# 基于 Transformer 的 NeRF：面向前向场景表示的探索

（作者信息略）

**Abstract**

Neural Radiance Fields（NeRF）通过对三维空间中的辐射场进行隐式建模，实现了高质量新视角合成。然而，经典 NeRF 的训练属于“场景专属的逆向优化”：针对单一场景迭代优化一个 MLP，并在体渲染积分中进行大量采样与反向传播，导致训练代价高、泛化能力弱。本文探索一种“前向（forward）预测”的替代范式：利用视觉 Transformer（以 VGGT 为代表的多视角聚合架构）从多视角图像直接预测可用于体渲染的场景表示，从而减少/避免对每个场景的长时间优化。我们提出两条实现路径：（1）预测 NeRF 的 MLP 参数（超网络/权重生成）；（2）预测结构化的低维网格表示（Tri-plane/Hybrid Grid）并配合全局共享的轻量解码器（Tiny MLP）。此外，针对动态场景，我们提出一种在特征空间进行“语义一致”的时间插值方法：不显式对时间维度做回归，而是对跨时刻 token 进行匹配与插值，从而生成中间时刻的稳定表示。本文给出方法设计、训练目标与工程实现中的关键 trick，并讨论该前向范式在泛化、语义一致性与动态生成方面的意义与局限。

关键词：NeRF；Transformer；前向预测；Tri-plane；动态场景；特征插值

## 1 引言

NeRF 通过学习连续函数

$$f_\theta : (\mathbf{x}, \mathbf{d}) \mapsto (\mathbf{c}, \sigma),$$

将空间坐标 $\mathbf{x} \in \mathbb{R}^3$ 与视角方向 $\mathbf{d} \in \mathbb{R}^3$ 映射为颜色 $\mathbf{c} \in \mathbb{R}^3$ 与体密度 $\sigma \in \mathbb{R}$，再通过体渲染积分生成像素颜色。经典训练流程对每个场景都需进行长时间的 per-scene optimization，并伴随高昂的射线采样与体渲染计算。

本文探索：能否让多视角 **Transformer** 直接输出可渲染的辐射场表示，从而将 **NeRF** 从“逆向优化”转向“前向预测”？

## 2 背景与相关工作（简述）

### 2.1 NeRF 与体渲染

对相机射线 $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$ 采样 $N$ 个点 $\{t_i\}$，得到 $(\mathbf{c}_i, \sigma_i)$，像素颜色通过离散体渲染近似：

$$\hat{\mathbf{C}}(\mathbf{r}) = \sum_{i=1}^{N} T_i \alpha_i \mathbf{c}_i, \quad \alpha_i = 1 - \exp(-\sigma_i \Delta_i), \quad T_i = \prod_{j<i} (1 - \alpha_j).$$

### 2.2 VGGT：面向多任务 3D 推理的前向网络（CVPR 2025 Best Paper）

VGGT 是一类前向（feed-forward）多视角网络：给定一张、少量或大量视图，可直接推理相机内外参、深度图、点图、稠密点云以及 3D 点轨迹等关键 3D 属性。其结构可概括为“Aggregator + Head”范式：

- **Token** 化与相机 **token**：用 DINO 类骨干将输入图像 patchify 为 tokens，并拼接 camera tokens。
- 交替注意力融合：交替使用 frame-wise self-attention 与 global self-attention，实现单帧细节与跨视角一致性的统一。

- 任务头解耦输出：camera head 输出相机参数；DPT head 输出深度/点图/稠密特征等。

本文借鉴该范式，将 head 的输出目标从传统几何要素扩展为可用于 **NeRF** 体渲染查询的场景表示。

# 3 Novel NeRF：从逆向场景优化到 once-for-all 前向场景表征

传统 NeRF 是逆向范式：对每个新场景进行大量迭代优化才能获得可渲染的场景参数，带来效率与部署泛化两方面的瓶颈。本文探索 once-for-all 的前向生成：给定多视角图像，模型直接输出可渲染的场景表示 $\mathcal{S}$，从而在“获得场景表示”阶段做到一次推理。

从表示角度看，$\mathcal{S}$ 也可视为对场景的压缩表示：可存储、可传输，并可通过渲染得到任意视角投影，具备发展为通用 3D 表示的潜力。

## 3.1 两条主体路线

- 主体 **A**：**T-NeRF**（本文展开）：利用 VGGT 的多视角聚合能力，前向预测辐射场表示。
- 主体 **B**：另一主体：由于时间与资源限制，本报告暂留空。

## 3.2 T-NeRF 的两种实现方案与训练（基于 NeRF-MAE 的 RGB/Sigma 场监督）

T-NeRF 继承 VGGT 的 Aggregator + Head 结构：Aggregator 输出全局 token 表征 $\mathbf{Z}$，Radiance Head 预测可渲染表示 $\mathcal{S}$。

