

基于 QT 的飞行大战游戏设计与实现

刘 飞 白俊鸽

成都锦城学院计算机与软件学院 四川 成都 611731

【摘要】本文主要写了基于 Qt 和 C++ 技术在 Windows 开发环境下的一个飞行对战游戏，讲解了游戏实现的具体过程以及每一个类在游戏中有什么作用，实现的原理、过程。增加了一些小功能，增加的游戏的可玩性，本游戏操作简单，比较容易实现，适合新手学习。

【关键词】Qt; C++; 飞行游戏

1 引言

Qt 是跨平台的应用开发框架，支持各大主流操作系统。它既可以开发 GUI 程序，也可用于开发非 GUI 程序^[1]，Qt 是完全面对对象的，在 Windows 操作系统下安装好 Qt，将环境配置好就可以编辑 C++ 程序了。运行本项目就直接进入游戏，点击 Esc 按钮可以暂停游戏，弹出 Ui 界面，Ui 界面中可以选择继续游戏，退出游戏，查看历史记录。本文项目的重要知识点有如何让游戏中的图片自动化，如何改变游戏角色的运行轨迹，背景音乐、图片的添加，还有就是一些游戏角色的布局，最后将得分保存保存到数据库中。

2 每个类实现的功能

在本游戏中主要是通过 widget 类来实现游戏的运行，主要是使用了一个 m_Timer 每 10 毫秒刷新，调用坐标更新函数、绘图事件。config.h 类主要定义了游戏中的大部分参数，比如子弹的个数，速度，出现的刷新时间，敌机的数量，刷新时间等等，me 类主要实现了我方飞机的初始化，还有就是接收鼠标坐标的更新坐标函数，爆炸效果实现的有三个类，主要实现我方飞机、敌方飞机的爆炸效果，因为每个角色图片不同，所以每个角色的爆炸效果都是不同的，两个子弹类，分为我方发射的子弹和敌方发射的子弹，两则的区别就在于子弹的速度，刷新时间，子弹的个数等，Map 类就是实现地图的初始化，you 类就是敌机的初始化了。

3 界面的设计

本游戏有两个界面游戏界面以及游戏暂停时出来的界面，游戏界面就实现飞机大战，而暂停后弹出一个界面，可以返回游戏，退出游戏以及查看历史得分等等，因为有返回游戏按钮实现继续游戏，所以暂停的界面不能直接关闭，直接关闭会造成游戏界面一直暂停，无法退出也无法继续游戏，所以使用 setWindowFlags(Qt::FramelessWindowHint) 将暂停界面设计成无边框形式，就只能点击窗口里面的按钮，因为隐藏了标题栏，所以界面无法移动，所以需要重写鼠标点击、鼠标移动事件实现界面的拖拽功能。

4 具体类的实现

4.1 Map 类的实现

Map 是游戏地图类，我们直接让地图向下滚动，间接实现飞机向前飞行，所以我们需要两张图片，图片是

可以无缝连接的，然后定义图片的初始位置，第一张图片放在游戏界面的上边，第二张图片放入游戏界面的里面，设置滚动速度，然后创建一个坐标更新函数。地图的 y 坐标加上地图的滚动速度，更新图片，跑到底部时重置。

4.2 计时器实现

计时器 m_Timer 是这个游戏的核心部分，开始先设置 m_Timer.setInterval(10)，设置计时器 10 毫秒刷新一次，每一次刷新都要调用 Map 更新坐标函数，当坐标更新以后我们还要在我们的窗口显示出来，所以在调用 update 函数，update 是系统自带的调用 paintEvent

函数。计时器实现代码如下：

```
connect(&m_Timer,&QTimer::timeout,[=]() {
    isstop();
    // 敌机出场
    youtoScene(); // 敌机 1 出现
    you2toScene(); // 敌机 2 出现
    // 更新游戏中元素的坐标
    updatePosition();
    // 绘制到图片中
    update(); // 系统自带的调用 paintEvent 函数
    // 碰撞检测
    collisionDetection(); // 子弹击中敌机 1
    collisionDetection1(); // 子弹集中敌机 2
    Crash_Me();
});
```

