

基于 Qt 实现的图片数据标注工具 文档

吴佳龙 2018013418

简介

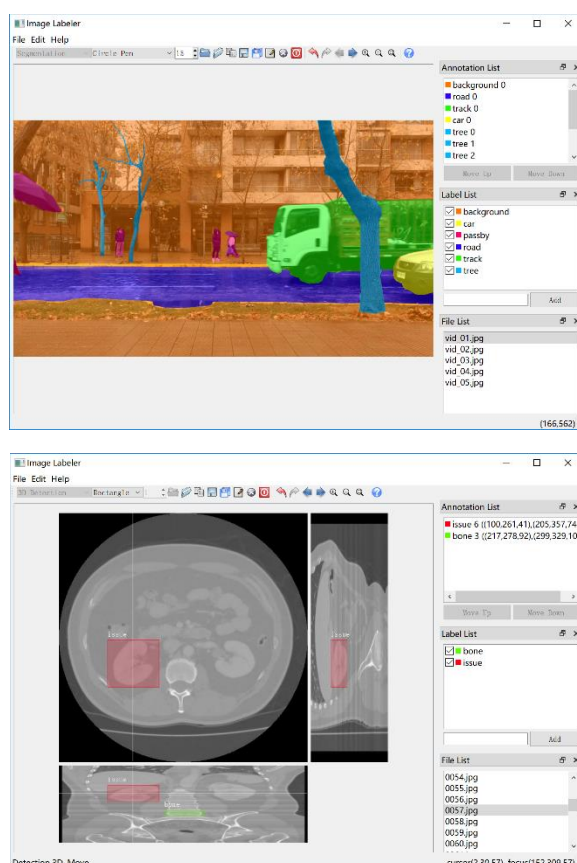
本次作业利用 Qt 开发了一款具备图形界面的图片数据标注工具，可用于人工智能领域的的数据准备，具体地，能够供用户对 2D 和 3D 图片进行以下两种标注：

