# 计算机图形学基础

# 网格简化 实验报告

陈键飞 2010011291

# 实验目的

使用网格简化算法简化一个网格。即，将原来面片数较多的三角网格简化成为面片数较少的网格。并且使得模型尽量保持原样。

# 实验原理

使用了边塌缩的网格简化算法，其原理如图1。一般来说，每塌缩一条边，减少以它为一边的两个面片。

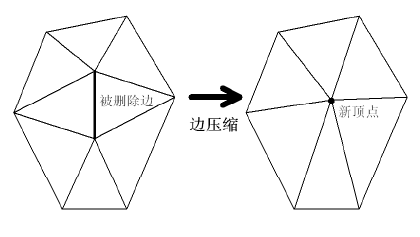


图 1：边塌缩原理

边塌缩方法需要考虑的两个问题是：

* 塌缩哪条边；
* 边塌缩后，原有的两个顶点移动到哪个位置。

本实验使用了[1]中使用的方法。使用一个二次型作为估价函数，每次选取代价最小的边塌缩。此外，还可以解得使代价最小的新位置。

更详细地说，将塌缩后的顶点*v*与原来包含它的面片的距离和的平方作为估价函数。设*v*的齐次坐标为*v*，平面*pi*的齐次坐标为*pi*。则两者的距离就是其点积。

其中=，。这样，将移到的代价就是。

如果对代价函数求导，会发现使其取得最小的是方程组

的解。如果这个方程组无解，不妨就单纯地取。

# 实验内容

#### 算法

本实验对比了三种网格化简方法的效果：

1. 以边的长度作为估价函数；
2. 以(1)式作为估价函数，取
3. 以(1)式作为估价函数，取(2)式作为新的顶点。

#### 可视化

本实验开发的工具支持三种使用方式：

1. 以动画的形式对比原型和3种方法的效果（图3）。
2. 指定目标面片数，在应用程序中查看模型（图2）。
3. 指定目标面片数，导出obj文件，使用其他软件查看模型（图2）。

1. (b)

图 2：使用OpenGL渲染的模型(a)，使用3Ds max查看生成的模型(b)

