

Mandelbrot 集可视化项目报告

宋嘉民

2025 年 3 月 26 日

摘要

本报告详细介绍了 Mandelbrot 集的数学理论和计算方法，以及基于 C++ 实现的 Mandelbrot 集可视化项目。该项目支持多种输出格式、丰富的颜色映射和动态缩放效果。此外，项目还利用 CUDA 进行 GPU 加速计算，显著提高了生成大型高精度分形图像的效率。报告将从数学原理、算法设计、项目架构等方面进行阐述。

1 Mandelbrot 集的数学理论

1.1 复数与复平面

复数是形如 $z = a + bi$ 的数，其中 a, b 是实数， i 是虚数单位，满足 $i^2 = -1$ 。复数可以在复平面上表示，横坐标代表实部 a ，纵坐标代表虚部 b 。复数的模定义为 $|z| = \sqrt{a^2 + b^2}$ ，表示该点到原点的距离。

1.2 Mandelbrot 集的定义

Mandelbrot 集是复平面上的点 c 的集合，满足迭代序列 $z_{n+1} = z_n^2 + c$ 在初始值 $z_0 = 0$ 的条件下保持有界（不发散到无穷大）。

用数学语言表示，Mandelbrot 集 M 定义为：

$$M = \{c \in \mathbb{C} : \lim_{n \rightarrow \infty} |z_n| \neq \infty\} \quad (1)$$

1.3 收敛性与边界

如果对于某个复数 c ，存在常数 R 使得对所有的 n 都有 $|z_n| \leq R$ ，则 c 属于 Mandelbrot 集。可以证明，如果某次迭代中 $|z_n| > 2$ ，则序列一定

发散，即该点不属于 Mandelbrot 集。这个重要结论为数值计算提供了有效的判停条件。

1.4 迭代深度与边界结构

对于不在 Mandelbrot 集内的点，我们关注它经过多少次迭代后首次满足 $|z_n| > 2$ 。这个迭代次数称为“逃逸时间”，它反映了该点与 Mandelbrot 集的“距离”。逃逸时间越大，点越接近 Mandelbrot 集的边界。

Mandelbrot 集的边界呈现出极其复杂的分形结构，在任意尺度下观察都能发现相似但不完全相同的精细结构。这种无限复杂性是分形几何的典型特征，也是 Mandelbrot 集视觉吸引力的来源。

2 Mandelbrot 集的计算方法

2.1 基本算法

计算某点是否属于 Mandelbrot 集的基本算法如下：

Algorithm 1 计算点 c 的迭代次数

```
1: procedure COMPUTEITERATIONS( $c, maxIterations$ )
2:    $z \leftarrow 0$ 
3:    $iterations \leftarrow 0$ 
4:   while  $|z| \leq 2$  AND  $iterations < maxIterations$  do
5:      $z \leftarrow z^2 + c$ 
6:      $iterations \leftarrow iterations + 1$ 
7:   end while
8:   return  $iterations$ 
9: end procedure
```

在实际计算中，我们无法进行无限次迭代，因此设定一个最大迭代次数作为截止条件。如果达到最大迭代次数仍未发散，我们认为该点属于或非常接近 Mandelbrot 集。

3 项目结构与功能

3.1 系统架构

项目分为三个主要模块：

- **MandelbrotSet** 类：负责核心计算，包括 CPU 和 GPU 实现
- **Image** 类：处理图像生成和颜色映射
- **主程序**：提供命令行接口和用户交互

3.2 支持的功能

本项目支持以下功能：

- 生成 PPM 和 PNG 格式的静态图像
- 多种颜色映射方案（HSV 平滑渐变和正弦波周期变化）
- 通过 CUDA 加速的 GPU 计算
- 生成展示局部细节的缩放动画

3.3 命令行接口

系统提供了友好的命令行接口：

```
1 // 基本使用
2 ./mandelbrot --basic           // 生成基本 PPM 图像
3 ./mandelbrot --png             // 生成正弦波颜色的 PNG 图像
4 ./mandelbrot --png s          // 生成 HSV 平滑颜色的 PNG 图像
5 ./mandelbrot --zoom            // 生成缩放动画
6 ./mandelbrot --cuda --png      // 使用 CUDA 加速生成 PNG 图像
```

4 结果展示

4.1 视觉效果

不同的颜色映射方案产生了不同的视觉效果：

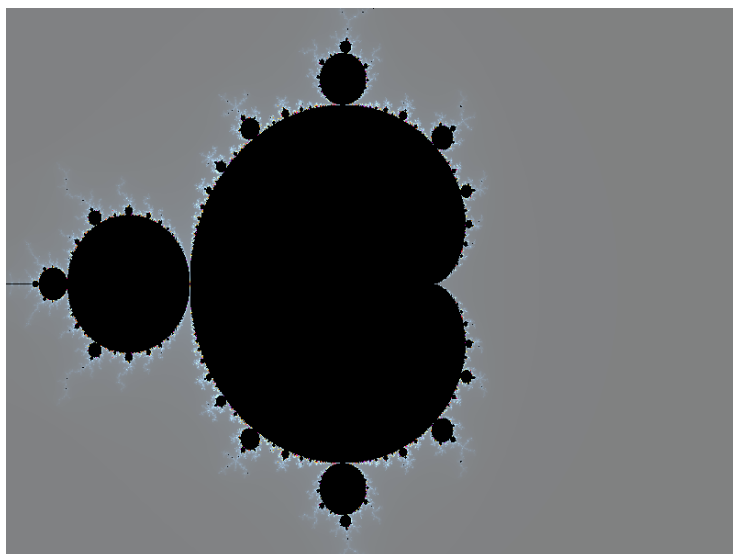


图 1: 生成的 Mandelbrot 集图像

- **基本 PPM 输出**: 使用 HSV 颜色空间, 提供连续的色彩变化
- **正弦波映射**: 产生周期性颜色变化, 突出不同”层次”的边界
- **HSV 平滑映射**: 创建彩虹般的平滑渐变, 适合展示细节结构

5 结论与展望

本项目成功实现了 Mandelbrot 集的高效计算和多样化可视化。通过深入研究 Mandelbrot 集的数学原理, 我们开发了能够生成静态图像和动态缩放动画的系统, 并利用 GPU 并行计算提高了性能。

项目的主要结果包括:

- 多种颜色映射技术, 提供丰富的视觉效果
- CUDA 加速的高效计算实现
- 支持多种输出格式和缩放动画