扩散概率场

Peiye Zhuang[1†], **Samira Abnar**[2] , **Jiatao Gu**[2] ,**Alexander G. Schwing**[3], **JoshuaM. Susskind**[2] , **MiguelÁngel Bautista**[2]1斯坦福大学 [2]Apple [3]伊利诺伊大学厄巴纳-香槟分校[1]peiye@stanford.edu [2]{abnar, jgu32, jsusskind,mbautistamartin}@apple.com [3]aschwing@illinois.edu

arXiv:2303.00165v1 [cs.CV] 1 Mar 2023

摘要

扩散概率模型已迅速成为图像、三维几何、视频及其他领域生成建模的重要方法。然而，要将扩散生成建模应用于这些领域，必须针对每个领域分别精心设计去噪网络，而且往往假设数据位于欧几里得网格上。本文介绍扩散概率场（DPF），这是一种扩散模型，可学习定义在度量空间上的连续函数分布，这类函数通常称为场。我们将扩散概率模型的表述显式扩展，以处理这种场参数化，从而能够定义一种端到端学习算法，避免像以往方法那样需要用潜向量表示场（Dupont 等，2022a；Du 等，2021）。我们通过实验证明，在使用同一个去噪网络的情况下，DPF 能有效处理二维图像和三维几何等不同模态，还能对定义在非欧度量空间上的场分布进行建模。

1 引言

扩散概率建模已迅速成为学习数据分布的核心方法，并在图像（Nichol 与 Dhariwal，2021年）、视频（Ho 等人，2022）乃至三维几何（Luo & Hu，2021）等多个领域取得了令人瞩目的实证成果。尤其是，去噪扩散概率模型（通常称为 DDPM 或扩散生成模型）（Ho 等人，2020；Nichol 与 Dhariwal，2021）及其连续时间扩展（Song 等人，2021b）都提出了比生成对抗网络（Goodfellow 等人，2014年）或基于能量的模型（Du 等人，2020）等先行方法更稳定的训练目标。此外，实证结果表明，扩散生成模型在图像域中的表现优于生成对抗网络（Dhariwal & Nichol，2021），且在训练过程中较少受到模式寻求病态的影响（Kodali 等人，2017年）。

一个扩散生成模型由三个主要组件构成：前向（或扩散）过程、反向（或推断）过程，以及去噪网络（也称为分数网络[1]，这是由于其与去噪分数匹配方法 Dickstein 等人（2015年）的等价性）。大量工作探讨了前向和反向过程的不同定义（Rissanen 等人，2022；Bansal 等人，2022；Song 等人，2021a），重点关注图像域。然而，当前扩散模型仍有两个尚未解决的注意事项。第一，通常假定数据位于离散欧几里得网格上（例外包括关于分子（Hoogeboom 等人，2022）和点云（Luo 与 Hu，2021）的工作）。第二，去噪网络会针对每个特定数据域进行大量调优，图像（Nichol 与 Dhariwal，2021）、视频（Ho 等人，2022）或几何数据（Luo 与 Hu，2021）分别采用不同的网络架构。

为了将扩散生成模型的成功推广到科学与工程中大量多样的领域，需要统一得分表述。重要的是，这种统一使我们能够在具有不同几何结构的不同数据域之间使用同一个分数网络，而无需要求数据存在于离散欧几里得网格中，或将数据投影到离散欧几里得网格中。

[†]这项工作是在 P.Z. 担任 Apple 实习生期间完成的。[1]本文交替使用“评分网络/函数”和“去噪网络/函数”这两个术语。

为此，本文提出了扩散概率场（DPF）。DPF 通过学习连续函数上的分布，朝着跨领域统一扩散生成建模这一宏大目标迈进。为实现这一点，我们从函数视角看待数据，将数据点 $\boldsymbol{x} \in \mathbb{R}^d$ 解释为函数 $f\colon\ \mathcal{M} \to \mathcal{Y}$（Dupont 等，2022a；Du 等，2021）。函数 f 将度量空间 $\mathcal{M}$ 中的元素映射到信号空间 $\mathcal{Y}$。这种数据的函数视角通常被称为场 表示（Xie 等，2022），我们也用“场”来指代此类函数。图 1 展示了这种场解释的示意图。以图像域为例可以看出，图像既可以解释为多维数组 $\boldsymbol{x}_i \in \mathbb{R}^{h \times w} \times \mathbb{R}^3$，也可以解释为场 $f\colon\ \mathbb{R}^2 \to \mathbb{R}^3$，将二维像素坐标映射到 RGB 值。这种场视角使我们能够在同一参数化下统一看似不同的数据域。例如，三维几何数据通过 f 表示：$\mathbb{R}^3 \to \mathbb{R}$，而球面信号则成为场 $f\colon\ \mathbb{S}^2 \to \mathbb{R}^d$。

为了统一不同数据域上的生成建模，场数据表示在近期的三种方法中展现出良好前景：从数据到 functa（Functa）（Dupont 等，2022a）、生成流形学习（GEM）（Du 等，2021）以及生成对抗随机过程（GASP）（Dupont 等，2022b）。前两种方法采用潜在场参数化（Park 等，2019），通过超网络（Ha 等，2017）对场网络进行参数化，该超网络以可训练的潜向量为输入。在训练过程中，针对每个场的潜向量都会在初始重构阶段进行优化（Park 等，2019）。在 Functa （ Dupont 等，2022a ）中，作者随后提出在独立的第二训练阶段学习优化后潜变量的分布，这与 Rombach 等人（2022）和 Vahdat 等人（2021）的方法类似。Du 等人（2021）在重构阶段定义了额外的潜在邻域正则化项。之后以非参数方式进行采样：从优化后的潜向量集合中随机选取一个潜向量，并将其投影到局部邻域中，然后添加高斯噪声。图 2 对 Functa（Dupont 等，2022a）、GEM（Du 等，2021）与我们的 DPF 之间的差异进行了直观总结。GASP（Dupont 等，2022b）采用生成对抗网络范式：生成器生成一个场，而判别器则对场中的离散点进行操作，并区分输入来源，即。例如，是真实数据还是生成数据。

与先前工作（Dupont 等，2022a；Du 等，2021；Dupont 等，2022b）不同，我们采用扩散生成模型 来建模函数，并提出一种单阶段 方法。这使得无需依赖初始重建阶段，也无需调整对抗博弈，即可高效进行端到端训练；我们的实证结果表明，这种方法表现出色。我们的贡献总结如下：

