

学号	姓名	论文规范性 (10)	问题分析与调研 (30)	方案创新性 (20)	实验结果分析与讨论 (40)	结课论文总成绩 (100)
21301016	索昊	6	22	15	33	76

实验方法描述过于简单；缺乏详细结果分析



## 计算机图形学课程设计（论文）

### 基于 Koch 曲线的分形图形绘制

### Drawing fractal graphics based on Koch curves

学 院： 软件学院

专 业： 软件工程

学生姓名： 索昊

学 号： 21301016

指导教师： 吴雨婷

北京交通大学

2024 年6 月

## 摘要

分形理论是近二十年发展起来的一门新兴理论，其发展至今已经成功融入多个领域的实际应用。但由于其复杂性与无规则性，很难通过手工绘制的方式获取分形图形。借由其自相似性的特性，使用计算机通过递归的算法称为绘制分形图形的有利方法。除去使用基本的分形理论进行分形图形绘制以外，科学家和图形工作者使用计算机总结出了很多分形图形的绘制方法。本文主要基于Koch 曲线的方法进行了分形图形的绘制，并根据递归深度对绘制图形进行对比。

**关键词：**分形理论，计算机图形学，Koch曲线，OpenGL

## 目录

摘要 .....	2
1. 引言 .....	4
2. 相关工作介绍 .....	4
3. 方法描述 .....	4
4. 实验设置 .....	4
4.1 GLAD.....	5
4.2 GLFW.....	5
4.3 GLM.....	5
4.4 GLUT.....	5
5. 实验过程与结果 .....	5
5.1 实验过程概要 .....	5
5.2 实验结果 .....	6
6. 结论 .....	7
参考文献 .....	8

## 1. 引言

世界上的图形千变万化，大部分图形都是结构复杂的，结构简单规整的图形很少，除了三角形，四边形等用欧氏几何可以绘制出来的，大部分很难通过手工绘制的方法获得，分形图形是其中的一种。现在随着计算机的发展和算力的提升，这类图形逐渐可以通过计算机进行绘制。

自分形几何诞生以来，科学家和图形工作者已经用计算机绘制了许多分形图形，如 Cantor 集；Koch 曲线；Sierpinski 集；Julia 集 Mandelbrot 集等，还有一些根据分形理论生成的图形。分型学的应用也十分广泛。

本文将分形理论作为理论基础，借助 OpenGL 作为计算机绘制图形的基本攻击，以 Koch 曲线作为分形图形的绘制方法进行分形图形绘制。本文根据递归深度对绘制的结果进行区分，并且对各类图形进行对比。

## 2. 相关工作介绍

分形理论是近二十年才发展起来的一门新的理论，其发展至今，已在自然科学领域，社会科学领域取得显著成果。有关分形学的国际会议有增无减。

分形理论越来越多地被应用到计算机图形学的领域。计算机不断提升的算力，为非规整形状的图形绘制提供了有利工具。借助计算机生成分形图形，从少数数据生成复杂的自然景物图形，是我们在仿真模拟方面前进了一大步<sup>[1]</sup>。虽然分形具有无规则性和复杂性，但自相似性是分形理论的基础，所以可以借助一些算法在计算机上生成分形图形，如分形的递归算法、分形的字符串的替换算法、分形的迭代算法、分形的逃逸时间算法<sup>[2]</sup>等。除去借助基本的分形理论进行图形绘制，Cantor 集、Koch 曲线、Sierpinski 集、Julia 集、Mandelbrot 集、牛顿法迭代分形、L-system 分形等都是经典的分形图形。

## 3. 方法描述

Koch 曲线是典型的分形曲线，其构造过程是通过反复使用生成元的相似图形，来取代每一直线段，其构造过程也决定了计算机上绘制该曲线的递归算法，即函数自己调用自己的过程<sup>[3]</sup>。

具体来说，曲线的生成元为图 3.1 所展示的图形。曲线由第一折线段反复迭代成缩小比例为  $1/3$  的生成元而成。只要约定好迭代层数，之后反复迭代，就可以得到期望的曲线。



图 3.1 Koch 曲线生成元

## 4. 实验设置

此次实验是使用 C++ OpenGL 库进行分形图形绘制。实验环境主要需要配置 OpenGL 相关的实验环境。

## 4.1 GLAD

GLAD 是继 GL3W, GLEW 之后, 当前最新的用来访问 OpenGL 规范接口的第三方库。简单来说, GLAD 主要用来调用 OpenGL 的各种方法。是使用 OpenGL 进行图形绘制的基本依赖。

## 4.2 GLFW

GLFW (Graphics Library Framework 图形库框架) 是配合 OpenGL 使用的轻量级工具程序库。主要用于创建并管理窗口和 OpenGL 上下文, 同时提供了处理手柄、键盘、鼠标输入的功能。

## 4.3 GLM

GLM (OpenGL Mathematics) 是一个专门针对 OpenGL 和图形学的数学库, 旨在提供各种数学函数和数学结构, 以便于在图形编程中进行数学计算。

## 4.4 GLUT

GLUT (OpenGL Utility Toolkit) 是一个用于创建窗口和处理用户输入的库, 旨在简化 OpenGL 应用程序的开发过程。它提供了一组简单易用的函数。通过 GLUT 开发者可以快速搭建起一个基本的 OpenGL 应用程序框架而无需关注底层的窗口管理和时间处理细节。

