

1.

本文档主要介绍坦克大战游戏的整个设计流程,以及过程中所遇到的问题。

# 坦克大战

项目说明书

小组成员:

周展鹏、苏恺淇、蔡礼权

## 游戏介绍

本游戏是简单的 3D 坦克大战,玩家可以以第一视角或者第三视角对坦克进行控制,通过移动坦克和调整坦克炮台来发射炮弹,消灭坦克。

## 2. 游戏规则

- 玩家可以任意拾取某一辆坦克,对其进行控制。
- 其他坦克全部消灭后会全部重新出现。
- 玩家在规定时间内消灭所有的坦克达到  $N$  轮,则胜利,否则失败。

## 3. 详细设计

### 3.1 开发环境

- 系统环境
  - Windows7
- 使用工具
  - Visual Studio 2010
  - Visual Studio 2008
  - PhotoShop CS6
- 编程语言及所用库
  - C++
  - OpenGL

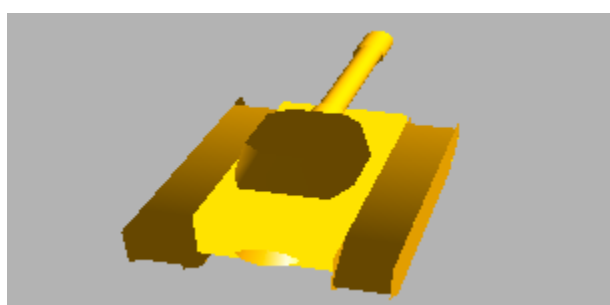
## 3.2 3D 建模

### 3.2.1 基础模型

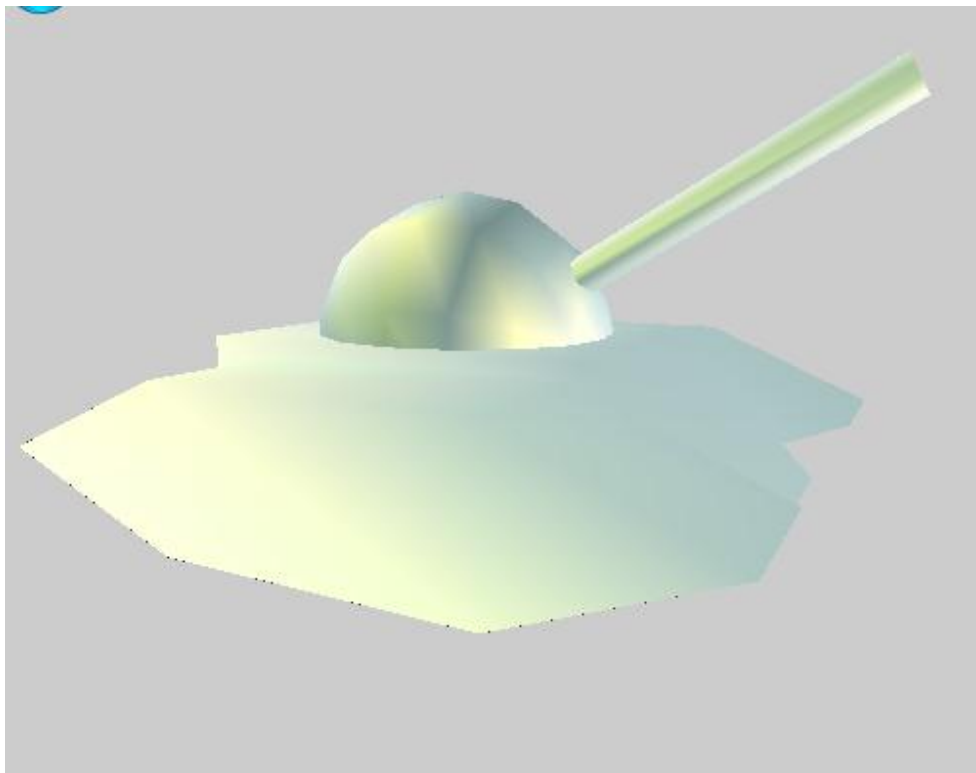
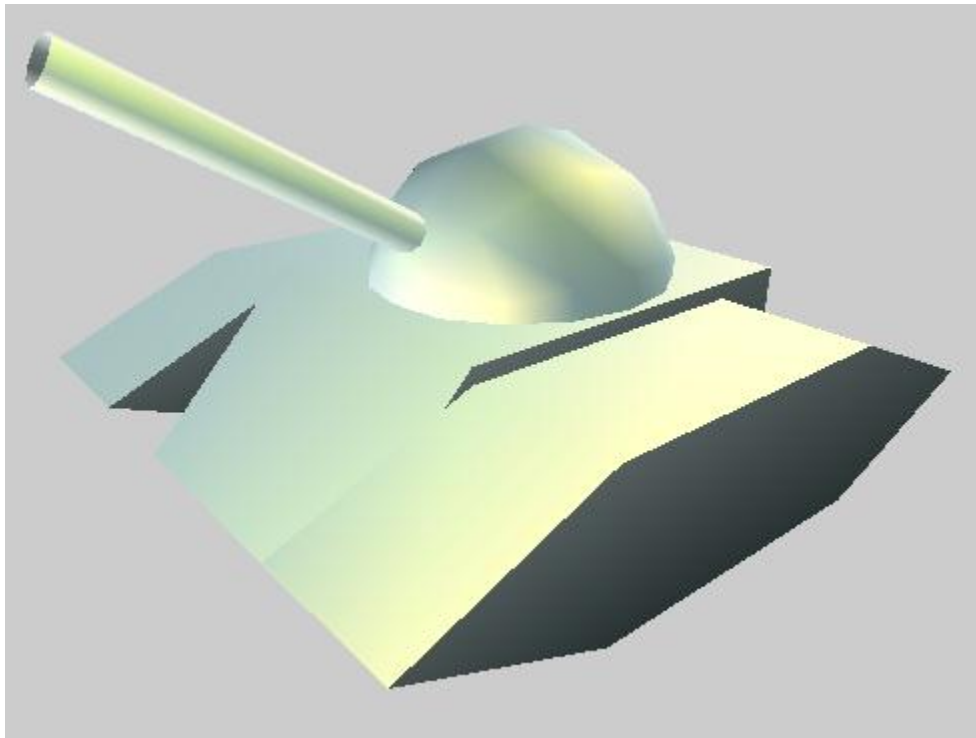
#### 3.2.1.1 坦克模型