- 我们提出了扩散概率场（DPF），将扩散生成模型的公式扩展到场表示。
- 我们采用显式场参数化，在单阶段模型中构建场上的概率生成模型。这与近期工作（Dupont 等，2022a；Du 等，2021）不同，并通过实现端到端学习简化了训练过程。
- 我们通过实验证明，DPF 能够成功捕捉不同领域中函数的分布，例如图像、三维几何和球面数据，并且优于近期工作（Dupont 等，2022a；Du 等，2021；Dupont 等，2022b）。

2 背景：去噪扩散概率模型

去噪扩散概率模型（DDPM）属于潜变量模型这一广泛家族。关于深入的综述，读者可参阅 Everett（2013）。简而言之，为了从有限样本 $q(\boldsymbol{x}_0)$ 的经验分布中学习参数化数据分布 $p_\theta(\boldsymbol{x}_0)$，DDPM 反转了一个扩散马尔可夫链（即，前向扩散过程）：该过程通过逐步向数据 $\boldsymbol{x}_0 \sim q(\boldsymbol{x}_0)$ 添加高斯噪声，在 T 个时间步内生成潜变量 $\boldsymbol{x}_{1:T}$，如下所示：

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) := \mathcal{N}\left(\boldsymbol{x}_{t-1}; \sqrt{\bar{\alpha}_t}\boldsymbol{x}_0, (1-\bar{\alpha}_t)\mathbf{I}\right). \tag{1}$$

此处，$\bar{\alpha}_t$ 是截至时间步t、按照人工设计的调度方案计算得到的固定方差的累积乘积。Ho 等（2020）指出了两项重要观察，使 DDPM 的训练更加高效：i）式（1）采用闭式形式对前向扩散过程进行采样。ii）反转扩散

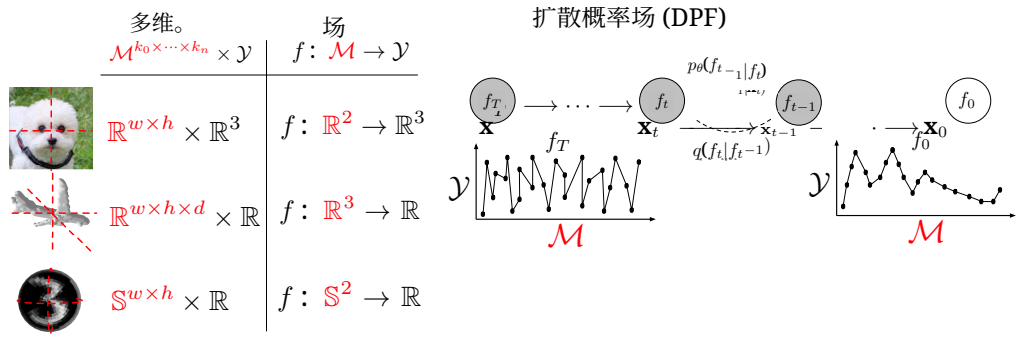


图 1：左图展示了如何将每个数据域参数化为一个场。为便于可视化，我们用 红色 表示场函数的输入（例如，一个度量空间 $\mathcal{M}$，场定义于其中）。右图展示了 DPF 的图模型；DPF 是一种用于捕捉场分布的扩散生成模型。在 DPF 中，潜变量是场 $f_{1:T}$，可对其进行连续求值。通过对数据进行场参数化，我们使用同一分数网络实现，统一了不同数据域（图像、三维形状、球面图像）上的扩散生成建模。

该过程等价于学习一系列权重共享的去噪（或分数）网络 ϵ_θ。将式 (1) 重新参数化为 $\boldsymbol{x}_t = \sqrt{\bar{\alpha}_t}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$，即可得到“简单”的 DDPM 损失

$$\mathcal{L}_\theta = \mathbb{E}_{t\sim[0,T],\boldsymbol{x}_0\sim q(\boldsymbol{x}_0),\epsilon\sim\mathcal{N}(0,\mathbf{I})}\left[\|\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t)\|^2\right], \tag{2}$$

这使得数据分布 $p_\theta(\boldsymbol{x}_0)$的学习既高效又可扩展。

在推断时，我们通过祖先采样（Ho 等人，2020）计算 $\boldsymbol{x}_0 \sim p_\theta(\boldsymbol{x}_0)$。具体而言，我们首先采样 $\boldsymbol{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$，然后迭代地应用分数网络 ϵ_θ对 $\boldsymbol{x}_T$进行去噪，从而反转扩散马尔可夫链并得到 $\boldsymbol{x}_0$。采样 $\boldsymbol{x}_{t-1} \sim p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$等价于计算以下更新

$$\boldsymbol{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{x}_t - \frac{1-\alpha_t}{\sqrt{1-\alpha_t}}\epsilon_\theta(\boldsymbol{x}_t, t)\right) + \mathbf{z}, \tag{3}$$

其中，在每个推断步骤中都会注入一个随机分量 $\boldsymbol{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$，类似于通过朗之万动力学（Welling 与 Teh，2011）进行采样。

式（2）中学习目标的核心部分是分数网络 ϵ_θ，它控制着边缘分布 $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$。值得注意的是，分数网络针对每个特定的数据域进行了大量定制。例如，在图像域中，分数网络基于包含多个自注意力模块的 UNet（Ronneberger等人，2015；Nichol 与 Dhariwal，2021）。相比之下，对于分子等三维结构，分数网络则基于图神经网络（Hoogeboom 等人，2022）。为统一分数网络在不同数据域中的设计，本文提出扩散概率场（DPF），通过将数据样本表示为场，引入一种可应用于多个领域的统一分数网络形式。

3 扩散概率场

扩散概率场（DPF）是一种能够捕捉场分布的扩散生成模型。给定场上的经验分布 $q(f_0)$，其中的场存在于一个未知的场流形上；场 $f_0 : \mathcal{M} \to \mathcal{Y}$ 将度量空间 $\mathcal{M}$ 中的元素映射到信号空间 $\mathcal{Y}$。例如，在图像域中，图像可以定义为一种将二维像素坐标映射到 RGB 值 $f_0 : \mathbb{R}^2 \to \mathbb{R}^3$ 的场。在 DPF 中，潜变量 $f_{1:T}$ 是可连续求值的场。要解决场上的扩散生成模型学习问题，