# 5. 实验过程与结果

此次实验中主要使用本文第四部分所描述的环境进行基于 Koch 曲线的分形图形绘制。5.1 主要叙述实验过程与部分代码实现。5.2 主要进行实验结果的展示。

## 5.1 实验过程概要

根据第三部分对于 Koch 曲线原理的说明, 实验首先需要进行的是定义基本元。实验中首先绘制一个正三角形, 如图 5.1。



图 5.1 实验初始图形

在此次实验中，绘制分形的基本方法是使用递归算法。在计算机程序设计中，递归是指一个过程直接或间接地调用其自身的一种算法[31]。此次实验中使用直接调用自身的方法设计递归算法，如图 5.2。绘制逻辑是：分别在正三角形的三条边上进行基于Koch 曲线方法的绘制。方法 dividLine 中使用Depth 参数进行递归深度的控制，如图 5.1 所示。

```
void dividLine(aPoint a, aPoint b, int Depth)
{
    if (Depth == 0)
    {
        // 组成顶点组
        float vertices[] = {
            a.x, a.y, a.z, 1.0f, 1.0f, 1.0f,
            b.x, b.y, b.z, 0.0f, 0.0f, 0.0f,
        };
        glBufferData(GL_ARRAY_BUFFER, sizeof(vertices), &vertices, GL_DYNAMIC_DRAW);
        glDrawArrays(GL_LINES, 0, 2);
    }
    else
    {
        aPoint v1 = mix(a, b, 1.0f / 3.0f);
        aPoint v3 = mix(a, b, 2.0f / 3.0f);
        aPoint v2 = calculatev2(v1, v3);
        // 递归生成图形
        dividLine(a, v1, Depth - 1);
        dividLine(v1, v2, Depth - 1);
        dividLine(v2, v3, Depth - 1);
        dividLine(v3, b, Depth - 1);
    }
}
```

图 5.2 分形图形绘制的递归算法

## 5.2 实验结果

此次实验中，一共有深度为 0，1，5，10 四幅图片被绘制，如图 5.3 所示。

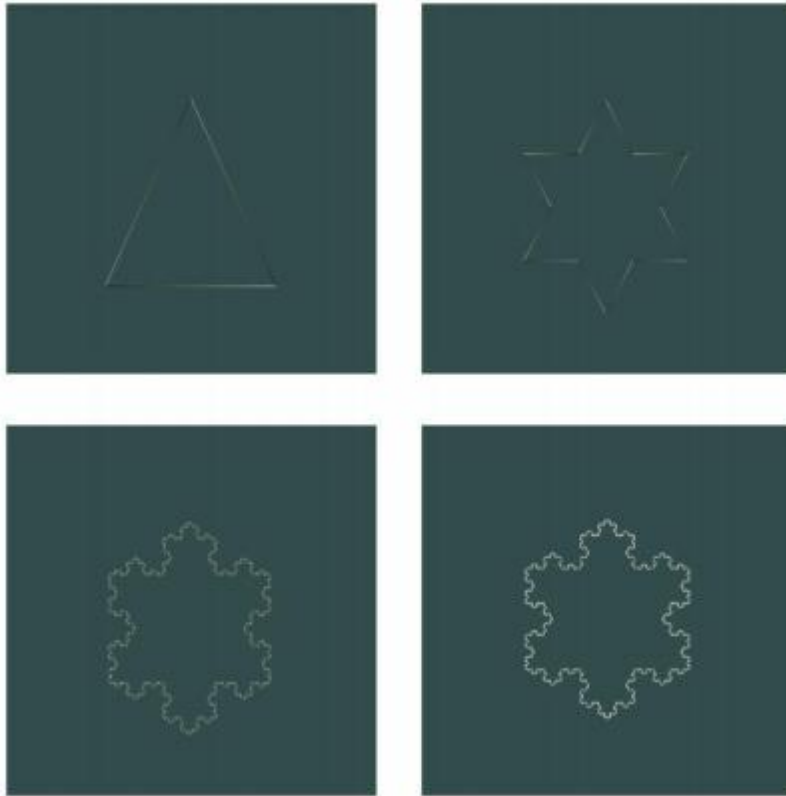


图 5.3 （左上）深度为 0 的分形绘制（右上）深度为 1 的分形图绘制（左下）深度为 5 的分形图绘制（右下）深度为 10 的分形图绘制

## 6. 结论

根据第二小节对Koch 曲线原理的介绍可知，分形图形的绘制是通过初始图形中的每一条线段进行。此次实验中所设置的初始图形为正三角形，如图 5.3 左上。深度为 1 时展现了Koch 图形绘制的基本原理。如图 5.3 右上所示，图形在初始正三角行的基础上，在每一条边的基础上添加了基本元的相似图形，最终形成一个类六芒星的图形。由于原始图形是一个正多边形，当递归深度加深时，图形逐渐趋于雪花形状，如图 5.3 左下与图 5.3 右下所示。由于计算机硬件性能限制，深度为 10 的图形绘制并不清晰，且绘制后性能出现明显下降。

## 参考文献

- [1]郭冬梅、孙兰凤等. 分形几何在计算机图形学中的应用. 机械设计, 2001 (2), 28~30
- [2]徐淑平、李春明. 分形图的生成算法研究. 微机发展, 2005, 15 (9): 4~6
- [3]Mandelbrot B B. The Fractal Geometry of Nature. San Francisco: Freeman, 1982