1、检测标注：标明覆盖图片中某物体的尽可能小的长方形（立方体）方框，并且给方框添加标签。

2、分割标注：标明物体在图片中覆盖的像素（体素），同样给该形状添加一个标签。

程序源代码可在 Qt 5.12.4, Qt Creator 4.9.2 环境下编译运行。

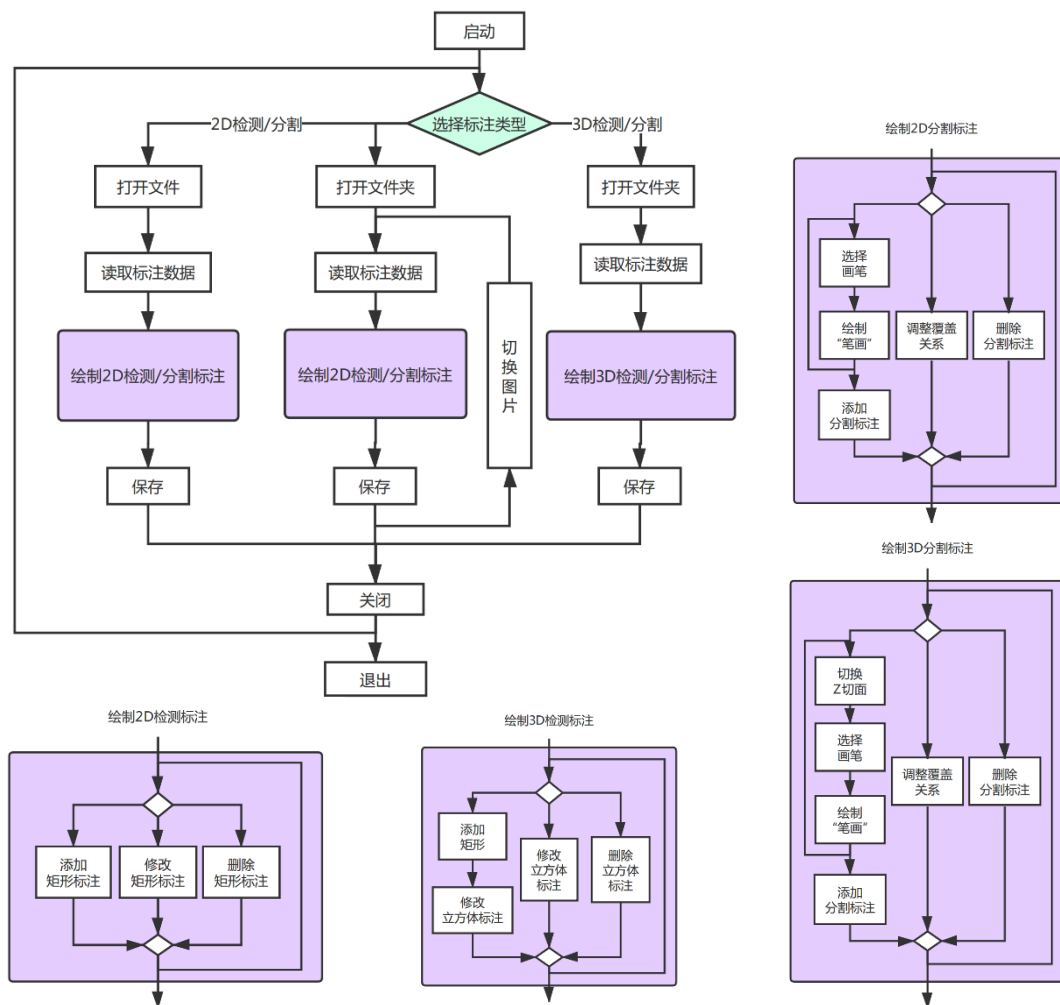
该工具的图形界面如下：



该文档的第二节将简要说明程序运行的主要流程，第三节将详细说明程序的各个功能及示例，第四节将分析各个程序模块的逻辑关系。

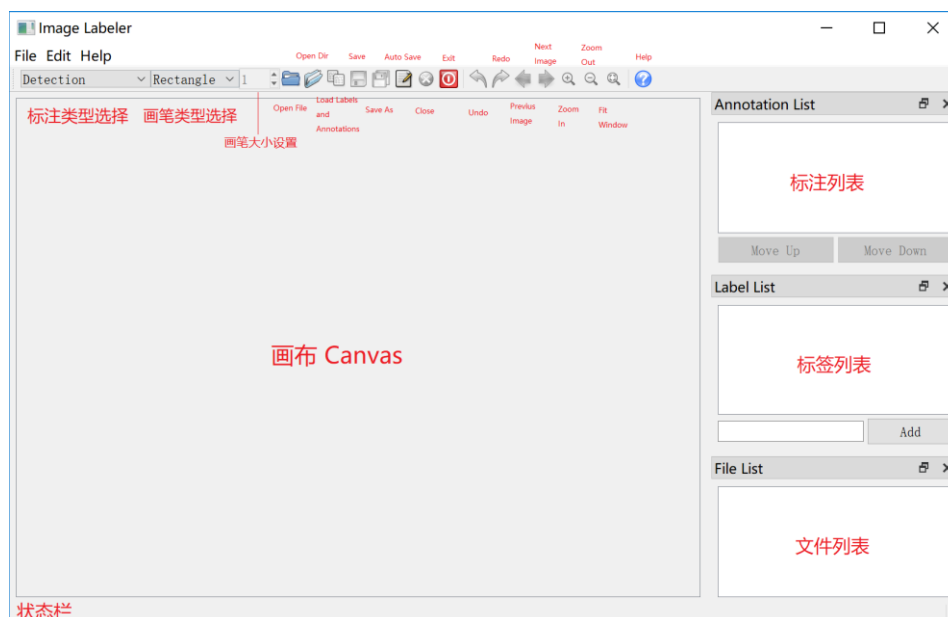
程序运行流程

程序运行流程的示意图如下，具体操作可见下一节（程序功能与示例）。



程序功能与示例

启动程序出现如下界面，主要的部件的名称在下图中标出。



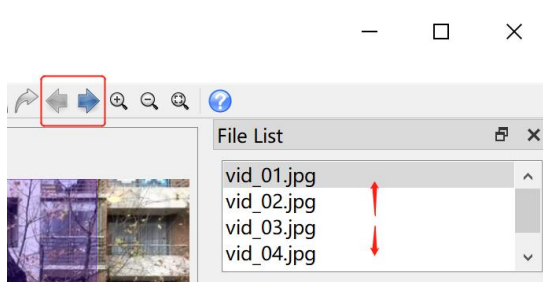
注：无在答辩后额外实现的功能。

1、功能

1) 读入

对于 2D 图片，点击工具栏中的“Open File”按钮或者“File”菜单栏中的“Open File”选项就会弹出对话框供选择一个图片文件打开，支持的文件扩展名为 jpg 或 png，在窗口左半部分将显示被打开的文件；点击图中“Open Dir”按钮或“File”菜单中的相应选项，就会弹出对话框供选择一个文件夹，选中文件夹后会打开该文件夹中的所有图片文件（扩展名为 jpg 或 png），并将它们按照文件名排序显示在文件列表中，第一张图片将显示在窗口中。

文件夹中图片的切换有多种方式：在文件列表中用鼠标点击文件名即可切换到相应文件；点击文件列表后按上下键可切换相邻图片；点击工具栏或菜单中“Previous Image”、“Next Image”或按下它们的快捷键（Ctrl + Up/Down）也可切换相邻图片。