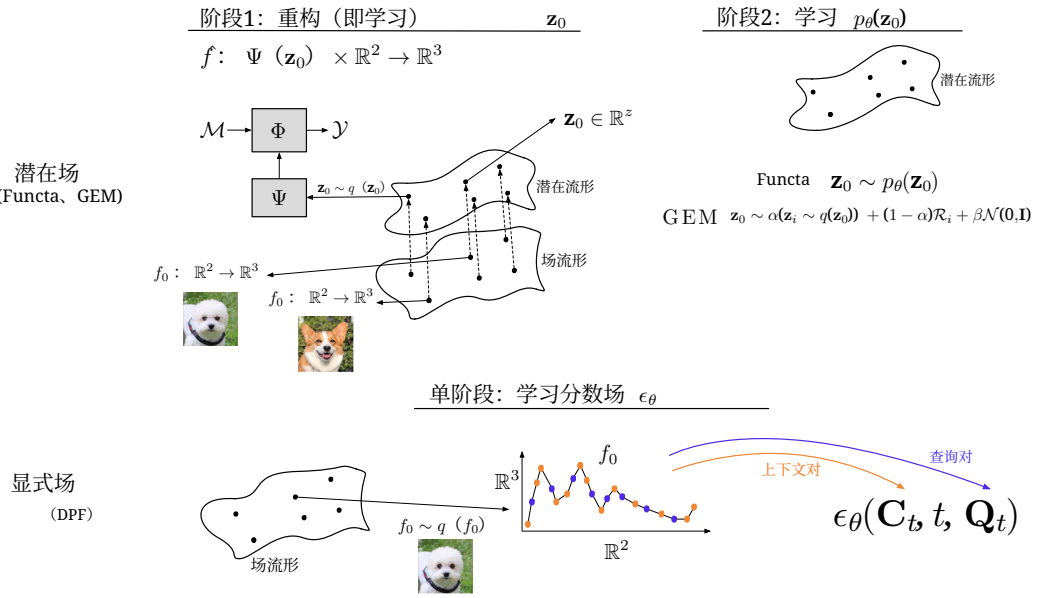


图 2：在 DPF 中，我们通过一组坐标-信号对（即上下文对）显式地参数化一个场，而不是像 Functa（Dupont 等，2022a）或 GEM（Du 等，2021）那样使用潜向量 $\boldsymbol{z}_0$。这种显式场参数化使我们能够避开以往方法中的重建训练阶段，转而直接对场的分布建模，而不是对用于编码场的潜变量分布建模。

我们需要在前向过程中，以及在评分网络和反向过程中，妥善处理场表示的无限维特性。

我们采用显式场参数化，其中一个场由一组坐标-信号对 $\{(\boldsymbol{m}_c, \boldsymbol{y}_{(c,0)})\}, \boldsymbol{m}_c \in \mathcal{M}, \boldsymbol{y}_{(c,0)} \in \mathcal{Y}$ 表征，我们将其称为上下文对。为清晰起见，我们按行堆叠上下文对，并将所得矩阵记为$\mathbf{C}_0 = [\mathbf{M}_c, \mathbf{Y}_{(c,0)}]$。这里，$\mathbf{M}_c$表示所有上下文对中的坐标部分，而$\mathbf{Y}^{(c,0)}$ 表示时间 $t = 0$处所有上下文对中的信号部分。需要注意的是，按照设计，坐标部分不依赖时间步。[2]这与 Functa（Dupont 等，2022a）或 GEM（Du 等，2021）有一个关键区别：二者都采用场的潜变量参数化，其中学习得到的场 f： $\Psi(\boldsymbol{z}_0) \times \mathcal{M} \to \mathcal{Y}$通过超网络模型 Ψ (Ha 等，2017)由潜权重向量 $\boldsymbol{z}_0$ 参数化。采用潜变量参数化会引入一个重建阶段，在该阶段中，首先优化潜变量 $\boldsymbol{z}_0$，以重构其对应的场（Park 等，2019；Du 等，2021）（*i.e.*，定义位于未知潜在流形上的潜变量经验分布 $q(\boldsymbol{z}_0)$）。随后，在第二个训练阶段中，独立地学习潜变量的先验 $p_\theta(\boldsymbol{z}_0)$（Dupont 等，2022a）。相比之下，我们的显式场参数化使我们能够构建分数场网络，从而让 DPF 直接对场的分布建模，并取得更好的性能（参见第 4 节）。图 2 展示了 Functa（Dupont 等，2022a）和 GEM（Du 等，2021）中的潜在场参数化与 DPF 中的显式场参数化之间的区别。

采用显式场参数化，我们通过扩散信号并保持坐标不变，为上下文对定义前向过程。因此，上下文对的前向过程如下：

$$\mathbf{C}_t = [\mathbf{M}_c, \mathbf{Y}_{(c,t)} = \sqrt{\bar{\alpha}_t}\mathbf{Y}_{(c,0)} + \sqrt{1-\bar{\alpha}_t}\epsilon_c], \tag{4}$$

其中 $\epsilon_c \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ 是一个大小适当的噪声向量。现在我们转而讨论如何为场构建分数网络。根据定义，分数网络需要以上下文对（*ie.*，即场参数化）作为输入，并且需要能够在 $\mathcal{M}$ 中进行连续求值，以便成为

[2]我们假设场的几何结构不会改变。

算法 1训练

重复
  $(\mathbf{C}_0, \mathbf{Q}_0) \sim$ 均匀分布$(q(f_0))$
  $t \sim$ 均匀分布$(\{1, \ldots, T\})$
  $\epsilon_c \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, $\epsilon_q \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  $\mathbf{C}_t = [\mathbf{M}_c, \sqrt{\bar{\alpha}_t}\mathbf{Y}_{(c,0)} + \sqrt{1-\bar{\alpha}_t}\epsilon_c]$
  $\mathbf{Q}_t = [\mathbf{M}_q, \sqrt{\bar{\alpha}_t}\mathbf{Y}_{(q,0)} + \sqrt{1-\bar{\alpha}_t}\epsilon_q]$
  对以下内容执行梯度下降步骤
  $\nabla_\theta \left\| \epsilon_q - \epsilon_\theta\left(\mathbf{C}_t, t, \mathbf{Q}_t\right) \right\|^2$
直至收敛

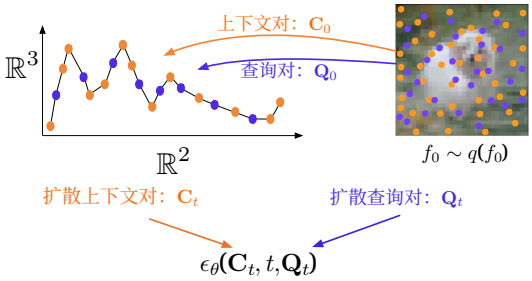


图 3：**左：** DPF训练算法。**右**：图像域中某个场的训练迭代过程示意图。定义见第 3 节。

一个场。我们通过使用 查询对 $\{\boldsymbol{m}_q, \boldsymbol{y}_{(q,0)}\}$ 来实现这一点。与上下文对相同，我们按行堆叠查询对，并将所得矩阵记为 $\mathbf{Q}_0 = [\mathbf{M}_q, \mathbf{Y}_{(q,0)}]$。注意，前向扩散过程对上下文与查询对的定义是等价的：

$$\mathbf{Q}_t = [\mathbf{M}_q, \mathbf{Y}_{(q,t)} = \sqrt{\bar{\alpha}_t}\mathbf{Y}_{(q,0)} + \sqrt{1-\bar{\alpha}_t}\epsilon_q], \tag{5}$$