方案一：直接预测 **NeRF** 的 **MLP** 参数（超网络）

$$\theta = g_\phi(\mathbf{Z}), \qquad (\mathbf{c}, \sigma) = f_\theta(\gamma(\mathbf{x}), \gamma(\mathbf{d})).$$

该方案输出维度较大，训练更不稳定。

方案二：预测 **Tri-plane/Hybrid Grid** + 共享 **Tiny MLP**（推荐） head 预测三个二维特征平面 $\mathbf{F}_{xy}, \mathbf{F}_{xz}, \mathbf{F}_{yz}$，对查询点做双线性插值并拼接输入共享解码器：

$$(\mathbf{c}, \sigma) = h_\psi([\mathbf{f}_{xy}(\mathbf{x}), \mathbf{f}_{xz}(\mathbf{x}), \mathbf{f}_{yz}(\mathbf{x}), \gamma(\mathbf{x}), \gamma(\mathbf{d})]).$$

该方案输出更低维、归纳偏置更强，通常更稳定。受限于时间与算力，本项目完成主要代码与小规模训练验证；该路线被认为是 promising but under-trained。

训练数据与损失 本文使用 NeRF-MAE 预训练数据（RGB/Sigma 场数据）：在三维网格上提供每个 3D 点的 $\mathbf{c}^*(\mathbf{x})$ 与 $\sigma^*(\mathbf{x})$。因此可采用点监督（field supervision）：采样 $\{\mathbf{x}_i\}$，回归预测 $(\mathbf{c}_i, \sigma_i)$：

$$\mathcal{L}_{field} = \lambda_c \cdot \frac{1}{N} \sum_i \|\mathbf{c}_i - \mathbf{c}_i^*\|_1 + \lambda_\sigma \cdot \frac{1}{N} \sum_i \|\sigma_i - \sigma_i^*\|_1.$$

方案二中对 Tri-plane/grid latent 加 $L_2$ 正则：

$$\mathcal{L}_{grid} = \|\mathbf{F}_{xy}\|_2^2 + \|\mathbf{F}_{xz}\|_2^2 + \|\mathbf{F}_{yz}\|_2^2,$$

最终：$\mathcal{L} = \mathcal{L}_{field} + \lambda_{grid}\mathcal{L}_{grid}$（另可选叠加渲染重建项）。

训练 **trick**（节选）

- 先冻结 Aggregator，仅训练新增 head；必要时再以小学习率解冻后层。
- 使用 AdamW/weight decay 对 head 做适度正则，缓解输出塌缩。
- 对密度 $\sigma$ 使用非负参数化（如 softplus），提升数值稳定性。

此外，在 dynamic NeRF 的 deformation 基线实现中，我们在关键 MLP 模块中加入 **LayerNorm**，以抑制训练早期的梯度爆炸与数值不稳定；若不引入归一化/正则化层，通常需要更保守学习率或更多训练迭代才能稳定收敛。

# 4 Dynamic NeRF：朴素基线与特征空间插值

## 4.1 朴素 Dynamic NeRF（两种方案）

两类常见动态 NeRF 的目标是学习带时间的辐射场：

$$f_\theta : (\mathbf{x}, \mathbf{d}, t) \mapsto (\mathbf{c}, \sigma).$$

方案 **A**：时间嵌入直接注入（**Straightforward / Time-as-Input**）

$$(\mathbf{c}, \sigma) = f_\theta([\gamma(\mathbf{x}), \gamma(\mathbf{d}), \gamma(t)]).$$

方案 **B**：静态场 + 动态残差（**Deformation / Residual Dynamics**）

$$(\mathbf{c}_s, \sigma_s) = f_{\theta_s}(\mathbf{x}, \mathbf{d}), \quad (\Delta\mathbf{c}, \Delta\sigma) = f_{\theta_d}(\mathbf{x}, \mathbf{d}, t),$$

$$(\mathbf{c}, \sigma) = (\mathbf{c}_s + \Delta\mathbf{c},\ \sigma_s + \Delta\sigma).$$

实现上，为了提升训练稳定性并加速收敛，我们在 deformation 的动态分支（或残差 MLP）中额外加入 **LayerNorm**。

## 4.2 量化结果（8 个动态场景）

我们在 D-NeRF 的 8 个动态场景上对两种朴素方案进行量化评估，报告 PSNR/SSIM 及其标准差。

**Deformation** 方案（静态场 + 动态残差）

| Scene | PSNR (dB) | SSIM | PSNR Std | SSIM Std |
|---|---|---|---|---|
| bouncingballs | 28.70 | 0.9602 | 3.31 | 0.0261 |
| hellwarrior | 24.82 | 0.9487 | 2.15 | 0.0145 |
| hook | 29.17 | 0.9634 | 3.22 | 0.0228 |
| jumpingjacks | 32.39 | 0.9742 | 4.09 | 0.0327 |
| lego | 21.83 | 0.8481 | 1.37 | 0.0422 |
| mutant | 31.25 | 0.9736 | 3.23 | 0.0147 |
| standup | 33.31 | 0.9804 | 3.61 | 0.0114 |
| trex | 31.38 | 0.9726 | 2.85 | 0.0247 |
| Average | 29.11 | 0.9527 | 2.98 | 0.0236 |