对于 **3D 图片**，在标注类型的选择框内选择 3D 标注（3D 检测或分割）后，点击“Open Dir”选择一个文件夹后，将读入该文件夹中所有图片文件并按照文件名排序显示在列表中，它们被认为是 3D 物体垂直 Z 轴的各个切面。

不论是哪种标注任务下，打开文件后都会自动寻找同文件夹下默认的输出文件是否存在，若存在则将其读入；默认的输出文件名如下表。

2D 单张图片检测标注及标签	\$filename\$_detect_labels_annotations.json
2D 单张图片分割标注及标签	\$filename\$_segment_labels_annotations.json
2D 文件夹中标签文件	labels.json
2D 文件夹中图片检测标注	\$filename\$_detect_annotations.json
2D 文件夹中图片分割标注	\$filename\$_segment_annotations.json
2D 图片分割结果：Color Map	\$filename\$_segment_color.png
2D 图片分割结果：Label Id Map	\$filename\$_segment_labelId.png
3D 文件夹检测标注及标签	detect3d_labels_annotations.json
3D 文件夹分割标注及标签	segment3d_labels_annotations.json

Table 1 默认输出文件名

若希望读入的标注数据不在同一文件夹下或与默认输出文件名不同，可点击工具栏或菜单栏中的“Load Labels and Annotations”手动选取一个由本图片标注工具输出的具有正确格式的 json 文件，并读取其中的标签与标注数据。

2) 输出

可点击工具栏或菜单中的“Save”，将标注数据存在以默认的输出文件名保存在同一文

件夹下；也可点击“Save as”，将自己选择一个路径和文件名将当前图片的标注数据存储在指定的文件中(注：即使在打开文件夹的情况下，另存为也只会另存为当前图片的标注数据，而非整个文件夹的标注数据)。

在切换当前图片、打开新的文件（文件夹）、关闭文件、退出程序之前，会检查是否有未保存的修改，若是则弹出对话框让用户选择保存、丢弃或取消；可选中工具栏或菜单中的“Auto Save”来在上述操作中自动保存未保存的修改，而不会弹出对话框。

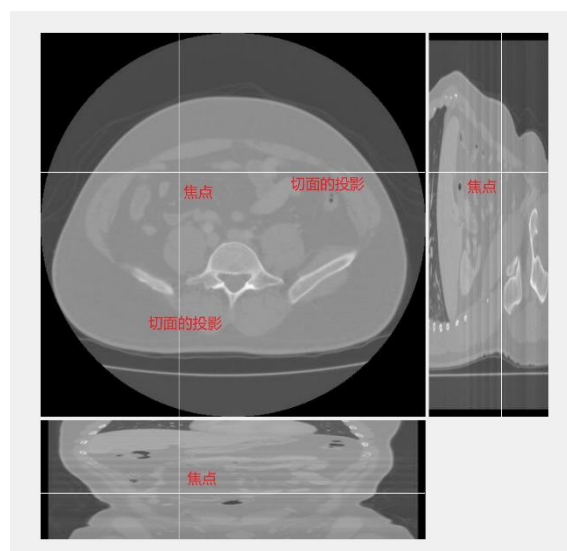
对于分割标注，除了保存绘制标注时的原始数据以外，还额外保存了两张图像作为结果，它们分别是：一张 Color Map，被覆盖的像素的 RGB 值为对应标签的颜色；一张 Label Id Map，被覆盖的像素的灰度值为对应标签的 id。



3) 显示

打开 2D 文件后，在窗口的左半部分会显示当前的图片。

打开 3D 文件后，会显示 3D 物体平行于三个坐标平面的切面。我们来准确地定义接下来文中所说的“切面”：切面是指平行于某一坐标平面（ xOy ， yOz ， zOx ）的某一平面与 3D 物体的交形成的图像。在本图片标注工具中，显示三个切面，它们对应坐标平面以及坐标方向如下图所示。由此定义“焦点”：三个切面的交点坐标定义为“焦点”。可知焦点在三个切面中都显示为一点，而两个切面在另一切面中投影为两条直线，在窗口中显示为两条白色直线，它们的交点即为“焦点”。



读入对应的标注数据后，2D 检测标注将在图片上以矩形方框的形式显示，且矩形左上角用文本标注了对应的标签，矩形的边框和内部将用对应标签的颜色填充，2D 分割标注将在其对应的像素上覆盖上一层具有透明度的标签自身的颜色；3D 检测标注的形式是长方体

方框，但在三个切面上（切面与长方体相交的情况下）都能显示为矩形方框：3D 分割标注对应的所有体素将被着色为标签的颜色，当切面与之有交时将显示出来。

不同标签都有不同的颜色，标注被以相应标签的颜色着色，标签及颜色在标签列表中显示，且颜色可在右键菜单中修改，详见界面说明部分。

显示的图片（包括标注）支持缩放操作，点击工具栏或菜单中的“Zoom in”“Zoom out”可分别将其放大缩小，点击“Fit Window”将调整图像大小刚好与窗口大小适应（即不出现滚动条的最大尺寸）。

