
北京航空航天大学

2023 - 2024 学年 第一学期期末

《计算图形学》

期末课程报告

班 级 2106** 学 号 21371***

姓 名 *** 成 绩

2023 年 12 月 31 日

第一部分：图形绘制(70%)

1.1 (必做题) 简述对图形绘制流水线中几何处理和像素处理两个阶段的认识和理解。(包括但不限于固定绘制流水线，可以从对两个阶段的组成、功能、转换、数据、问题和能否合并成一个阶段等多角度阐述自己的**系统认识**和理解)。(不少于300字)

图形绘制流水线是计算机图形学中用于生成图像的基本流程。它通常包括几何处理阶段和像素处理阶段。

(1) 几何处理阶段

几何处理阶段负责处理图形的几何属性，如形状、位置、方向等。其**组成及功能**包括：

1. 顶点处理：图形的基本构建块是顶点，这一阶段对顶点进行变换，将其从模型空间变换到世界空间、视图空间，最后变换到投影空间，以便后续的投影和裁剪操作。
2. 裁剪和图元组装：裁剪是为了排除在视锥体外部的几何图元部分，以减少后续处理的工作量。这包括裁剪在视锥体之外的顶点，避免不可见的几何体进入光栅化阶段。在裁剪之后，可以将通过裁剪的顶点组装成几何图元，如三角形或线段。这个步骤将顶点按照定义的图元拓扑关系组合成更高级的图元结构。

几何处理阶段的主要功能是对图形进行空间变换和裁剪，**转换数据**包括顶点坐标、变换矩阵等。主要**问题**包括几何表示精度问题、裁剪算法效率等。

(2) 像素处理阶段

像素处理阶段则负责处理图形的像素信息，包括颜色、光照、纹理等。其**组成及功能**包括：

1. 光栅化：将几何处理阶段输出的图元（如三角形）转换为屏幕上的像素，形成所谓的片元，以便进行后续的像素处理。
2. 片元处理：片元处理阶段是对每个像素进行的最后一步处理。这个阶段包括执行着色、深度测试、透明度计算等操作，以决定最终像素的颜色和属性。

像素处理阶段的主要功能是确定像素的最终颜色值，**转换数据**包括片元的位置、颜色信息等。主要**问题**包括着色器效率、纹理映射质量、像素混合算法等。

（3）两阶段能否合并？

虽然几何处理和像素处理阶段都有各自的功能和数据流，但在实际图形渲染中，它们通常被分开以利于并行处理，提高图形渲染的效率。然而，一些现代图形渲染技术，如延迟渲染和着色器中的几何处理，逐渐模糊了这两个阶段的界限，使得一些处理可以在一定程度上合并或重叠，以提高效率和灵活性。