本游戏所使用模型为第一、二次作业的坦克模型，小组每个组员都各自设计了一只坦克。因为之前的作业都已经交过设计的详细文档，就不在这重复了。

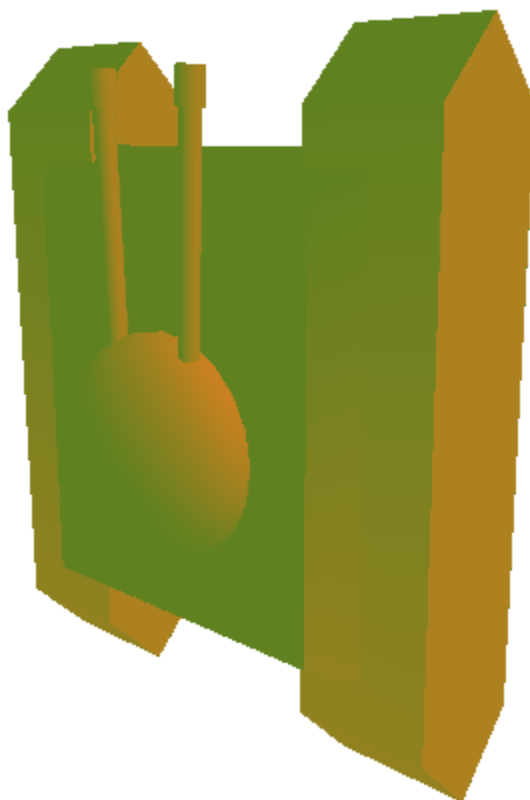
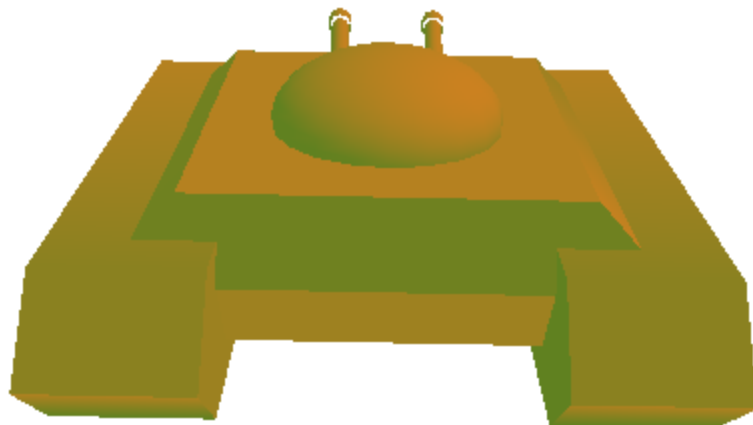
##### ➤ Tank\_Model\_Cailq