4) 操作

以下详细说明进行图片数据标注的操作方法。

在打开文件前，需在标注类型选框先选择需要标注的类型，分别为 2D 检测标注 (Detection)，2D 分割标注 (Segmentation)，3D 检测标注 (3D Detection)，3D 分割标注 (3D Segmentation)。

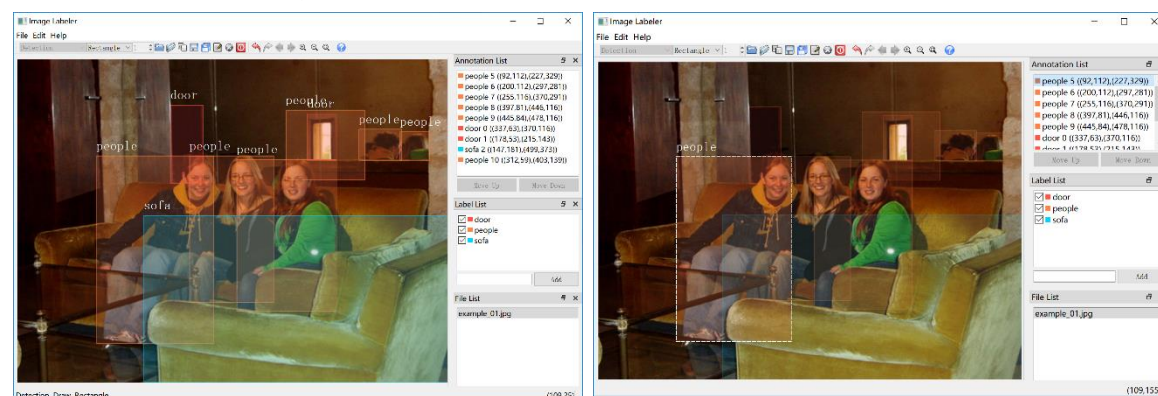
2D 检测标注

对于单个文件，打开后直接进入“绘制模式”，在右下角状态栏中显示鼠标指针当前所在图像像素坐标系中的位置。

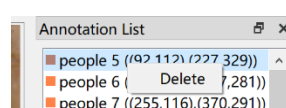
点击图片上某一点，就确定了矩形框的一个顶点，移动鼠标后再点击鼠标，就确定了其对角顶点，两顶点就此确立了一个标注的矩形方框。若误触鼠标左键，按右键即可取消当前的绘制。

此时若标签列表中已经选中了某一个标签（点击即可，参见界面部分说明），则马上生成一个该标签的标注，否则将弹出一个对话框供用户选择一个标签。

在标注列表中选中一个标注（点击即可，参见界面部分说明），则进入“选中模式”，此时可任意拖动选中的标注对应的矩形框的四边；按下 Esc 键或者点击标注列表空白处即可取消选中，返回“绘制模式”。



支持删除标注的操作，在图像上点击右键，包含该点的最近添加的一个标注会被删除；或者在标注列表右键菜单中点击“Delete”即可删除选中标注。



对于文件夹中的多张图片，标注的操作模式与单张图片一致。但是，文件夹中的所有图片共享同一个标签列表，而标注列表则分别独立，各不相同，切换图片时需点击“Save”。

2D 分割标注

对于单个文件，打开后直接进入“绘制模式”，状态栏中同样显示鼠标像素坐标。

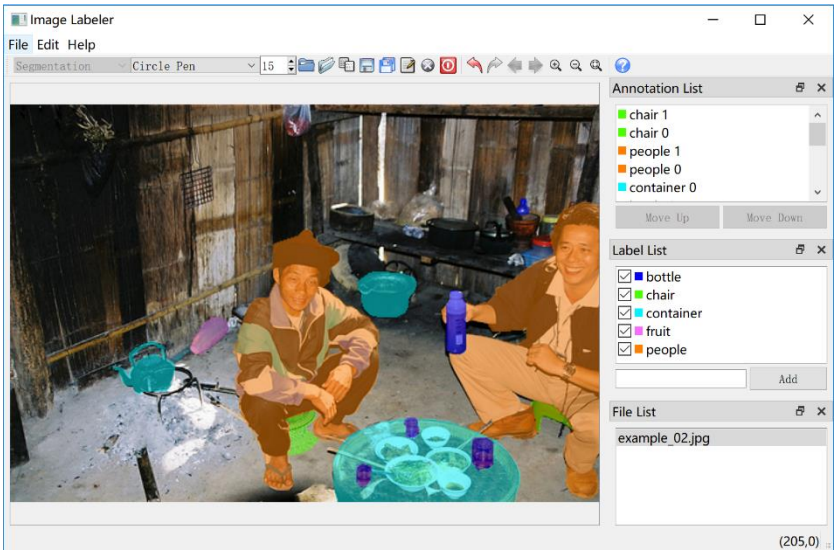
分割标注共有四种“画笔”可选，分别为圆形画笔（Circle Pen），方形画笔（Square Pen），光滑轮廓（Contour），多边形轮廓（Polygonal Contour），使用画笔进行一次绘制后覆盖的像素我们称为一个“笔画”，多个笔画的并集可以成为一个“分割标注”。组成一个分割标注的笔画可以是不同的画笔类型。对于圆形和方形画笔，可在工具栏中选取画笔大小。画笔的具体操作说明如下：