其中， $\epsilon_q \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ 是一个大小适当的噪声向量。然而，底层场仅由上下文对定义，而查询对仅作为评估分数网络的点。由此得到的分数场 模型表述如下：$\hat{\epsilon}_q = \epsilon_\theta(\mathbf{C}_t, t, \mathbf{Q}_t)$。

分数场模型的设计空间涵盖了所有能够处理集合形式数据的架构，例如 Transformer 或多层感知机。尤其是，高效 Transformer 架构为处理大量上下文与查询对提供了直接的方法，还能通过注意力机制让查询对与上下文对良好交互。我们的大多数实验采用 PerceiverIO（Jaegle 等，2022），这是一种高效的 Transformer编码器-解码器架构；详情参见图 11 和 B 节。此外，我们在 D 节展示，其他架构（例如原始Transformer编码器（Vaswani 等，2017）和 MLP混合器（Tolstikhin 等，2021））同样可行，能够提供非常灵活的设计空间，同时不牺牲 DPF 表述的通用性。

利用显式场表征和分数场网络，我们得到算法 1 和算法 2 中的训练与推理过程，并附有图像场表示的示例图。训练时，我们从 $f_0 \sim$ 均匀分布$(q(f_0))$ 中均匀采样上下文与查询对，并仅使用公式 (4) 和公式 (5) 中的前向过程对其信号进行扰动。然后，我们训练分数场网络 ϵ_θ，使其在给定上下文对的条件下对查询对中的信号进行去噪（使用 PerceiverIO实现的可视化见图 11）。采样时，为生成场 $f_0 \sim p_\theta(f_0)$，我们首先定义将在其上评估该场的查询对 $\mathbf{Q}_T = [\mathbf{M}_q, \mathbf{Y}_{(q,T)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})]$。请注意，采样期间评估分数场的点数必须固定（例如，若要生成一幅包含 $32 \times 32 = 1024$ 个像素的图像，我们在时间 $t = T$ 定义 1024 个查询对 $\mathbf{Q}_T$）。接着，将查询对的随机子集作为上下文对。我们利用上下文对对查询对进行去噪，并遵循原始DDPM（Ho 等人，2020）中的祖先采样方法。[3] 请注意，在推理期间，上下文对和查询对的坐标保持不变，只有相应的信号值会发生变化。采样结果由 $\mathbf{Q}_0$ 给出，它是在坐标$\mathbf{M}_q$处对 f_0 进行评估的结果。

4 实验结果

我们展示了多个领域的结果：二维图像数据、三维几何数据和球面数据。在所有领域中，我们使用相同的评分网络架构。我们只调整度量空间 $\mathcal{M}$ 和信号空间 $\mathcal{Y}$ 的维度，以及网络容量（即，层数、隐藏

[3]像 DDIM（Song 等人，2021a）这样更高效的采样方法可以轻松适配到 DPF。

算法2采样

1: $\mathbf{Q}_T = [\mathbf{M}_q,\ \mathbf{Y}_{(q,t)} \sim \mathcal{N}(\mathbf{0}_q,\ \mathbf{I}_q)\ ]$
2: $\mathbf{C}_T \subseteq \mathbf{Q}_T$ ▷ 随机子集
3: **对于** $t = T, \ldots, 1$ **do**
4: $\quad \boldsymbol{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ 如果 $t > 1$，否则 $\boldsymbol{z} = \mathbf{0}$
5: $\quad \mathbf{Y}_{(q,t-1)} = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{Y}_{(q,t)} - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}} \epsilon_\theta(\mathbf{C}_t, t, \mathbf{Q}_t) \right) + \sigma_t \boldsymbol{z}$
6: $\quad \mathbf{Q}_{t-1} = [\mathbf{M}_q, \mathbf{Y}_{(q,t-1)}]$
7: $\quad \mathbf{C}_{t-1} \subseteq \mathbf{Q}_{t-1}$ ▷ 与步骤 2 中相同的子集
8: **结束循环**
9: **返回** f_0 在坐标$\mathbf{M}$处的取值 $_q$

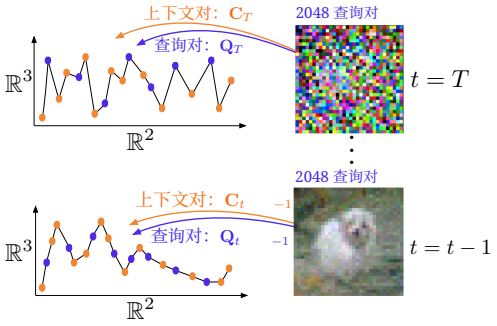


图 4：**左侧：** DPF采样算法。**右侧**：图像域中场的采样过程示意图。

CelebA-HQ 64^2	FID ↓	精度 ↑	召回率 ↑
变分自编码器（Kingma 和 Welling，2014年）	175.33	0.799	0.001
StyleGAN2（Karras 等人，2020a）	5.90	0.618	0.481
Functa（Dupont 等人，2022a）	40.40	0.577	0.397
GEM（Du 等人，2021）	30.42	0.642	0.502
GASP（Dupont 等，2022b）	13.50	0.836	0.312
DPF（我们的方法）	13.21	0.866	0.347

表 1：**CelebA-HQ 上图像生成的定量评估**（Karras 等人，2018）。中间的横线将领域特定方法（上方）与领域无关方法（下方）分隔开。

CIFAR-10 32^2	FID ↓	IS ↑
PixelIQN（Ostrovski 等人，2018）	49.46	5.29
NCSNv2（Song 和 Ermon，2020）	31.75	-
StyleGAN2（Karras 等人，2020a）	3.26	9.74
DDPM（Hoetal.,2020）	3.17	9.46
GEM（Du 等，2021）	23.83	8.36
DPF（我们的方法）	15.10	8.43

表 2：CIFAR-10 上图像生成的定量评估（Krizhevsky，2009）。中间的横线将领域特定方法（上方）与领域无关方法（下方）分隔开来。

需要时使用的每层单元数等）。我们使用 PerceiverIO 架构（Jaegle 等，2022）实现场评分网络 ϵ_θ。这是一种高效的 Transformer，使我们能够扩展上下文对和查询对的数量。更多架构超参数和实现细节见附录。