##### ➤ Tank\_Model\_Sukq



➤ Tank\_Model\_Zhouzp



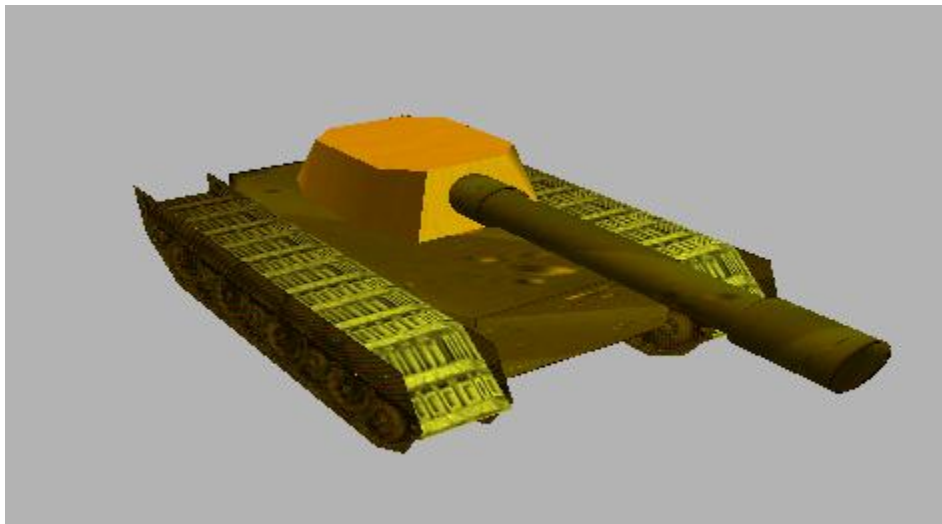
### 3.2.1.2 陆地模型

陆地模型的本体为一个  $200*200*1$  的长方体，在陆地上会设置有随机生成的石头。石头由球体变形，并贴图得到。

### 3.2.2 光照纹理

#### 3.2.2.1 纹理贴图后的坦克





### 3.2.2.2 纹理贴图加整体光照之后的坦克世界

在纹理这块，本来是每个小组成员都有贴图，但是不知道是由于vs2010，还是本项目本身的设计问题，位图加载过多会产生系统混乱。问题至今尚未解决，所以只能暂时使用其他人的贴图。



