

数字媒体处理综合实验

姓名： 桑燊

班级： 2014211602

学号： 2014212128

时间： 2017年03月03日

实验目标

实现一个图像浏览器功能：

- 打开和显示常见格式的图像
- 通过拖动和滚轮实现图片的拖拽和缩放

实验思路

首先设计打开和显示常见格式图像的功能，思路如下：

- 使用 `QGraphicsView` 类进行图片的显示
- 使用 `QGraphicsScene` 类作为画布
- 使用 `QGraphicsPixmapItem` 类作为图像对象
- 使用 `QFileDialog` 类用于导入文件

在上一个功能的基础上，设计拖动和滚轮实现图片的拖拽和缩放，设计思路如下：

- 定义一个新类，继承自 `QGraphicsPixmapItem`
- 重写 `mousePressEvent`、`mouseMoveEvent`、`wheelEvent`、`setPos`、`setScale` 等函数
- 将之前显示图片使用的 `QGraphicsPixmapItem` 类替换成新定义的类

实验步骤

1. 创建一个新工程，继承自 `mainwindow`，设计一个简易界面如下：

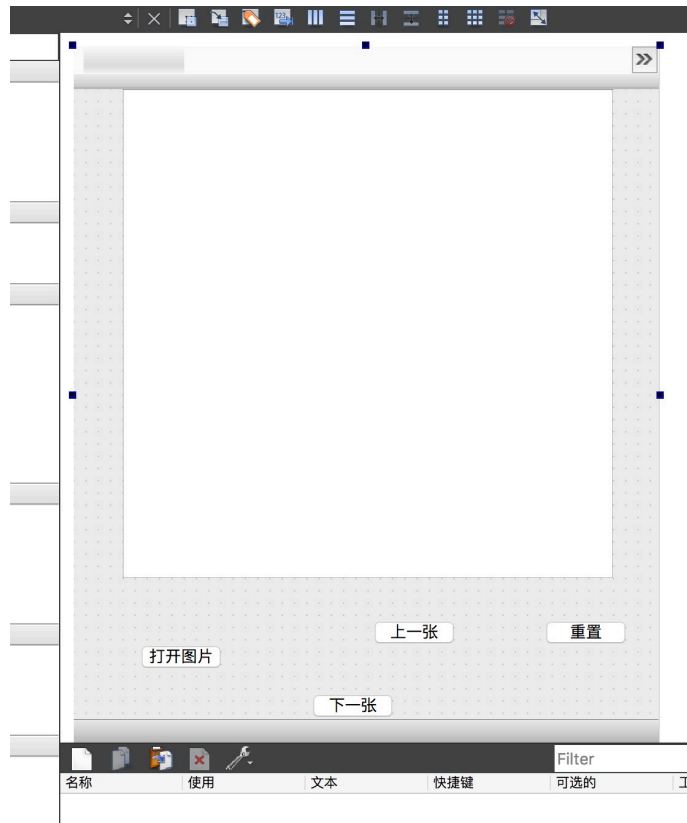


图 1. 界面设计

2. 在头文件中引

入 `QGraphicsView`、`QGraphicsScene`、`QGraphicsPixmapItem`、`QFileDialog` 等头文件并且声明各个变量。

3. 在cpp文件中构造函数中实例化这些变量。

4. 给打开图片按钮添加槽函数 `on_btn_open_clicked()`，槽函数的定义如下：

```
void MainWindow::on_btn_open_clicked(){
    QStringList files;
    if(fileDialog->exec()){
        files = fileDialog->selectedFiles();
    }

    // 若没有选择文件
    if(files.length() == 0){
        return;
    }

    img = QImage(files[0]);
    showImage(img);
}
```

点击打开图片按钮以后触发该函数,首先建立一个 `QStringList` 类型变量，弹出选择框然后通过返回的路径加载该图片，然后调用 `showImage()` 函数进行图片的显示。

5. 上面的 `showImage()` 函数定义如下:

```
void MainWindow::showImage(QImage img){
    // 将图片设置为QGraphicsItem
    gpi->setPixmap(QPixmap::fromImage(img));

    // 将Item加入到QGraphicsScene
    gs->addItem(gpi);

    // 将QGraphicsScene添加到UI的graphicsView中
    ui->graphicsView->setScene(gs);
}
```

这样选择并且显示图片功能即完成了。

6. 创建一个新的类 `MyMouseEvent` 用于鼠标控制事件, 该类继承自 `QgraphicsPixmapItem` .

7. 在该新类中重写 `mouseMoveEvent`、`mousePressEvent`、`wheelEvent` 函数, 其中 `mousePressEvent` 不需要实现内容, 其他两个函数代码如下:

```
void MyMouseEvent::mouseMoveEvent(QGraphicsSceneMouseEvent *event){
    // 需要进行坐标系转换
    setPos(pos() + mapToParent(event->pos()) - mapToParent(event->lastPos()));
}

void MyMouseEvent::wheelEvent(QGraphicsSceneWheelEvent *event){

    int delta = event->delta();
    double factor = scale();
    // 缩放比例
    double rate = 0.1;

    if(delta > 0){
        factor *= (1 + rate);
    }else if(delta < 0){
        factor *= (1 - rate);
    }
    // 设置缩放中心点
    setTransformOriginPoint(boundingRect().width()/2, boundingRect().height()/
2);
    setScale(factor);
}
```