4.1 二维图像

我们展示了在两个标准图像基准上的实证结果：CelebA-HQ（Karras 等人，2018）64^2和 CIFAR-10（Krizhevsky，2009）32^2。所有图像数据集都映射到其场表示中，其中图像 $\boldsymbol{x} \in \mathbb{R}^{h \times w \times 3}$ 表示为函数 $f : \mathbb{R}^2 \to \mathbb{R}^3$，由坐标- RGB 对定义。在表 1 和表 2 中，我们将 DPF 与 Functa（Dupont 等，2022a）、GEM（Du 等，2021）和 GASP（Dupont 等，2022b）进行比较；这些都是采用场表示、与领域无关的生成模型。为求完整，我们还报告了特定领域的变分自编码器（Kingma 和 Welling，2014）、StyleGAN2（Karras 等人，2020a）和 DDPM（Ho 等人，2020）的结果。同样，在 CIFAR-10 数据上，我们还与多种特定领域的方法进行比较，包括自回归方法（Ostrovski 等人，2018）和基于分数的方法（Song 和 Ermon，2020）。我们针对不同数据集报告弗雷歇 Inception 距离（FID）（Heusel 等人，2017）、Inception 分数（IS）（Salimans 等人，2016）以及精确率/召回率指标（Sajjadi 等人，2018）。

我们观察到，DPF 在 CelebA-HQ 64^2 和 CIFAR-10 数据上都取得了出色的生成性能。特别是，与 Functa（Dupont 等，2022a）和 GEM（Du 等，2021）等旨在统一不同数据域生成建模的近期方法相比，DPF 表现更优。具体而言，我们观察到，在 CelebA-HQ 64^2（Karras 等人，2018）上，DPF 的精确率分数在所有领域无关方法中最高，FID 分数也极具竞争力。一个有趣的发现是，GASP（Dupont 等人，2022b）报告的 FID 分数与 DPF 相当。然而，检查图 5 所示的定性示例时，我们发现 GASP 的样本含有对抗方法通常会产生的伪影。相比之下，DPF 生成的样本更加连贯，且没有伪影。我们认为，这些伪影所导致的纹理多样性是 GASP 与 DPF 的 FID 分数存在差异的原因（类似发现也曾被讨论

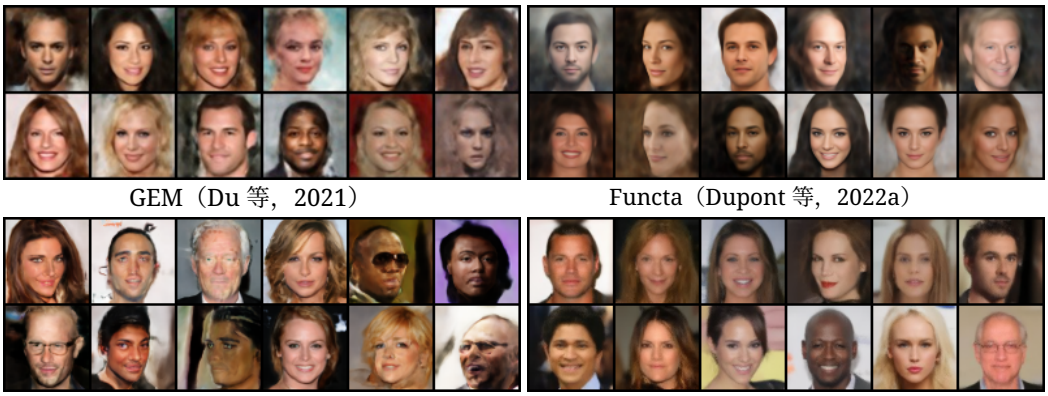


图 5：**不同领域无关方法的定性比较**（CelebA-HQ 64^2（Karras 等人，2018））。DPF 能够生成多样且全局一致的样本，不会出现 Functa（Dupont 等，2022a）和 GEM（Du 等，2021）等依赖初始重建步骤的两阶段方法所产生的背景模糊，也不会出现 GASP（Dupont 等，2022b）等对抗方法产生的伪影。本图中的所有图像均由模型生成，即*ie*，我们并未直接复现 CelebA-HQ 数据集（Karras 等人，2018）中的图像。

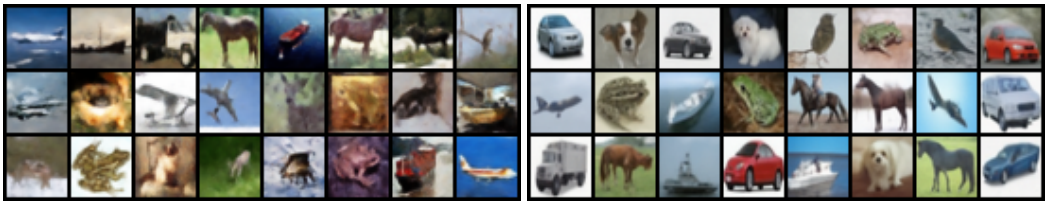


图 6：**定性比较** GEM（Du 等，2021）与 DPF 在 CIFAR-10（Krizhevsky，2009）上的表现。与 GEM（Du 等，2021）相比，DPF 生成的样本更加清晰，同时也能更好地捕捉经验分布。

于（Dupont 等，2022a））。最后，在研究表 2 所示的 CIFAR-10 实证结果时，我们再次发现，DPF 在 FID 和 IS 两项指标上都优于 GEM（Du 等，2021）。

值得注意的是，我们可以观察到通用方法与最新的特定领域方法（如 StyleGAN2（Karras 等人，2020a）和 DDPM（Ho 等人，2020））之间存在差距。这一差距主要由两个因素造成。首先，特定领域方法可以在分数网络中采用针对其领域量身定制的设计选择（即，面向图像的卷积神经网络中的平移等变性）。其次，表 1 和表 2 中报告的通用方法与特定领域方法通常采用不同的训练和推理设置。为进一步研究这一问题，请读者参阅 A 节，其中我们在相同的训练和评估设置下比较 DPF 与 DDPM（Ho 等人，2020）的性能。

最后，我们分别在图 5 和图 6 中展示了领域无关方法在 CelebA-HQ 64^2（Karras 等人，2018）和 CIFAR-10（Krizhevsky，2009）上的定性生成结果。我们注意到，对于 CelebA-HQ，DPF 生成了多样且全局一致的样本，没有出现两阶段方法（Dupont 等，2022a；Du 等，2021）中观察到的模糊现象（尤其是在背景中）；这类方法依赖于初始重构步骤。DPF 也没有出现对抗方法（Dupont 等，2022b）产生的伪影。在 CIFAR-10（Krizhevsky，2009）上进行评估时，我们发现 DPF 生成的结果比 GEM（Du 等，2021）更加清晰且更加多样。

