

# MVSNeRF

## MVSNeRF: Fast Generalizable Radiance Field Reconstruction from Multi-View Stereo

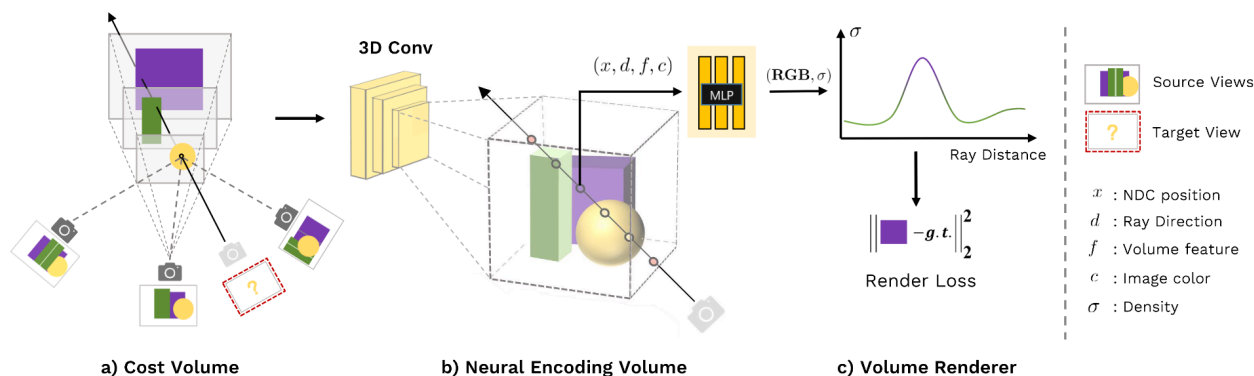


Figure 2. Overview of MVSNeRF. Our framework first constructs a cost volume (a) by warping 2D image features onto a plane sweep. We then apply 3D CNN to reconstruct a neural encoding volume with per-voxel neural features (b). We use an MLP to regress volume density and RGB radiance at an arbitrary location using features interpolated from the encoding volume. These volume properties are used by differentiable ray marching for final rendering (c).

本文主要介绍了 **MVSNeRF** 方法的具体流程和每个公式的思路。

## 1. 问题背景与目标

传统的神经辐射场（NeRF）方法需要长时间优化每个场景，这在实际应用中非常耗时。**MVSNeRF** 的目标是提供一种高效的、具备良好泛化能力的辐射场重建方法，能够在少量视图（如 3 张图像）下快速生成辐射场，并进行新视角合成。

## 2. 辐射场重建概述

MVSNeRF 的核心思想是通过多视角立体（MVS）的 **代价体积（Cost Volume）** 构建来辅助几何信息推理，并结合神经渲染实现辐射场重建。

# 辐射场函数

MVSNeRF 将辐射场定义为：

$$\sigma, r = MVSNeRF(x, d; I_i, \Phi_i) \tag{1}$$

- $x$ : 3D 空间中的位置
  - $d$ : 视角方向
  - $\sigma$ : 体积密度
  - $r$ : 方向相关的 RGB 辐射
- 

## 3. 代价体积构建

---

MVSNeRF 使用代价体积来帮助理解场景的几何结构，具体步骤如下：

### 图像特征提取

使用 2D CNN 提取图像特征：

$$F_i = T(I_i) \tag{2}$$

- $F_i$ : 输入图像  $I_i$  的特征图

### 特征图映射

根据相机参数，将每张图像的特征图映射到参考视角的不同深度平面：

$$F_{i,z}(u, v) = F_i(H_i(z)[u, v, 1]^T) \tag{3}$$

$$\mathcal{H}_i(z) = K_i \cdot (R_i \cdot R_1^T + \frac{(t_1 - t_i) \cdot n_1^T}{z}) \cdot K_1^{-1} \quad (3)$$

这个公式表示如何将每个视角的特征图  $F_i$  映射到一个参考视角的深度平面  $z$ ，用于构建代价体积。公式中的符号定义如下：

- $F_{i,z}(u, v)$ : 表示视角  $i$  在深度平面  $z$  上投影到参考视角后的位置  $(u, v)$  的特征值。
- $F_i$ : 视角  $i$  的特征图。
- $H_i(z)$ : 视角  $i$  在深度  $z$  处的齐次变换矩阵，用于将图像坐标投影到 3D 空间再投影回参考视角。
- $[u, v, 1]^T$ : 参考视角图像中的像素坐标，用齐次坐标表示。

这个公式的作用是将不同视角的特征图通过深度变换  $H_i(z)$  映射到参考视角的指定深度平面上。对于每一个深度平面  $z$ ，可以得到一个映射后的特征图  $F_{i,z}$ ，从而用于构建代价体积。这种映射有助于在不同视角间对齐特征，从而聚合来自不同视角的信息，用于推测场景的几何结构。

## 代价体积

计算代价体积的每个体素的特征向量：

$$P(u, v, z) = \text{Var}(F_{i,z}(u, v)) \quad (4)$$

- **Var** 表示不同视角下特征的方差，用于捕捉几何和外观变化。

## 4. 辐射场的神经编码体积

构建的代价体积传递给 3D CNN，生成神经编码体积：

$$S = B(P) \quad (5)$$

- $S$ : 神经编码体积，包含每个体素的场景几何和外观信息。
- 使用 3D UNet 结构，通过上采样和跳跃连接生成有意义的场景编码。

## 5. 体积属性的回归

使用 MLP 从神经编码体积中回归体积密度和辐射：

$$\sigma, r = A(x, d, f, c), \quad f = S(x) \quad (6)$$

- $f$ : 在  $S$  中三线性插值得到的特征
- $c$ : 从输入图像中获取的像素颜色，用于增强高频细节

## 6. 体积渲染与端到端训练

使用体积渲染方程计算每个像素的颜色：

$$c_t = \sum_k \tau_k (1 - \exp(-\sigma_k)) r_k, \quad \tau_k = \exp\left(-\sum_{j=1}^{k-1} \sigma_j\right) \quad (7)$$

- 通过可微分的光线行进实现新视角图像的生成。

训练使用 L2 渲染损失：

$$L = \|c_t - \tilde{c}_t\|_2^2 \quad (8)$$

- $\tilde{c}_t$ : 来自目标视角的真实颜色。

## 7. 神经编码体积的优化

当有密集图像输入时，可以对编码体积和 MLP 进行快速微调优化，以进一步提高渲染质量。

---

## 总结

---

MVSNeRF 结合了多视角立体和神经渲染，实现了快速且泛化性强的辐射场重建，并能够在少量输入图像的情况下生成高质量的新视角合成。