（以下五道题中选作三题）

1.2 （5 选 3）证明由旋转和平移组成的任何变换序列都等价于先作一个以原点为不动点的旋转然后再进行一次平移。

证明：

引理一：旋转变换的复合仍为旋转变换，平移变换的复合仍为平移。

引理二：对于先旋转再平移的操作，总能找到一种效果相同的先平移再旋转的操作，即对于任意平移矩阵乘旋转矩阵 TR ，总能找到相应的 $R'T'$ ，反之亦然。

引理一、二的正确性是显然的。下面证明题目要求的命题，即对于由旋转和平移组成的任意变换序列 $M_1M_2...M_n$ ，总能找到一个以原点为不动点的旋转然后

再进行一次平移的变换 TR 与其等价。

首先反复使用引理二，不断交换旋转变换和平移变换的顺序，将原序列 $M_1M_2...M_n$ 转换为如下形式：

$$M_1M_2...M_n = T_1T_2...T_iR_1R_2...R_j$$

其中 T_i 为平移变换， R_i 为旋转变换。

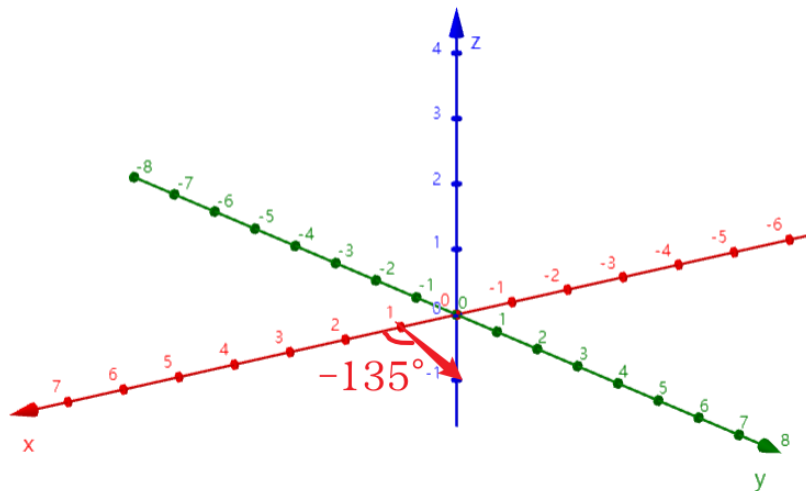
接下来使用引理一，将平移变换 $T_1T_2...T_i$ 和旋转变换 $R_1R_2...R_j$ 各自合并为一次变换 T 和 R ：

$$T_1T_2...T_iR_1R_2...R_j = TR$$

故 $M_1M_2...M_n = TR$ ，得证。

1.3 (5 选 3) 如果我们用正投影来绘制坐标轴，那么 x 轴和 y 轴将位于纸面，但 z 轴将指出纸面。现在设计一个新的投影方式，使得坐标轴投影后 x 轴和 y 轴之间的夹角成 90° ， z 轴与 x 轴之间的夹角成 -135° 。求这个投影图所对应的投影变换矩阵。

设原始坐标系为三维正交坐标系，观察到 $-z$ 的方向向量 $+x$ 的方向向量相减可以得到与 x 轴成 -135° 的向量，如下图所示：



将这个向量单位化并作为新的+z 方向，即可构造出 x 轴和 y 轴之间的夹角成 90° ，z 轴与 x 轴之间的夹角成 -135° 的投影矩阵：

$$P = \begin{bmatrix} 1 & 0 & -\frac{\sqrt{2}}{2} & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{\sqrt{2}}{2} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

1.6(5 选 3)简要说明法线贴图中 TBN 矩阵的定义和作用。

TBN 矩阵是一个三阶矩阵，通常用来转换局部空间中的法线贴图纹理坐标系到世界空间或相机空间中。

TBN 代表切线 (Tangent)、双切线 (Bitangent) 和法线 (Normal)，它们用来描述表面的局部几何属性。TBN 矩阵由以上三个向量在**目标空间**的坐标 (世界坐标或相机坐标) 构成，这样就可以用于将纹理空间中的法线向量转换为目标空间中的法线向量，以便准确地模拟表面细节并进行正确的光照计算。

下面的公式展示了 TBN 矩阵进行坐标变换的过程：

$$\mathbf{n}_{world} = TBN * \mathbf{n}_{local}$$

第二部分：几何建模(25%)

2.1 (必做题) 简述几何建模中对曲面建模方法中三种不同的形式：多边形网格模型、参数曲面方程和符号距离函数隐式模型的理解和各自特点的认识。(不少于300字)

在几何建模中，对曲面进行建模常用的方法包括多边形网格模型、参数曲面方程和符号距离函数隐式模型。这些方法各自具有不同的特点和优势，适用于不同的建模需求。

1. 多边形网格模型：

多边形网格模型是最常见和直观的曲面表示方法之一。它由许多小的多边形组成，这些多边形相连形成网格，从而近似表示出整个曲面。多边形网格模型适用于描述复杂的表面结构和形状，易于渲染和显示。它也是许多计算机图形学和动画领域的标准表示方式。然而，多边形网格模型可能需要大量的面来准确表示曲面的细节，因此在处理高度曲面时可能需要大量的计算资源。

2. 参数曲面方程：

参数曲面方程使用参数化的数学方程来描述曲面。通常以参数化形式表示 x 、 y 和 z 的函数，如 Bezier 曲线、B 样条曲面等。这些方程提供了对曲面的精确控制，能够很好地描述平滑曲面和几何形状。参数曲面方程能够有效地进行曲面变形和编辑，广泛应用于 CAD 软件和图形设计领域。然而，它们可能难以捕捉某些复杂形状或特定几何特征，且在一些情况下可能需要复杂的数学运算。

3. 符号距离函数隐式模型：

符号距离函数隐式模型使用数学函数来表示曲面，函数值为某一点到曲面最近点的距离。它以隐式函数形式表示曲面，使得在空间中能够精确地描述曲面的形状和拓扑特征。这种模型可以更好地处理曲面的拓扑变化和复杂几何结构，例

