

GaussianImage 方法的具体思路和流程

以下是 **GaussianImage** 方法的具体思路和流程的详细分解：

1. 方法思路

GaussianImage 的核心思路是用 **2D 高斯点阵列** 表示和压缩图像，以显著降低存储复杂度和计算负担，同时提高渲染速度。

1.1 动机

传统图像表示和压缩方法（如 JPEG、神经隐式表示 INRs）存在以下局限：

- 高存储需求：** 对高分辨率图像的代表需要大量参数。
- 计算复杂度高：** 尤其是 INRs，需要大规模神经网络才能捕获图像细节。
- 渲染速度慢：** 这些方法通常在解码阶段无法满足实时需求。

GaussianImage 借鉴了 **3D Gaussian Splatting** 方法，但专注于 2D 图像，提出了更加高效的 **2D 高斯拟合与累积求和光栅化** 方法。

2. 方法流程

GaussianImage 的实现流程可以分为以下四个主要阶段：

2.1 2D 高斯点形成

目的

将图像中的信息拟合为一组 2D 高斯点，每个高斯点包含以下属性：

- 位置 (μ)：高斯中心的坐标。
- 协方差矩阵 (Σ)：描述高斯的形状和方向性。
- 颜色系数 (c) 和 不透明度 (o)：定义高斯点的颜色和强度。

方法

高斯点的协方差矩阵 (Σ) 通常是正定的，为了便于优化和控制其有效性，采用分解形式表示：

1. Cholesky 分解：

$$\Sigma = LL^T \quad (1)$$

其中 L 是下三角矩阵。

2. 旋转-缩放分解 (Rotation-Scaling, RS) :

$$\Sigma = RSS^T R^T \quad (2)$$

R 是旋转矩阵， S 是缩放矩阵。

结果

相比 3D 高斯点（59 个参数），2D 高斯点只需 **8 个参数**，大幅降低存储需求。

2.2 基于累积求和的光栅化

目的

将 2D 高斯点映射到图像平面，为每个像素计算其颜色值。

创新点

区别于传统的透明度叠加（alpha blending），提出了 **累积求和（Accumulated Summation）** 方法：

- 不再依赖高斯点的排序，从而简化计算。
- 完全利用覆盖像素的所有高斯点，提高拟合精度。

公式

像素 i 的颜色值计算公式为：

$$C_i = \sum_{n \in N} c'_n \cdot \exp(-\sigma_n), \quad \sigma_n = \frac{1}{2} d_n^T \Sigma^{-1} d_n \quad (3)$$

- c'_n 是高斯点的颜色系数与不透明度的加权结果。
- σ_n 是高斯点到像素的距离，通过协方差矩阵调节。

优点

- 计算效率高：去除了透明度的复杂累积计算。
 - 结果鲁棒：不受高斯点顺序的影响。
-

2.3 压缩流水线

目的

在拟合后，通过量化技术进一步压缩图像数据。

步骤

1. 图像拟合：使用高斯点拟合图像。
2. 量化：
 - 位置量化：使用 16 位浮点数保留位置参数。
 - 协方差矩阵量化：通过非对称量化（asymmetric quantization），将协方差矩阵的分解参数压缩为 b -bit 整数。

- 颜色量化：采用 残差矢量量化（Residual Vector Quantization, RVQ），逐步编码颜色信息。

3. 部分回退编码：将部分数据编码为初始位，再对剩余部分应用位回退压缩，进一步减少码率。

公式示例

协方差矩阵分解参数的量化：

$$\hat{l}_{ni} = \text{round} \left(\frac{l_{ni} - \beta_i}{\gamma_i} \right), \quad l_{ni} = \hat{l}_{ni} \cdot \gamma_i + \beta_i \quad (4)$$

- β_i 和 γ_i 是偏移和缩放因子。
-

2.4 训练与优化

目的

学习高斯点的参数，使得拟合的图像与原始图像的误差最小。

方法

1. 损失函数：

$$L = L_{\text{rec}} + \lambda L_c \quad (5)$$

- L_{rec} ：重建损失（如 L_2 损失）。

- L_c : 用于量化代码本优化的承诺损失。

2. 优化器：使用 Adan 优化器，初始学习率 10^{-3} 。

3. 性能分析

- 速度：

- 解码速度达 2000 FPS，适合实时应用。

- 压缩率：

- 使用部分回退编码后，与最先进方法（如 COIN++）相比，具有竞争力的压缩率。

- 存储效率：

- 每个高斯点仅需 8 个参数，压缩比达 $7.375\times$ （相较于 3D 高斯）。
-

总结

GaussianImage 提供了一种高效、轻量的图像表示与压缩方法，其核心是通过 2D 高斯拟合和累积求和光栅化，大幅降低计算复杂度，同时保持高质量的图像重建。