圆形画笔	按下鼠标左键，拖动鼠标后释放左键，圆形画笔覆盖的像素就构成一个“笔画”
方形画笔	操作方式同圆形画笔，画笔形状略有不同
光滑轮廓	按下鼠标左键，拖动鼠标后释放左键，其包围的区域（若起终点不重合则直线连接起终点）中的像素构成一个“笔画”
多边形轮廓	点击鼠标左键，即绘制了多边形的第一个顶点，接下来重复点击左键，依次绘制多边形的顶点，在绘制最后一个多边形的顶点时双击，该多边形包围的像素构成一个“笔画”

Table 2 不同类型的画笔

在绘制笔画的过程中，点击鼠标右键，即可删除上一个绘制的笔画。

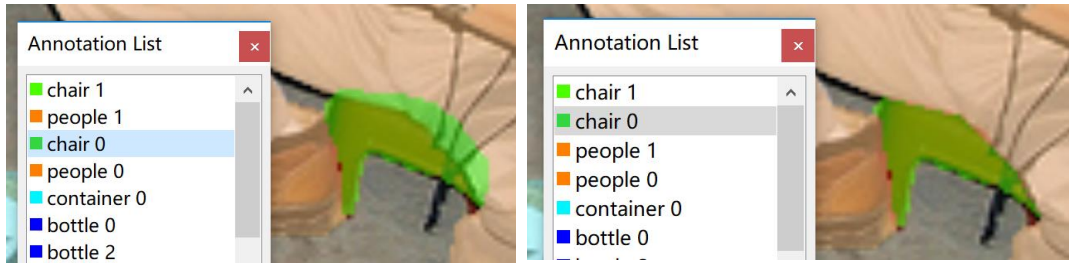
绘制完若干个笔画后，按下回车键，以上笔画合并成一个分割标注，该标注的标签与检测标注以相同的方式确定（列表选中或对话框选择）。



支持删除某个标注，在标注列表中右键菜单中点击“Delete”。

当一个标注的区域覆盖了另一个标注的部分区域时，该区域属于后添加的标注，即后添加的标签会覆盖之前的标签。可以在标注列表中选中某个标注后（即进入“选中模式”），点击“Move Up”和“Move Down”按钮调整标注在列表中位置，调整标注之间的覆盖关系。

对于文件夹中的多张图片，如同检测标注一样，标签列表是共享的，而标注列表是独立的。



3D 检测标注

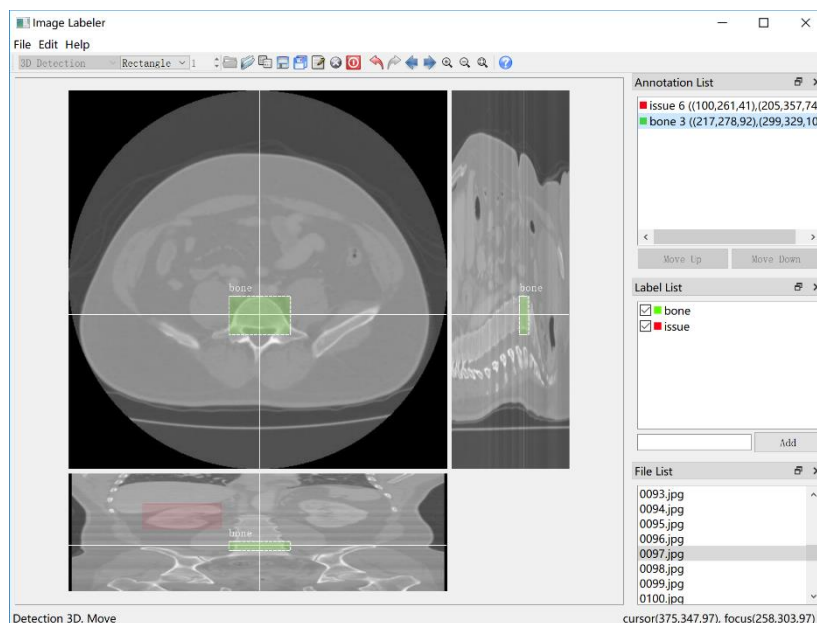
打开文件夹后，直接进入“绘制模式”，状态栏同时显示焦点和鼠标指针的 3D 坐标。