4.3 我方飞机的实现

飞机是一张图片，所以在初始化时，需要将飞机的图片导入，需要用到 QPainter 类，QPainter 既可以绘制几何图形，也可以绘制像素映射、图像和文字。如果要在绘图设备（一般为窗口部件）上绘图，只需创建一个 QPainter，再将指针传到该设备中，即可实现作图操作。^[2] 设置飞机初始的 x, y 坐标，还要设置一个矩形边框，边框的大小就是飞机图片的大小，方便后面进行碰撞检测，本项目的飞机移动是靠鼠标移动实现的，所以需要重写 mouseMoveEvent 函数，点击鼠标时返回鼠标的 x, y 坐标，更新飞机的坐标，当下一次计时器 m_Timer 刷新时，调用绘图事件将飞机移动到鼠标所移动到的地方，我们需要给飞机的移动增加一个限制条件，判断鼠标的坐标是否超出边界，超出边界的 x 或者 y 的时候，x、y 就不能在增加了，避免飞机移动到界面的外边。

4.4 子弹类 (bullet) 的实现

子弹类与上面的几个类有许多相同的地方，都要初始化图片、位置、速度、刷新的时间间隔，第一个不同点就是发射的子弹是有很多，我们不可能创建几万发子弹，所以我们在 bullet 中增加了一个 bool 值，当这个值为 true 时，子弹是不会显示的，相当于将子弹放入了弹夹中，发射子弹时 bool 值改为 false，绘图事件将子弹画出来，当子弹超出了屏幕是 bool 值改为 true 放入弹夹，如此往复就可以实现飞机发射子弹了。具体的实现需要在发射子弹的类中增加一个发射子弹的函数，在这个类中如果子弹到达发射的时间，子弹为空闲状态就可以初始化子弹的 (x, y) 发射出去了。具体的代码调用如下图 1 所示：

```
//累加时间间隔记录
m_recorder++;
//未达到发射子弹时间间隔，不发射子弹
if(m_recorder<BULLET_INTERVAL)
{
    return;
}
//达到发射时间
m_recorder = 0;
//发射子弹
for(int i = 0 ; i < BULLET_NUM ; i++)
{
    if(m_bullets[i].m_free)
    {
        //将空闲状态改为假
        m_bullets[i].m_free = false;
        //设置子弹的坐标
        m_bullets[i].m_X = m_X + m_Rect.width()*0.5;
        m_bullets[i].m_Y = m_Y;
        break;
    }
}
```

图 1：子弹类实现代码

4.5 敌机类的实现

有两种敌机，所以设计了敌机类有两个，第一种敌机速度快，刷新的快，会左右移动，但是不会发射子弹，第二种与第一种的相比速度、刷新都比较慢，也不会左右移动，但是增加了发射子弹的函数。

两种敌机都需要使用 QPixmap 类添加图片对象，初始化位置，用 QRect 创建一个矩形框，在创建一个 bool 值，当被子弹击中后 bool 值变为 true，当 m_Timer 刷新到这时候，这个图片就会隐藏，然后就是在函数中对坐标进行更新。移动的基本理论是根据不断变动的坐标点，将图片按照坐标所在的位置进行重绘，在控制敌机的移动，需要达到一个随机路径的效果，可以定义几组轨迹函数，比如： $[3]\sqrt{\text{pow}(220, 2) - \text{pow}(\text{curPos.ry}() - b, 2)} + 2$ 。

而第二种敌机在第一种敌机的情况下增加了一个子弹类，我们需要修改子弹的运动方向，速度，个数等等。上面只是对敌方飞机进行了初始化，还没有在游戏界面上显示出来，所以我们要在 widget 中调用这个类，创建一个敌方飞机的数组，在更新坐标函数中进行飞机的出场，便利所有飞机，bool 值为 false 的放飞，绘图事件中绘制敌方的飞机，将刷新坐标的函数和绘制飞机图片的函数写入计时器中，当程序开始运行的时候刷新坐标和图片，这个类就实现了。