**Straightforward** 方案（时间嵌入直接注入）

| Scene | PSNR (dB) | SSIM | PSNR Std | SSIM Std |
|---|---|---|---|---|
| bouncingballs | 31.64 | 0.9702 | 1.44 | 0.0065 |
| hellwarrior | 21.77 | 0.9153 | 0.91 | 0.0115 |
| hook | 24.29 | 0.9068 | 1.06 | 0.0145 |
| jumpingjacks | 27.58 | 0.9467 | 1.90 | 0.0173 |
| lego | 23.54 | 0.8581 | 0.74 | 0.0198 |
| mutant | 25.96 | 0.9283 | 1.16 | 0.0128 |
| standup | 26.20 | 0.9452 | 1.72 | 0.0157 |
| trex | 27.12 | 0.9338 | 1.15 | 0.0143 |
| Average | 26.01 | 0.9255 | 1.26 | 0.0141 |

### 4.3 结果分析（对应 **report.md** 的评论）

- 整体对比：Deformation 的平均 PSNR/SSIM 高于 Straightforward（29.11 vs 26.01 dB；0.9527 vs 0.9255），体现出“静态 + 动态残差”分解能降低优化难度并提升质量。
- 稳定性：Deformation 的 PSNR 标准差更大（2.98 vs 1.26），说明其在不同帧/视角的波动更大，但峰值质量更高；Straightforward 整体更稳定，但多数场景仍落后。
- 场景特异性：Deformation 在 8 个场景中取得 6 个场景的更优 PSNR/SSIM；Straightforward 在 bouncingballs、lego 等场景上也能达到可比甚至更高的分数，说明其在部分“相对规则/可回归”的动态中仍具可用性。
- 共同局限：两类朴素方法本质上都是对时间变化做连续回归，缺乏语义级约束，在遮挡/快速形变区域容易出现闪烁与模糊，并且训练对超参敏感。

### 4.4 **Novel Dynamic NeRF based on VGGT**：语义一致插值

我们提出在特征空间进行“语义一致”的时间插值：不显式对时间维度做回归，而是对跨时刻 token 进行局部匹配与插值，从而生成中间时刻的稳定表示。

实现要点（简述）

1. 局部匹配（语义对齐）：在 $t_1$ 的邻域内用余弦相似度寻找 $t_0$ token 的对应。
2. 线性插值：$\mathbf{z}(t_i) = (1-\alpha)\mathbf{z}(t_0) + \alpha\mathbf{z}(t_1)$，并对 2D token 位置插值。
3. 双线性泼溅 + 权重归一化：将连续位置 token 分配到邻近网格点并归一化，避免能量漂移。

**Demo**（对应 **report.md**） dynamic 的模型代码位于 tnerf/model/dynamic_tnerf.py。我们在 VGGT demo 基础上修改 UI:将预训练参数放置在 vggt/model_weights,运行 vggt/demo_gradio.py，通过页面加载历史记录,可观察到从两端时刻到中间时刻的点云插值结果（见 novel_dynamic.png）。

## 5 局限

- 训练数据与监督形式受限：本文主要依赖 NeRF-MAE 的 RGB/Sigma 场数据进行点监督训练。该监督对“场函数拟合”较直接，但与真实应用中的图像监督仍存在 domain gap；同时数据覆盖的场景类型有限，可能影响跨域泛化能力。
- 训练规模与算力预算不足：T-NeRF（尤其是 Tri-plane + Tiny MLP）被认为是更 promising 的路线，但本项目仅完成代码框架与小规模训练验证，尚未在更大规模数据与更长训练日程下充分释放性能上限。

- 前向场景表示仍需系统评测：目前对 once-for-all 的推理速度优势、存储压缩率、跨场景泛化（OOD），以及与经典 NeRF/Instant-NGP 的全面对比仍不充分。
- 动态方案的边界条件：基于 token 的语义一致插值依赖 VGGT backbone 的特征对齐能力。当两端关键帧差异过大（大位移、强遮挡、拓扑变化）或视角变化剧烈时，局部匹配可能失败，从而影响中间时刻生成稳定性。
- 工程细节仍有优化空间：Tri-plane 分辨率限制细节上限；动态分支训练仍依赖正则化（如 LayerNorm、weight decay）与超参；更合理的采样策略与损失权重可能进一步提升收敛速度与最终质量。

## 6 结论与未来工作

本文探索了基于 Transformer 的 NeRF 前向预测范式，提出两种实现路径并给出动态 token 插值方案。未来工作包括：引入层级/哈希网格以提升细节；将渲染监督与几何监督联合；把动态插值扩展到多关键帧与长序列；在真实数据与复杂相机轨迹上系统评测泛化能力。