### 3.3 系统框架

本程序主要由控制系统类、物理世界类、坦克类、炮弹类四个主要类来实现，其他包括物理世界障碍物类，以及坦克派生类。

➤ 控制系统类：

数据成员：包含有坦克对象，和物理世界对象，以及其他参数；



主要功能：用于实现对来自用户交互的响应函数接口；控制坦克还有物理世界对象的生成规则，以及碰撞的检测。

➤ 物理世界类：

数据成员：包含物理世界障碍物对象，以及其它参数；

主要功能：绘制物理世界（长方体地面），随机生成物理世界障碍物并绘制，生成障碍物链表以便碰撞检测。

➤ 坦克类：

数据成员：包含坦克基本参数和交互参数；

主要功能：在此基类中定义坦克的各项基本参数，和各项交互参数，通过对基类坦克进行交互实现即可对派生坦克进行操作。

➤ 派生坦克类：

数据成员：继承坦克基类的所有参数，并且拥有自身绘制模型所需的数据成员；

主要功能：实现坦克基类的虚函数，绘制坦克模型，并实现交互。

➤ 障碍物类：

数据成员：障碍物本身属性；

主要功能：实现障碍物的绘制。

➤ 炮弹类：

数据成员：拥有炮弹的基本属性，包括速度、爆炸范围等；

主要功能：坦克发射炮弹时产生，并绘制出炮弹发射的轨迹和爆炸效果。

## 3.4 系统交互

### 3.4.1 界面交互

#### 3.4.1.1 视角变换（通过鼠标右键点击）

##### ➤ 坦克第一视角

目标：当玩家选中一辆坦克之后可以获得坦克的第一视角，并且视角跟随坦克的移动的移送而移动，同时视野所在范围能够跟随坦克炮台的转动而改变。

实现：1. 获得坦克当前位置，并以此作为视点所在位置；

2. 获得坦克初始化方向以及坦克炮台转动参数，并由于坦克设计时以  $z$  轴正半轴为正方向，可通过以坦克为中心设置向外看的球坐标，算出视点所看方向。

##### ➤ 坦克第三视角

目标：以第三视角俯看坦克运动，并跟随某一选中坦克。

实现：以某一坦克所在位置作为视点观察中心。

---

#### 3.4.1.2 菜单设计

实现了鼠标中间键点击弹出菜单。

- 选中坦克：子菜单项，可直接选中某号坦克；
- 全自动：所有坦克自由移动；

---

#### 3.4.2 坦克交互

##### ➤ 坦克旋转和移动

目标：可以通过控制 **wsad** 四个键实现坦克的旋转和移动。

实现：通过设定移动交互参数，包括速度，转速等，并通过键盘回调函数和设置 **OpenGL** 相应模型变换函数参数来实现。

##### ➤ 炮台移动

目标：通过上下左右键四个键实现坦克炮台的旋转上下移动。

实现：通过设定交互参数，包括转速等，并通过键盘回调函数和设置 **OpenGL** 相应模型变换函数参数来实现。

##### ➤ 发射炮弹

目标：空格键发射炮弹，并实现炮弹单发，以及确定炮弹发射角度，运动轨迹。

实现：通过检测坦克炮弹发射状态来确定炮弹发射的状态，通过根据炮台转动角度和坦克偏移正方向角度来确定炮弹发射方向，并通过抛体运动的路径函数来实现坦克炮弹的运动轨迹。

---

#### 3.4.3 元件拾取

### ➤ 拾取坦克

目标：可以随意选中所想要的坦克，并对其进行控制。

### ➤ 拾取物件

目标：可以拾取陆地上可见的能够拾取的物体。

## 3.5 碰撞系统

本游戏的碰撞系统主要通过将所有可碰撞对象以链表形式存储，通过循环检测每个对象所包含的碰撞半径与两两对象之间的距离关系，来判定是否发生碰撞。

### ➤ 坦克与边界的检测

目标：检测坦克与陆地边界的碰撞，并通过检测结果来限制坦克的移动。和弥补随机生成坦克可能造成的坦克出界，将其置原点，并通过碰撞调整函数，调整。

### ➤ 坦克与坦克的检测

目标：检测坦克与坦克之间的碰撞，并解决坦克生成互相黏在一起的bug。

### ➤ 炮弹碰撞爆炸的检测

目标：检测炮弹是否与石头或者坦克发生碰撞，并检测炮弹是否与地面接触。

## 3.6 游戏规则实现

## 4. 任务分工

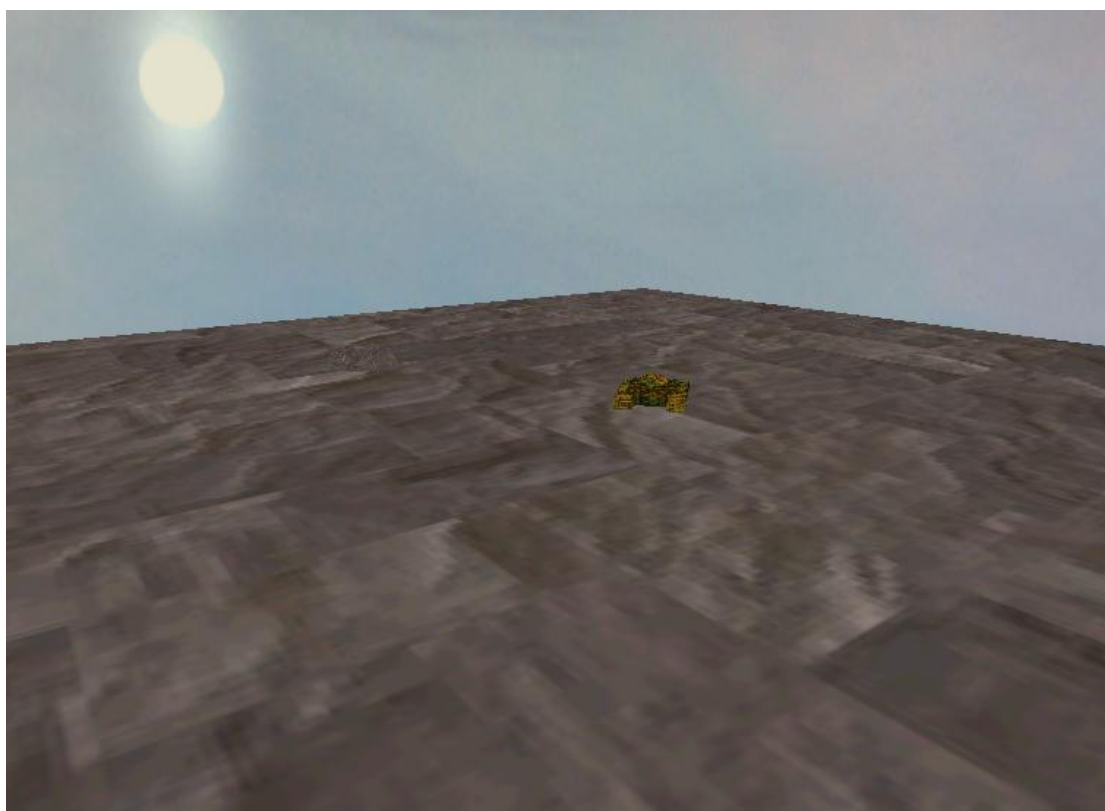
**周展鹏：**系统框架的搭建、坦克和石头边界的碰撞检测、模型导入整合、坦克交互设计实现、开发整合和文档撰写