# 实验效果

#### 算法对比

对比三种方法的简化模型如图3。同预料中一致，方法1因为优先简化短边，丢失了高频的细节特征；方法2保留了细节，但面片的位置有一定误差。方法3与原模型最为接近。

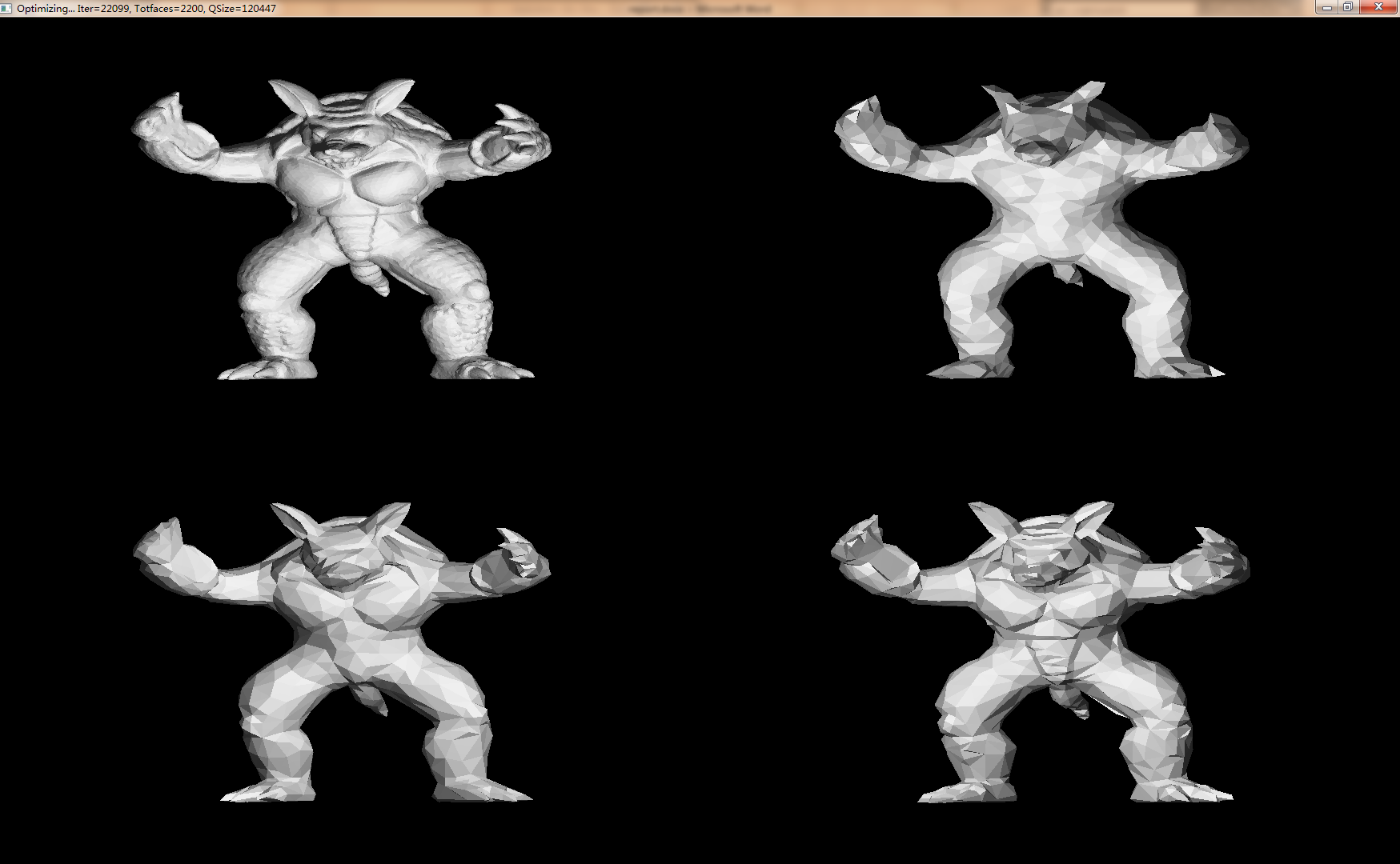


图 3：将46398个面片的Arma模型简化到2200个面片。左上、右上、左下、右下分别为为原模型、方法1、方法2、方法3

#### 面片数对模型质量的影响

不同面片数的效果对比如图4、图5。可以发现在这个大小下20%的面片数与原模型除了光滑程度差异不大。如果查看缩小后的模型，1%的面片数仍然可以看清物体的大致形状。

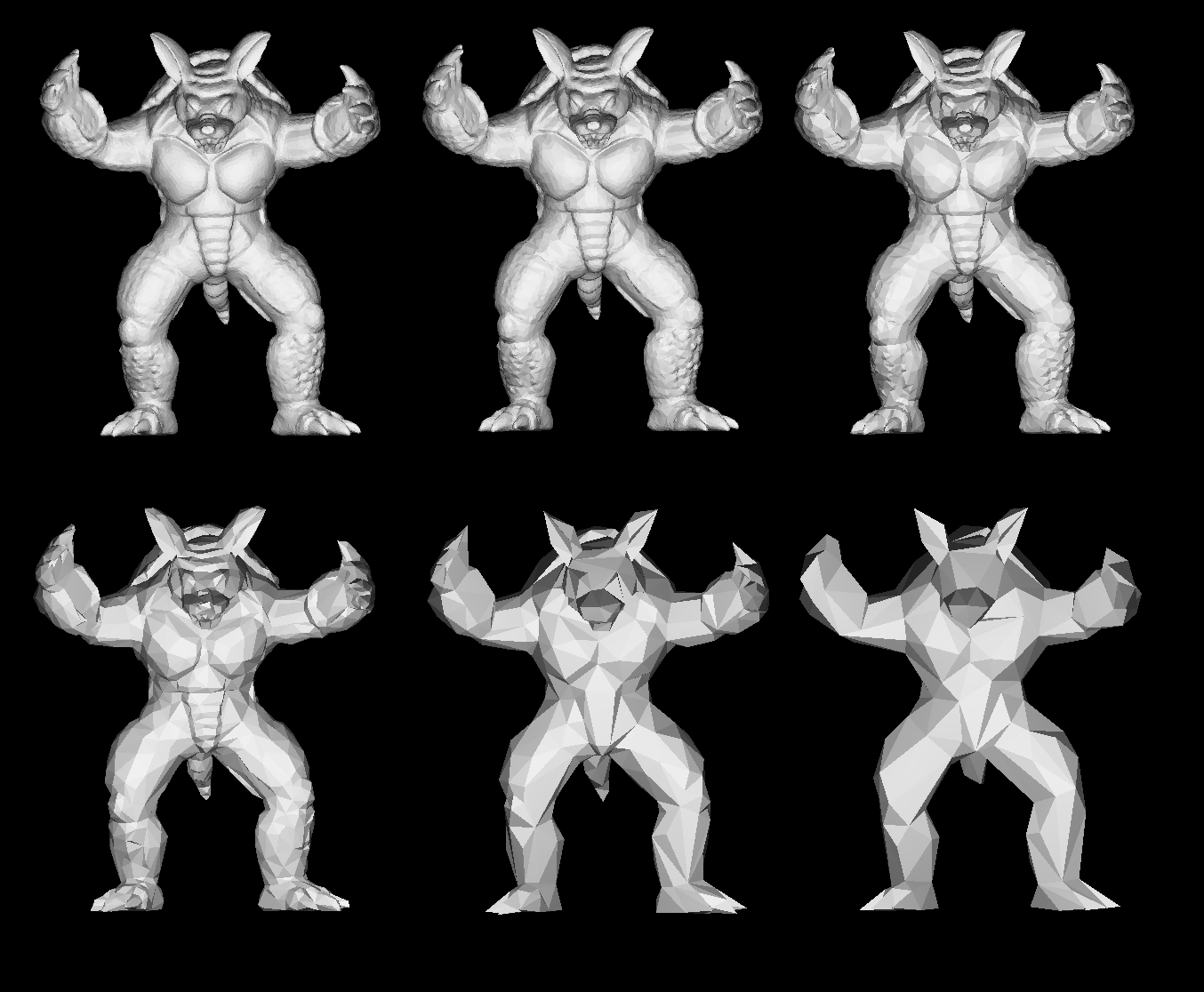


图 4：Arma模型，依次为100%、50%、20%、7%、2%、1%的面片数，最后一个模型共464个面片。



图 5：Buddha模型，依次为100%、50%、20%、7%、2%、1%的面片数，最后一个模型共1230个面片。



图 6：缩小后的Buddha模型，六个模型差别不大

# 实验总结

本次实验基本完成了预定目标。网格简化是强有力的解决存储空间和计算资源的工具。1%的面片在远距离看与原图形差别并不大，在渲染时可以大大节省计算开销。

# 参考文献

[1] Michael Garland and Paul S. Heckbert. 1997. Surface simplification using quadric error metrics. In Proceedings of the 24th annual conference on Computer graphics and interactive techniques (SIGGRAPH '97).