔除，并重新生成只保留关键数据的图像数据。
- 3.将具有关键部分数据的图像中的噪点等进行去除、过滤，包括去噪、平滑、分割、对齐等操作，以保证后续算法的结果准确性，并将图像尺寸统一、数据格式统一，生成可以进行三维重建的图像数据。

3.三维重建

将预处理后的图像数据转化为三维网格模型，并通过边界表示法（Boundary Representation, B-Rep）等方式对其进行表面重建



$$\begin{aligned} model(i, j, k) = & (1-a) \cdot (1-b) \cdot (1-c) \cdot dcm(x, y, z) + (1-a) \cdot (1-b) \cdot c \cdot dcm(x, y, z+1) \\ & + a \cdot (1-b) \cdot (1-c) \cdot dcm(x+1, y, z) + a \cdot (1-b) \cdot c \cdot dcm(x+1, y, z+1) \\ & + (1-a) \cdot b \cdot (1-c) \cdot dcm(x, y+1, z) + (1-a) \cdot b \cdot c \cdot dcm(x, y+1, z+1) \\ & + a \cdot b \cdot (1-c) \cdot dcm(x+1, y+1, z) + a \cdot b \cdot c \cdot dcm(x+1, y+1, z+1) \end{aligned}$$



(a) 未重采样 (b) 重采样后

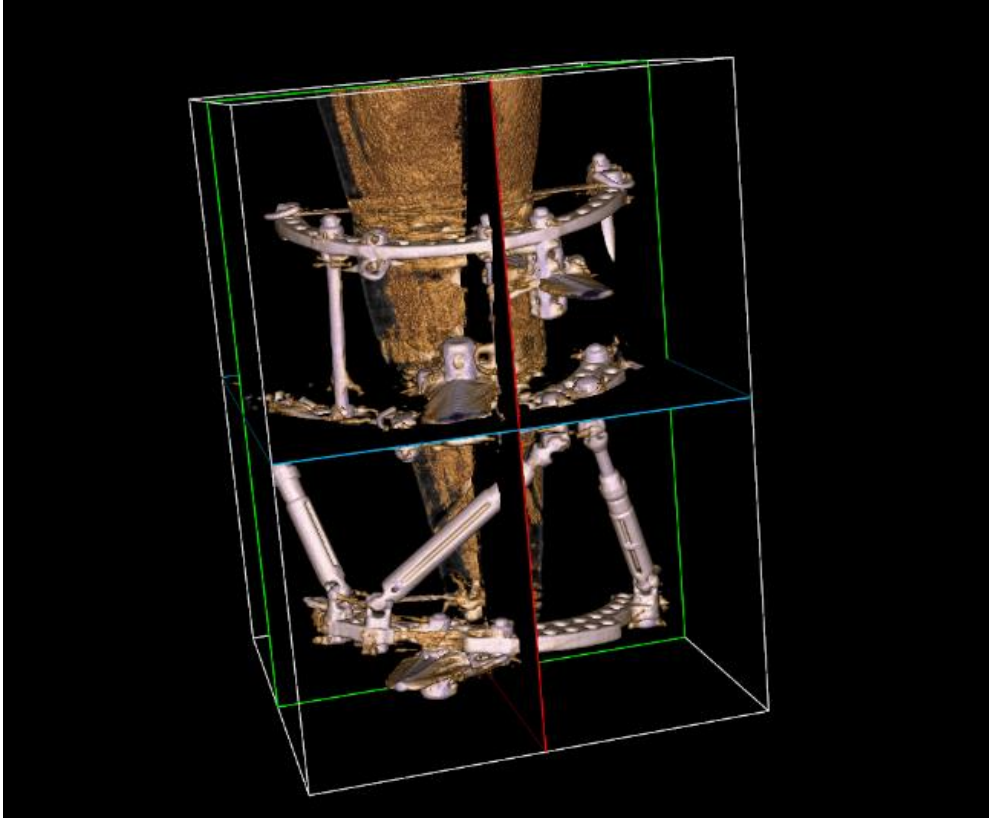
我们一般采用以下两种算法，图像数据进行网格化转换和表面重建：

- **Marching Cubes算法：**将医学影像数据离散化为一个个立方体，通过分析各个顶点的灰度值和位置信息，判断内部和外部的关系，从而得出每个立方体的表面信息。
- **Kinetic Tomography算法：**该算法利用计算机自动处理医学数据的技术，通过分析两张以上切面图的互相关信息，计算出三维形态信息。

在重建算法计算后，需要对重建数据进行优化：
模型优化：对重建的三维模型进行优化操作，以更好地满足医学应用的需求，如提高细节分辨率、去除噪声、保证拓扑关系等。
数据后处理：对优化后的模型进行后处理，包括纹理映射、颜色处理、渲染等操作，以得到最终的可视化结果。

4.模型渲染和可视化

根据上一步算法得到3D数据格式，最后我们可以将3D模型在软件系统中进行渲染和可视化操作：



渲染和可视化的开发逻辑：

- 1.首先将三维重建的模型数据输入VTK库、OpenGL库以及OpenCV库等三维渲染和处理库中，对抽取出的三维模型进行剔除、平滑、法向量计算等处理。
- 2.处理后模型映射到多边形数据结构中，设置模型的颜色等属性，将多边形数据结构传递给渲染器并在窗口中显示出三维模型。
- 3.将3D模型加载到自建的渲染引擎或UE5引擎中，利用引擎的窗口交互器，开发三维模型的旋转、缩放、平移等交互操作功能，使用户能够通过鼠标、键盘查看和操作3D模型。
- 4.将整套程序、算法，打包成适用于Windows电脑的客户端软件EXE程序，可以快速安装和便捷使用，日常使用查看DICOM数据文件的三维影像模型。