按住 Alt 键进入“移动焦点模式”，在任一个切面上鼠标点击在焦点上即可拖动焦点的坐标，点击在穿过焦点的某一切面的投影（显示为一条白色的直线）上即可拖动改变焦点的某一维坐标。松开 Alt 键退出“移动焦点模式”。

可在任意切面上依照 2D 检测标注的操作方法（两次点击）绘制一个矩形框，该矩形框在该切面的垂直方向上扩展一个默认的厚度即形成了一个立方体标注。

你同样可以在标注列表中选中一个标注，并在任何一个切面上拖动矩形的某一条边进行标注的编辑（实际上是平移立方体的某一个面，该面的平移在其他切面上也会有同步的显示）。

同样支持删除标注的操作。



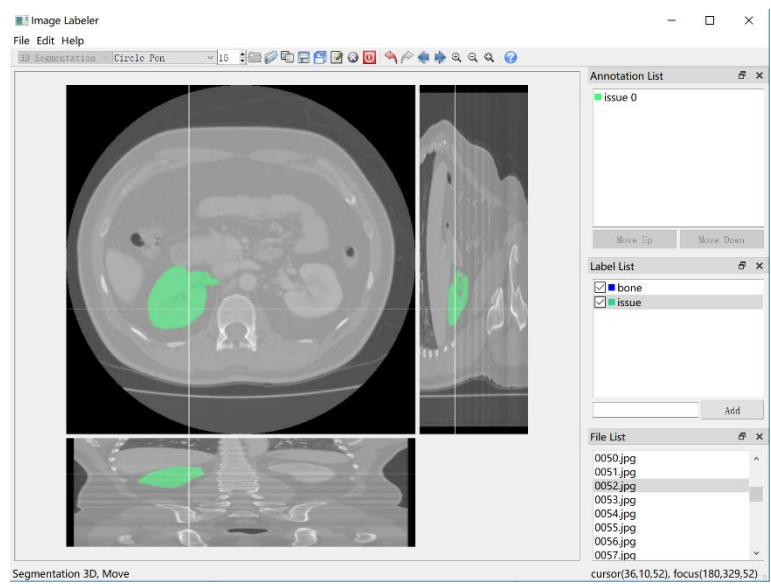
3D 分割标注

打开文件夹后，直接进入“绘制模式”，状态栏同时显示焦点和鼠标指针的 3D 坐标。

按住和松开 Alt 键进入和退出“移动焦点模式”。

3D 的分割标注同样由“笔画”构成，笔画只能绘制在 Z 轴对应的切面上，即中央视图上。3D 分割标注是 2D 分割标注的简单扩展，可看做一层层 2D 分割标注的叠加：一个分割标注的不同笔画可在不同切面上绘制，只需在画完若干笔画后按下回车键即可将当前所有笔画（即使在不同切面）的并集作为一个分割标注。

画笔的操作、当前笔画的撤销、**标注的删除**、Move up/down 等等都与 2D 分割标注一致。



2、界面

1) 菜单栏

菜单栏包括 **File**、**Edit**、**Help** 菜单。菜单中选项如下表：

File	Open File, Open Dir, Load Labels and Annotations, Close, Save, Save As, Auto Save, Exit
Edit	Undo, Redo, Zoom in, Zoom out, Fit Window, Previous Image, Next Image
Help	About

Table 3 菜单栏选项

其中 Undo、Redo 操作仅作用在标注数据的改变上，如标注的添加、编辑、Move up/down、删除等，不包括标签的添加、标签修改颜色、标签删除等与标注数据不直接相关的操作。实际上，在读入与输出数据时，标签与标注的数据是可以分离的。

2) 状态栏

这里的“状态栏”指的是由标注列表、标签列表、文件列表以及显示鼠标坐标的状态栏组成的显示“当前标注状态”的区域。

标注列表的每一项由颜色、标签、id（相同标签的标注用以不同的 id 区分）以及坐标信息（对于检测标注）组成。点击可选中一个标注，点击空白处可取消选中，按上下键可选择相邻的项。Move up/down 按钮只在分割标注、选中模式下可用。**选择某个标注右键可删除该标注。**

标签列表的每一项由 check 框（是否可见）、颜色、标签名组成。点击可选中一个标签，点空白处取消选中，按上下键可选中相邻的项。**选择某个标签右键可修改其颜色或删除该标签。**

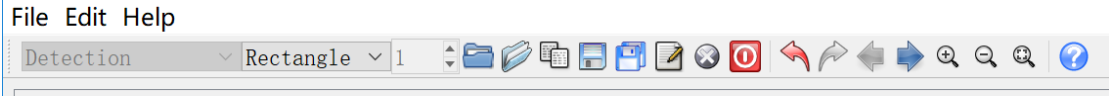
文件列表的每一项包含文件名，点击可切换至对应文件，上下键可切换至相邻文件。

在最下方的状态栏中，每当标注类型或者模式（绘制、选中、移动焦点）或画笔类型改变时，都将显示提示消息，持续 5s；在状态栏右侧会**实时显示鼠标当前所指向的像素坐标**。