8. 在 `mainwindoww` 类中, 将之前使用的 `QGraphicsPixmapItem` 类型数据修改为 `MyMouseEvent` 类型。

9. 构建, 运行程序。

实验结果

1. 运行初始界面如下：



图 2. 初始界面

2. 打开文件选择窗口：

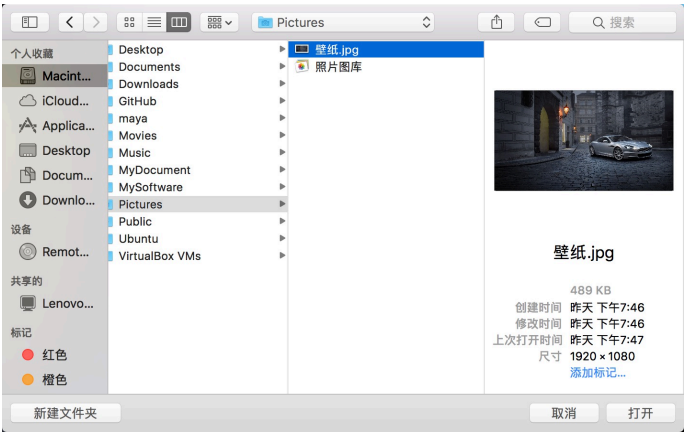


图 3. 文件选择

3. 选择文件后界面如下：

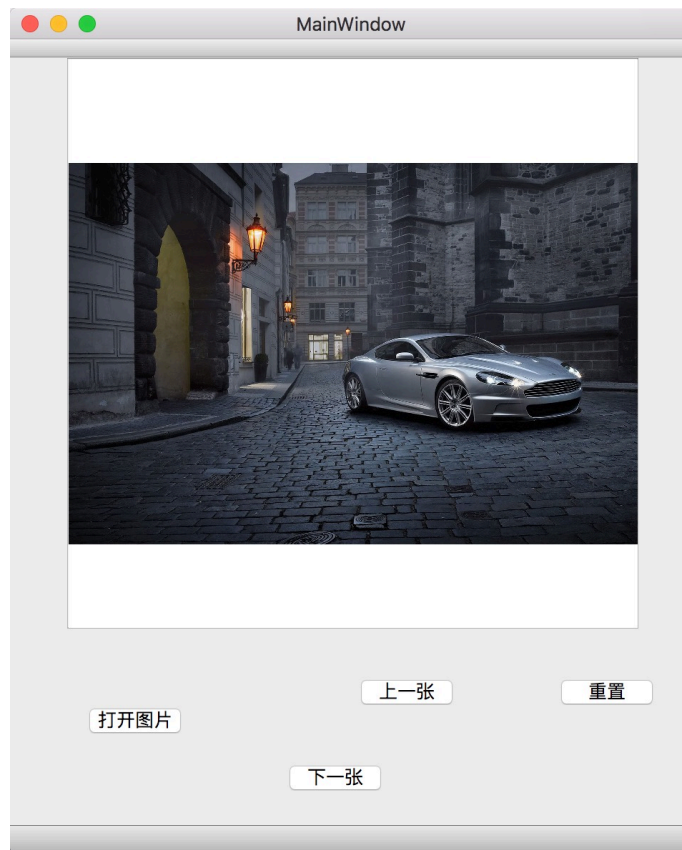


图 4. 选择文件后

4. 滚轮缩放：



图 5. 滚轮放大



图 6. 滚轮缩小

5. 鼠标拖动：



图 7. 鼠标拖动

问题&解决

- 在程序设计的过程中出现报错，每次程序关闭都会出错，如下所示：

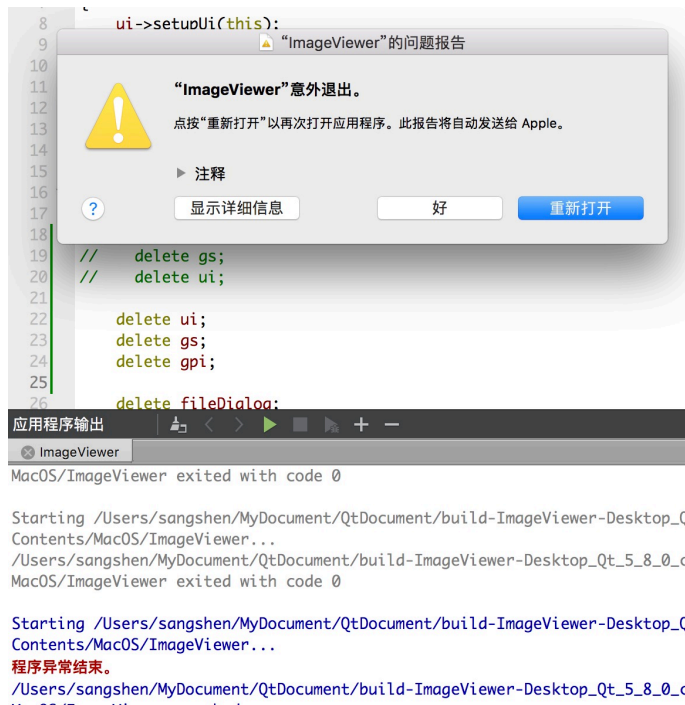


图 8. 报错信息

经调试后发现是析构函数出现问题，修改之前的析构函数为：

```
MainWindow::~MainWindow() {  
    delete ui;  
    delete gs;  
    delete gpi;  
    delete fileDialog;  
}
```

修改后为：

```
MainWindow::~MainWindow() {  
    delete gpi;  
    delete gs;  
    delete ui;  
    delete fileDialog;  
}
```

gpi要添加进入gs中，gs添加到ui中，所以析构顺序不可以弄错，这样就不会报错了。