4.2 三维几何

现在我们转向对三维物体分布进行建模的任务，并展示在 ShapeNet 数据集（Chang 等人，2015年）上的结果。我们遵循 GEM（Du 等，2021）的设置，并在

ShapeNet 64^3	覆盖率 ↑	最大均值差异 ↓
潜在 GAN（Chen 和 Zhang，2019）	0.389	0.0017
GASP（Dupont 等，2022b）	0.341	0.0021
GEM（Du 等，2021）	0.409	0.0014
DPF（我们的方法）	0.419	0.0016

表 3：**ShapeNet（Chang 等人，2015年）上的三维几何生成定量评估**。DPF 在覆盖率和最大均值差异指标方面均优于先前方法。

~ 35个 ShapeNet 对象，每个对象都表示为一个分辨率为 64^3 的体素网格。随后，每个对象都表示为一个函数 $f\colon \mathbb{R}^3 \to \mathbb{R}^1$。按照 GEM（Du 等，2021）的设置，我们报告覆盖率和最大均值差异指标（Achlioptas 等，2018）；这些指标是通过对真实值和生成的体素网格所得到的网格进行采样，各取 2048 个点后计算得出的。我们使用倒角距离对它们进行比较。在表 3 中，我们将 DPF 的性能与 3 种基线方法进行比较：潜在 GAN（Chen 和 Zhang，2019）、GASP（Dupont 等，2022b）和 GEM（Du 等，2021）。我们在分辨率 32^3 下训练 DPF。采样时，我们在分数场上评估 32^3 个查询对，然后将结果重塑为分辨率为 32^3 的三维网格，并通过三线性上采样将其提升至最终分辨率 64^3，以计算评估指标。

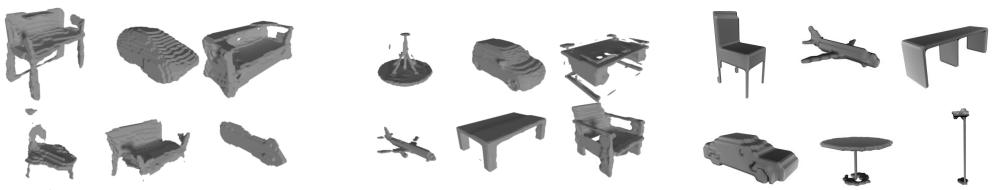
如覆盖率指标所示，在学习 ShapeNet 中物体的多模态分布方面，DPF 的表现优于 GEM（Du 等，2021）和 GASP（Dupont 等，2022b）。此外，尽管 GEM（Du 等，2021）在最大均值差异方面表现更好，但在对图 7 所示生成样本进行视觉比较时，我们并未观察到这种差异。我们认为，最大均值差异得分出现这种差异，是因为最大均值差异对细粒度几何的惩罚过重。

4.3 数据 ON $\mathbb{S}^2$

显然，DPF 也可以学习定义在欧几里得平面之外的场。为此，我们展示了定义在球面上的信号结果。具体而言，按照 Cohen 等人（2018）的做法，我们使用立体投影将图像数据映射到球面上。因此，每个得到的样本都表示为一个场 $f_0\colon \mathbb{S}^2 \to \mathbb{R}^d$。为了在 $\mathbb{S}^2$ 中均匀采样点，我们使用 Driscoll-Healy 算法（Driscoll & Healy, 1994），并分别以 32^2 和 64^2 的分辨率对球面 MNIST（LeCun 等人，1998）和 AFHQ（Choi 等，2020）数据进行采样。图 8 展示了 MNIST（LeCun 等人，1998）和 AFHQ（Choi 等，2020）图像投影到球面后的真实样本分布，以及我们的 DPF 生成的样本。不出所料，由于 DPF 不受度量空间 $\mathcal{M}$ 的几何结构限制，因此能够为定义在球面上的场生成清晰且多样化的样本。

5 相关工作

近年来，生成建模取得了显著进展，生成对抗网络（GANs）（Goodfellow 等人，2014年；Mao 等人，2017年；Karras 等人，2020b）和变分自编码器（VAEs）（Vahdat 和 Kautz，2020）都展现出令人瞩目的性能。更近来，基于扩散的生成建模也取得了卓越且令人信服的成果（Dickstein 等人，2015年；Ho 等人，2020；Song 等人，2021b）。



GASP（Dupont 等，2022b）　GEM（Du 等，2021）　DPF（我们的方法）

Fig图 7：**不同领域无关方法的定性比较**（ShapeNet 上，Chan　g 等人，2015年）。

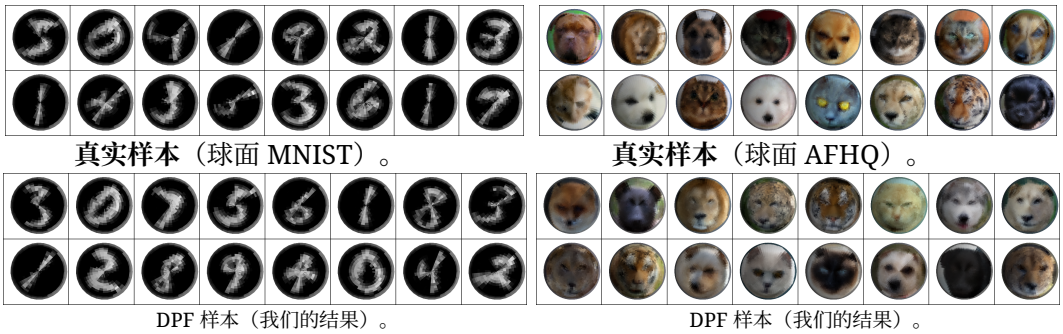


图 8：**球面版本 MNIST 和 AFHQ 的经验样本与生成样本的定性比较**（Choi 等，2020）。

DPF 中提出的表述与黎曼生成模型领域的研究（Bortoli 等，2022；Gemici 等，2016；Rozen 等，2021）相互独立。黎曼生成建模旨在明确地将学习到的密度约束在黎曼流形结构上。例如，黎曼生成模型可以学习球面 $\mathbb{S}^2$ 上的点 $\boldsymbol{x} \in \mathbb{S}^2$ 分布，并明确确保生成的任何样本都位于球面上。相比之下，DPF 学习的是信号样本集合的分布上的生成模型，这些信号定义在球面或任何其他度量空间上。