如处理融合、分裂和形变等操作。但是，隐式模型的计算代价较高，特别是在高维空间中。

这三种方法各有优势，可以根据具体的应用需求选择合适的模型。多边形网格模型适用于动画、游戏等领域的实时渲染；参数曲面方程适用于 CAD 和建模软件中的精确几何设计；而符号距离函数隐式模型则更适合处理复杂曲面的拓扑变化和形变。选择合适的曲面建模方法，需要考虑到所需精度、计算资源、几何复杂性等因素。

(以下两道题中选作一题)

2.3 (2 选 1) 简述如何利用贝塞尔曲线的递归特征来实现三次插值多项式曲线的绘制。

三次贝塞尔曲线可以通过四个控制点来定义，这四个点分别为起始点 P_0 ，结束点 P_3 ，以及两个中间控制点 P_1 和 P_2 。利用递归特性，可以实现三次插值多项式曲线的绘制，步骤如下：

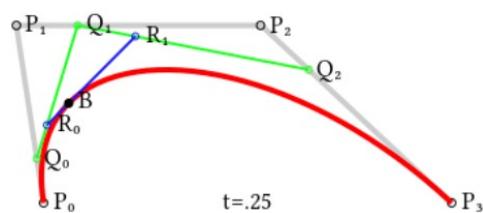
1. **定义控制点：**确定四个控制点 P_0 、 P_1 、 P_2 和 P_3 。
2. **递归计算中间点：**根据递推公式进行递归计算。贝塞尔曲线的递推公式如下：

$$P_i^k = \begin{cases} P_i, k=0 \\ (1-t)P_i^{k-1} + tP_{i+1}^{k-1}, k>0, i=0,1,\dots,n-k \end{cases}$$

其中 P 是点， k 是贝塞尔曲线的阶数， t 是按一定步长增加的参数，在 $[0, 1]$ 范围内变化。题目要求绘制三次曲线，所以 n 为 3，且递归到 $k=3$ 停止。

下图形象的展示了计算三次贝塞尔曲线的递归过程。灰色是 $k=0$ 时 4 个点的初始条件，绿色是递归到 $k=1$ 时得到的 3 个点，蓝色是递归到 $k=2$ 时得

到的 2 个点，最后递归到 $k=3$ 时停止得到最后一个点。将 t 按照步长增加，重复这个递归过程，就可以得到一组点来组成曲线。



3. **绘制曲线：**将计算出的点连接，即可绘制出一段三次贝塞尔曲线。

在实际应用中，绘制贝塞尔曲线时需要控制递归的次数或精度，以便在保持平滑性的同时控制计算的复杂度。

第三部分：课程反馈(5%)

3.1 (必做题) 谈谈你选本门课程的目的或初衷是什么？你认为本学期课程多大程度上满足了你的目的？对本学期课程学习的意见或建议。(不少于300字)

年龄很小时，我就痴迷于游戏和电影中栩栩如生的画面，一直很想把自己脑中的幻想世界也用 CG 表现出来。

到了中学，我的动手能力比童年提高了不少，便学习了很多软件，试着做了一些小玩意。一开始偏向于平面视觉，比如用 AE 做一些特效。之后便开始尝试三维领域，比如用 Blender 做一些简单的模型和动画。

高考后填志愿时，想填报相关的专业，比如数字媒体技术。结果发现这类专业一般归于艺术和应用方面，北航这个分段的学校，本科几乎不开设这种专业。

既然没机会去创作图形，那学习背后的原理也不错。所以我选择了计算机科学与技术专业，希望能学习图形学相关的理论。可在计算机领域明明如此重要且基础的一门学科，我没想到本科居然不会开设图形学的必修课……图形学真的很让人着迷。它背后的数学原理和计算机结合的相当美妙，最后呈现到荧幕上的效果又非常漂亮。

于是我便从大一开始等，一直等到大三，终于可以在教务上选一门图形学的课了！

这学期的课程基本满足了我的预期。它介绍了图形学的大部分领域，同时没有浮于表面，而是挑选了一些典型案例详细介绍背后的原理。此外，还要求我们动手实践，完成指定的实践任务。完成任务的成就感真的很让人开心，更锻炼了我们自学和编码的能力。

如果说对课程有什么建议，我还是希望对于课后实践能添加一些指导吧！虽然网络的资料非常丰富，但是第一次接触还是有些摸不到头脑。课程可以为每一个实践作业提供一份简易的指导书，能指点一些自学的方向就好。