3) 工具栏

工具栏中依次为：选择标注类型、选择画笔类型、设置画笔大小、打开文件、打开文件

夹、读取标注数据、保存、另存为、自动保存、关闭、退出、撤销、重做、上一张图片、下一张图片、放大、缩小、适合窗口大小、帮助，与菜单栏中功能完全一致。



3、加分项与创意

可通过闭合折线进行分割，详见上文关于分割标注“光滑轮廓”的说明。
可通过绘制平滑的闭合曲线进行分割，详见上文关于分割标注“多边形轮廓”的说明。
便捷的编辑标注的方法：矩形（立方体）标注可拖动任一条边（任一面）；分割标注可自由调整覆盖关系。
支持快捷键，如下表。（其他利用鼠标与键盘进行的快捷操作都已在上文提及，如按下 Alt 移动焦点）

Open File	Ctrl+O	Open Dir	Ctrl+D	Load Labels and Annotations	Ctrl+L
Save	Ctrl+S	Save As	Ctrl+Shift+S	Close	Ctrl+W
Exit	Ctrl+Q	Undo	Ctrl+Z	Redo	Ctrl+Shift+Z
Previous Image	Ctrl+Left	Next Image	Ctrl+Right		

Table 4 快捷键

程序模块分析

程序各个模块的关系如下图所示，下文将分别简要介绍，具体可参见各自头文件中对各接口的说明。程序的设计遵循显示与控制分离的原则，程序的模块可分为显示与控制两大模块。MainWindow 作为程序的主界面类，在构造之时就将显示与控制两大模块通过 signal-slot 机制连接起来。

1、用户界面与自定义部件

主界面类 MainWindow 的 ui 部分主要由画布（canvas）、标注列表（annoListWidget）、标签列表（labelListWidget）、文件列表（fileListWidget）以及工具栏、菜单栏、状态栏组成。

1) 画布

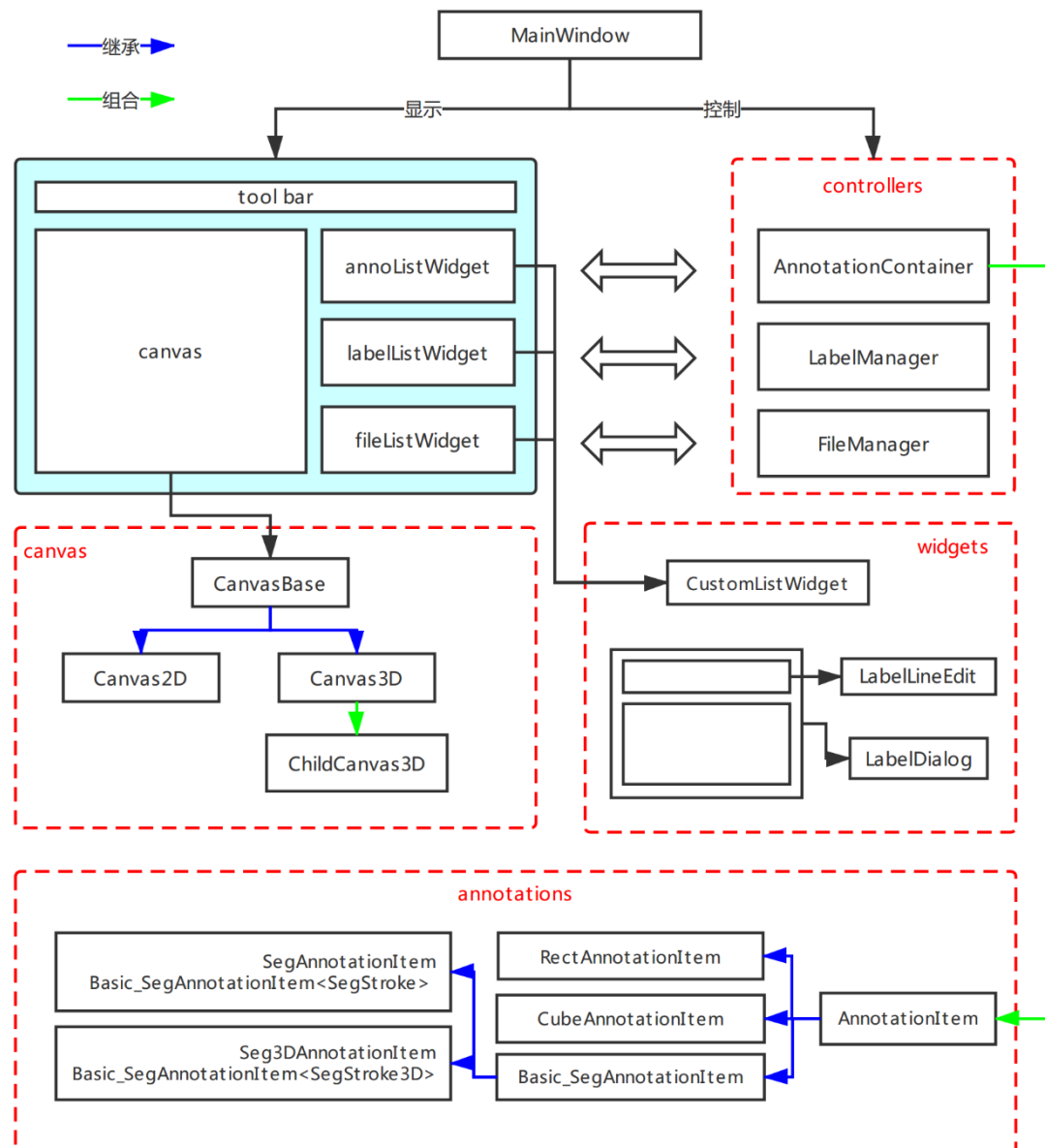
程序中的画布类是一种自定义的部件，用于从 LabelManager， AnnotationContainer 中读取、显示被标注的图片以及其上的标注，并将用户的标注操作通过 signal-slot 机制反馈给 MainWindow 类进行处理，画布类不直接对标注和标签的数据进行修改。