DPF 的 formulation 也不同于最近提出的 Functa（Dupont 等，2022a）、GEM（Du 等，2021）和 GASP（Dupont 等，2022b）。前两种方法采用潜在场参数化（Park 等，2019）和两阶段训练范式。然而，与我们的工作不同，Functa 和 GEM 中的场网络通过超网络（Ha 等，2017）进行参数化，该超网络以可训练的潜向量作为输入。在训练过程中，首先在自动解码或压缩阶段（Park 等，2019）优化每个场对应的一个小型潜向量。在第二阶段，根据这些潜向量学习概率模型。GASP（Dupont 等，2022b）利用生成对抗网络，其生成器生成场数据，而点云判别器则作用于离散化数据，旨在区分输入来源，即真实数据或生成数据。Functa（Dupont 等，2022a）或 GEM（Du 等，2021）等两阶段方法，使第二阶段的概率模型训练相比 DPF 更具计算效率。这种概率模型训练效率的提升通常以第一阶段将场压缩为小型潜向量为代价，而这一过程本身具有不可忽略的计算成本，尤其是对于大规模场数据集而言。

DPF 中提出的表述与近期关于神经过程（NP）的研究密切相关（Garnelo 等，2018；Kim 等，2019；Dutordoir 等，2022），这些研究也通过上下文与查询对来学习函数上的分布。与神经过程的表述不同，后者通过优化证据下界（Kingma 与 Welling（2014））来实现；我们将 DPF 表述为函数空间中的去噪扩散过程，从而得到稳健的去噪训练目标和强大的迭代推理过程。与同期关于神经过程的研究相比（Garnelo 等，2018；Kim 等，2019；Dutordoir 等，2022），我们并未明确表述一个条件推理问题，而是着眼于更具挑战性的任务：学习场上的无条件生成模型。我们针对二维图像和三维形状的复杂场分布，以及定义在非欧几里得几何上的场分布，对这一假设进行了广泛测试。

6 结论

本文通过引入 DPF，在场分布建模方面取得了进展。DPF 是一种扩散概率模型，无需采用初始重建阶段（Dupont 等，2022a；Du 等，2021），也无需调整不稳定的对抗方法（Dupont 等人，2022b），即可直接捕捉场分布。我们展示了 DPF 能够在不同领域捕捉场分布，而不对数据作任何假设。这使我们能够在不同领域采用相同的去噪架构，并获得令人满意的结果。此外，我们还展示了 DPF 可用于学习定义在非欧几里得几何上的场分布。

7 伦理声明

在考量 DPF 等生成模型的伦理影响时，有几个方面值得关注：使用生成模型创造虚假数据，例如“深度伪造”（Mirsky & Lee，2021）；训练数据泄露与隐私（Tinsley 等人，2021）；以及放大训练数据中已有的偏见（Jain 等人，2020）。关于生成建模伦理考量的深入综述，请参阅 Rostamzadeh 等人（2021）。此外，我们想强调，文中展示的所有人脸图像均由我们的模型生成，并非直接复现任何数据集中的人脸图像。

8 可复现性声明

我们非常重视结果的可复现性。我们提供了公开实现的链接，可用于复现我们的结果，详见 A 节和 B 节；此外，表 6 列出了所有训练参数。我们报告结果所使用的所有数据集均为公开数据集，可免费下载。

参考文献

P. Achlioptas、P. Diamanti、I. Mitliagkas 和 L. Guibas。《三维点云的表示学习与生成模型》。发表于*ICML*，2018 年。

A. Bansal、E. Borgnia、H. Chu、J. Li、H. Kazemi、F. Huang、M. Goldblum、J. Geiping 和 T.Goldstein。冷扩散：无需噪声即可逆转任意图像变换。*arXiv*，2022年。

V. Bortoli、E. Mathieu、M. Hutchinson、J. Thornton、Y. Teh 和 A. Doucet。黎曼分数生成建模。*arXiv*，2022年。

A. Chang、T. Funkhouser、L. Guibas 等。ShapeNet：一个信息丰富的三维模型库。*arXiv*，2015年。

Z. Chen 和 H. Zhang。学习用于生成式形状建模的隐式场。在*CVPR*，2019。

Y. Choi、Y. Uh、J. Yoo 和 J. Ha。StarGAN v2：面向多个领域的多样化图像合成。在*CVPR*，2020。

T. Cohen、M. Geiger、J. Köhler 和 M. Welling。球面卷积神经网络。载于*ICLR*，2018。

P. Dhariwal 和 A. Nichol. 扩散模型在图像合成方面胜过生成对抗网络. 载于*NeurIPS*，2021. J. Dickstein、E. Weiss、N. Maheswaranathan 和 S. Ganguli. 使用非平衡热力学进行深度无监督学习. 载于*ICML*，2015. J. Driscoll 和 D. Healy. 在二维球面上计算傅里叶变换和卷积. 《应用数学进展》，1994. Y. Du、S. Li 和 I. Mordatch. 使用基于能量的模型进行组合式视觉生成. 载于*NeurIPS*，2020. Y. Du、K. Collins、J. Tenenbaum 和 V. Sitzmann. 学习神经场的信号无关流形. 载于*NeurIPS*，2021. E. Dupont、H. Kim、S. Eslami、D. Rezende 和 D. Rosenbaum. 从数据到 functa：数据点是一个函数，你应该像对待函数一样对待它. 载于*ICML*，2022a. E. Dupont、Y. Teh 和 A. Doucet. 生成模型作为函数分布. 载于 *AISTATS*，2022b. V. Dutordoir、A. Saul、Z. Ghahramani 和 F. Simpson. 神经扩散过程. *arXiv*，2022. B Everett. 潜变量模型导论. 施普林格，2013. M. Garnelo、J. Schwarz、D. Rosenbaum、F. Viola、D. Rezende、SM Eslami 和 Y. Teh. 神经过程. *ICML*研讨会，2018.