4.6 爆炸效果实现

在本项目中有三个爆炸类，分别是我方飞机和敌方两种飞机的爆炸效果，爆炸效果播放我们也要实现位置

的设置、爆炸的时间间隔设置。本项目的爆炸效果是直接播放几张图片，所以在初始化时将图片保存在 vector 中，爆炸的坐标还要更新。实现爆炸效果是在 widget 类中，这里主要讲我方子弹击中敌方飞机的实现，遍历所有的敌机，判断 bool 值，当 bool 值为 true（表示这个飞机没有在图中），continue，在便利我方飞机发射的子弹，判断 bool 值，当上面两个 bool 值都为 false 时在进行判断碰撞测试，代码如下：m_enemys[i].m_Rect.intersects(m_hero.m_bullets[j].m_rect); 这个函数就用到我们类中初始化的 QRect，判断子弹是否与飞机碰撞了，碰撞后将两个 bool 值改为 false 值（子弹和飞机都消失），在将 (x, y) 坐标赋值给爆炸类中的 (x, y)，刷新播放爆炸效果图。其他的爆炸效果，比如我方飞机撞击敌方飞机，我方飞机中弹等，与上面的过程大同小异，便利的对象要变，判断碰撞的双方要变，爆炸的效果需要变等等。

4.7 其他类的实现

点击按钮 Esc 暂停游戏弹出窗口，在主窗体类中声明键盘响应函数 keyPressEvent，判断输入的时 Esc 键：if (event->key() == Qt::Key_Escape)，执行 m_Timer.stop() 使游戏暂停，弹出暂停界面，因为在暂停界面类中是没法重启计时器，所以在写个自定义的信号和槽实现继续继续游戏，而暂停界面的退出按钮也是自定义信号和槽，不过槽函数变为 exit(0)；

链接 MySQL，本项目是通过 ODBC 链接 MySQL 的，在链接数据库类中，主要有三个函数，第一个函数实现链接 MySQL，当链接成功时在界面输出“成功”，第二个函数实现将数据插入到 MySQL 中，当我方飞机爆炸时调用这个函数，将分数存储在 MySQL 中，第三个函数是将 MySQL 表中的数据传入到类中的 QVector 中，在遍历 QVector 将数据填入到 tableView，在使用 tableView 前需要创建一个标准的条目模型，代码如下：

```
this->model = new QStandardItemModel;
this->ui->tableView->setModel(model);
```

然后在调用 model 类添加表头数据就可以了。

播放爆炸时的音效 QSound::play(SOUND_BOMB); 而背景音乐播放需要 setLoops 实现循环播放的次数，当里面参数为 -1 时为无限循环。游戏运行后效果如图 2：

图 2：游戏成品截图



5 结束语

本文介绍了如何在 Qt 中实现飞行大战游戏，但是还有很多的不足点，比如在游戏播放爆炸效果的时候会出现短暂的卡顿现象，游戏中可以在增加一些动画，让界面变得在酷炫一点，当取得高分的时候请求输入玩家的姓名，给历史记录排序且只显示前 10 名，增加关卡的 boss 等等。由于本项目的代码量并不是很多，代码通俗易懂，Qt 初学者是可以这个项目来练习。

【参考文献】

[1] 袁媛, 王延红, 江凌, 林成地, 王玥. 基于 Qt 及 OpenCASCADE 的建模技术研究 [J]. 现代电子技术, 2013, 36(10): 74-77.

[2] 郑向宁. QT 坐标变换及应用 [J]. 电子技术与软件工程, 2015(05): 91.

[3] 崔志斌, 成荣荣, 李珂. 基于 QT 的飞行射击游戏的角色安排 [J]. 电脑知识与技术, 2017, 13(36): 58-59+62.