CanvasBase 是两种画布类的基类，继承自 QWidget，定义了一些 2D 画布和 3D 画布公共的接口，便于在 MainWindow 中统一地调用，如更改标注模式、图片缩放、关闭画布等。

Canvas2D 是用于对 2D 图片进行检测和分割标注的画布类，显示单张 2D 图片，用户可在上面绘制矩形方框（检测标注）以及分割标注。

Canvas3D 是用于对 3D 图片进行检测和分割标注的画布类，它由三个 ChildCanvas3D 组成，每个子画布显示沿着三个坐标轴之一将 3D 物体投影所得的图像，用于在子画布上进行的标注操作通过 signal-slot 机制传递给 Canvas3D 汇总并在通过这一机制传递给

MainWindow 处理。



2) CustomListWidget

CustomListWidget 类继承自 **QListWidget**，它作为 **MainWindow** 中的 ui 部件 **annoListWidget**、**labelListWidget**、**fileListWidget** 的类型，针对它们的功能定义了更方便的函数接口以及事件响应。它们负责响应用于存储管理数据的 **AnnotationContainer** 等类的数据变动并在界面中显示出来，同样无法直接访问修改数据。

3) LabelDialog 与 LabelLineEdit

LabelDialog 继承自 **QDialog**，设计了 ui 界面供用户在对话框中选择一种标签。

LabelLineEdit 继承自 **QLineEdit**，是 **LabelDialog** 中的一个部件，可关联一个 **QListWidget** 提供自动补全，点击选择等功能。

2、数据存储与管理类型

主界面类 **MainWindow** 还具有存储和管理标签、标注、文件相关数据的数据类型，分别

为 AnnotationContainer, LabelManager, FileManager。

AnnotationContainer 是用于**存储标注数据的数据结构**，并能够实现撤销和重做的可持久化操作；并且它在 MainWindow 中的实例与 annoListWidget 通过 signal-slot 机制相连接，用户通过列表部件选择了一个标注，则 AnnotationContainer 也设置该标注为选中从而可在画布上对该标注进行修改。可参阅上文中标注相关的具体操作及其示例。

LabelManager 是用于**管理标签的类**，它的实例与 labelListWidget、LabelDialog 都连接起来，能够存储用户添加的标签的信息；并且 labelListWidget 中相应的修改操作（改变标签的颜色、可见性等）都能同步修改相关联的 LabelManager 中的标签信息。可参阅上文中标签相关的具体操作及其示例。

FileManager 是用于**管理文件相关状态的类型**，它的实例的主要功能有：打开文件或文件夹，生成默认的输出文件名（包括标签和标注的输出文件）；记录距离上次保存后是否有未保存的修改；可选中文件，切换显示的图像，并同步 fileListWidget 的选中状态。可参阅上文中文件相关的具体操作及其示例。

3、单个标注与标签的存储类型

存储单个标注的数据类型是 AnnotationItem 及其子类，它们作为 AnnotationContainer 的成员。

存储单个标签的数据类型是 LabelProperty，它们作为 LabelManager 的成员。

参考文献及引用代码出处

- [1] Qt document <https://doc.qt.io/qt-5/index.html>
- [2] Random colors Qt <https://gist.github.com/ialhashim/b39a68cf48a0d2e66621>
- [3] QDarkStyleSheet <https://github.com/ColinDuquesnoy/QDarkStyleSheet>