**蔡礼权：**对象拾取、视角变换以及跟踪、坦克炮弹的发射以及运动轨迹的实现、炮弹碰撞爆炸检测及效果实现、改进键盘交互机制

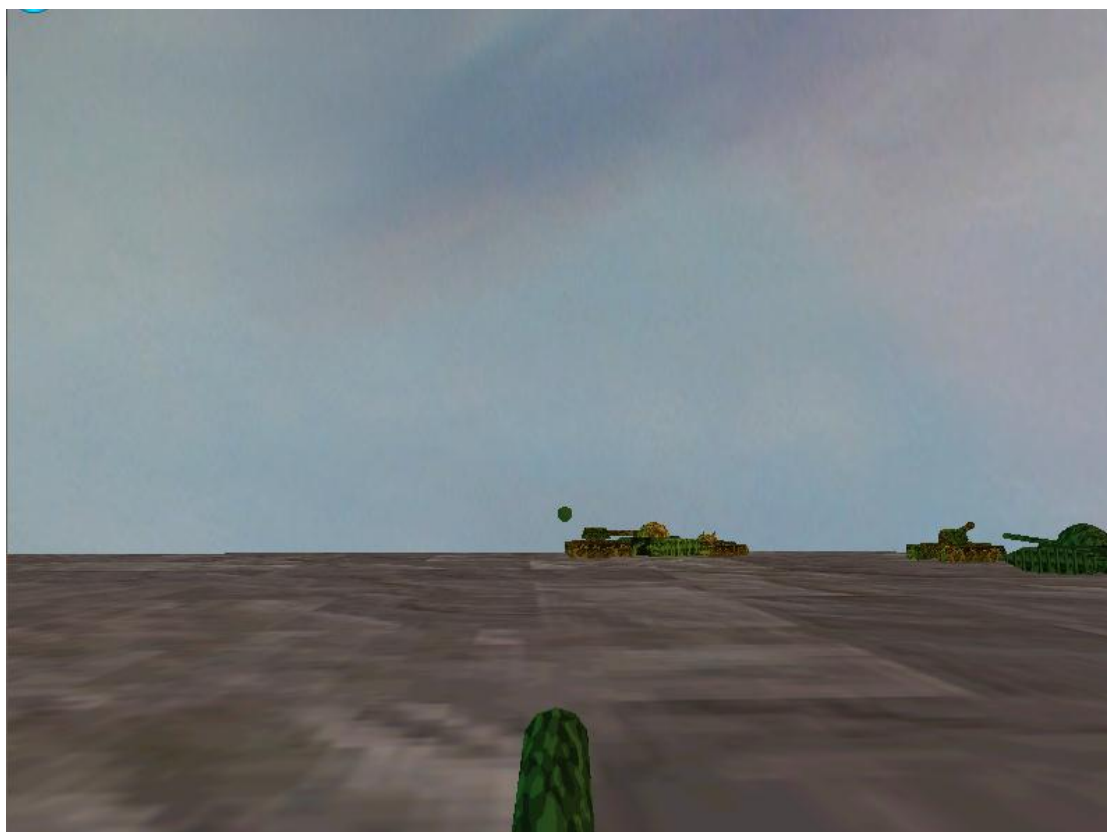
**苏恺淇：**菜单功能的设计实现、物理世界场景建模、场景贴图、整体光照、按键捆绑设计实现

## 5. 成果展示

### ➤ 第三人称视角



➤ 第一人称



## 6. 游戏说明

- 鼠标左键：点击拖动旋转视角
- 鼠标中键：点击可启用菜单
- 鼠标右键：点击坦克可以拾取，选中坦克状态下可切换视角
- 键盘 WSAD 键：移动坦克
- 方向键：移动坦克炮台
- 空格键：发炮