M. Gemici、D. Rezende 和 S. Mohamed. 黎曼流形上的归一化流. *arXiv*，2016. I. Goodfellow、J. Pouget-Abadie、M. Mirza、B. Xu、D. Warde-Farley、S. Ozair、A. Courville 和 Y. Bengio. 生成对抗网络. 载于*NeurIPS*，2014. D. Ha、A. Dai 和 Q. Le. 超网络. 载于*ICLR*，2017. M. Heusel、H. Ramsauer、T. Unterthiner、B. Nessler 和 S. Hochreiter. 使用双时间尺度更新规则训练的生成对抗网络收敛至局部纳什均衡. 载于*NeurIPS*，2017. J. Ho、A. Jain 和 P. Abbeel. 去噪扩散概率模型. 载于*NeurIPS*，2020. J. Ho、T. Salimans、A. Gritsenko、W. Chan、M. Norouzi 和 D. J Fleet. 视频扩散模型. 载于*ICLR*，2022. E. Hoogeboom、V. Satorras、C. Vignac 和 M. Welling. 用于三维分子生成的等变扩散. 载于*ICML*，2022. A. Jaegle、S. Borgeaud、J. Alayrac 等. Perceiver IO：一种用于结构化输入与输出的通用架构. 载于*ICLR*，2022. N. Jain、A. Olmo、S. Sengupta、L. Manikonda 和 S. Kambhampati. 不完美想象：生成对抗网络加剧偏见对人脸数据增强和 Snapchat 自拍滤镜的影响.*CORR*，2020. T. Karras、T. Aila、S. Laine 和 J. Lehtinen. 生成对抗网络的渐进式增长：提升质量、稳定性和多样性. 载于*ICLR*，2018. T. Karras、S. Laine 和 T. Aila. 用于生成对抗网络的基于风格的生成器架构. 载于*CVPR*，2019. T. Karras、M. Aittala、J. Hellsten、S. Laine、J. Lehtinen 和 T. Aila. 使用有限数据训练生成对抗网络. 载于*NeurIPS*，2020a. T. Karras、S. Laine、M. Aittala 等. 分析并改进 StyleGAN 的图像质量. 载于*CVPR*，2020b. H. Kim、A. Mnih、J. Schwarz、M. Garnelo、A. Eslami、D. Rosenbaum、O. Vinyals 和 Y. Teh. 注意力神经过程. *ICLR*，2019. D. Kingma 和 J. Ba. Adam：一种随机优化方法. 载于*ICLR*，2015. D. Kingma 和 M. Welling. 自编码变分贝叶斯. 载于*NeurIPS*，2014. N. Kodali、J. Abernethy、J. Hays 和 Z. Kira. 关于生成对抗网络的收敛性与稳定性. *arXiv*，2017. A. Krizhevsky. 从微小图像中学习多层特征. 技术报告，多伦多大学，2009. Y. LeCun、L. Bottou、Y. Bengio 和 P. Haffner. 应用于文档识别的基于梯度的学习. 载于*IEEE* 会议论文集，1998. S. Luo 和 W. Hu. 用于三维点云生成的扩散概率模型. 载于*CVPR*，2021. X. Mao、Q. Li、H. Xie 等. 最小二乘生成对抗网络. 载于*ICCV*，2017. Y. Mirsky 和 W. Lee. 深度伪造的创建与检测：综述. *CSUR*，2021. A. Nichol 和 P. Dhariwal. 改进的去噪扩散概率模型. 载于*ICML*，2021. G. Ostrovski、W. Dabney 和 R. Munos. 用于生成建模的自回归分位数网络. 载于*ICML*，2018. J. Park、P. Florence、J. Straub、R. Newcombe 和 S. Lovegrove. DeepSDF：学习用于形状表示的连续有符号距离函数. 载于*CVPR*，2019.

Joaquin Quinonero-Candela 和 Carl Edward Rasmussen。稀疏近似高斯过程回归的统一视角。《机器学习研究杂志》，6:1939–1959，2005年。S. Rissanen、M. Heinonen 和 A. Solin。基于逆热耗散的生成建模。*arXiv*，2022。R. Rombach 、 A. Blattmann 、 D. Lorenz、P. Esser 和 B. Ommer。使用潜在扩散模型进行高分辨率图像合成。发表于 *CVPR*，2022。O. Ronneberger、P. Fischer 和 T. Brox。U-Net：用于生物医学图像分割的卷积网络。发表于*MICCAI*，2015年。N. Rostamzadeh、E. Denton 和 L. Petrini。计算机视觉中的伦理与创造力。*arXiv*，2021。N. Rozen、A. Grover、M. Nickel 和 Y. Lipman。Moser流：流形上的基于散度的生成建模。发表于*NeurIPS*，2021。M. Sajjadi、O. Bachem、M. Lucic、O. Bousquet 和 S. Gelly。通过精确率和召回率评估生成模型。发表于*NeurIPS*，2018。T. Salimans、I. Goodfellow、W. Zaremba、V. Cheung、A. Radford 和 X. Chen。改进生成对抗网络训练技术。发表于*NeurIPS*，2016。J. Song、C. Meng 和 S. Ermon。去噪扩散隐式模型。发表于*ICLR*，2021a。Y. Song 和 S. Ermon。改进基于分数的生成模型训练技术。发表于*NeurIPS*，2020。Y. Song、J. Dickstein、D. P. Kingma、A. Kumar、S. Ermon 和 B. Poole。通过随机微分方程进行基于分数的生成建模。发表于*ICLR*，2021b。P. Tinsley、A. Czajka 和 P. Flynn。这张脸并不存在……但它可能是你的！生成模型中的身份泄露。发表于*WACV*，2021。Michael Tipping。相关向量机。神经信息处理系统进展，12，1999。Michalis Titsias。稀疏高斯过程中诱导变量的变分学习。发表于人工智能与统计，第567–574页。PMLR，2009年。Ilya O Tolstikhin、Neil Houlsby、Alexander Kolesnikov、Lucas Beyer、Xiaohua Zhai、Thomas Unterthiner、Jessica Yung、Andreas Steiner、Daniel Keysers、Jakob Uszkoreit 等。MLP-mixer：用于视觉的全MLP架构。神经信息处理系统进展，34：24261–24272，2021。A. Vahdat、K. Kreis 和 J. Kautz。在潜在空间中进行基于分数的生成建模。发表于 *NeurIPS*，2021。Arash Vahdat 和 Jan Kautz。NVAE：深层层次变分自编码器。神经信息处理系统进展，33：19667–19679，2020。Ashish Vaswani、Noam Shazeer、Niki Parmar、Jakob Uszkoreit、Llion Jones、Aidan N Gomez、Łukasz Kaiser 和 Illia Polosukhin。注意力机制就是你所需要的一切。神经信息处理系统进展，30，2017。M. Welling 和 Y. Teh。通过随机梯度朗之万动力学进行贝叶斯学习。发表于*ICML*，2011。Christopher Williams 和 Matthias Seeger。使用Nyström方法加速核机器。神经信息处理系统进展，13，2000。Y. Xie、T. Takikawa、S. Saito、O. Litany、S. Yan、N. Khan、F. Tombari、J. Tompkin、V. Sitzmann 和 S. Sridhar。视觉计算及其他领域中的神经场。发表于《计算机图形学论坛》，2022。F. Yu、A. Seff、Y. Zhang 等。LSUN：利用深度学习与人在环方法构建大规模图像数据集。*arXiv*，2015年。S. Zhai、W. Talbott、N. Srivastava、C. Huang、H. Goh、R. Zhang 和 J. Susskind。无注意力机制的 Transformer。发表于*ICML*，2022。

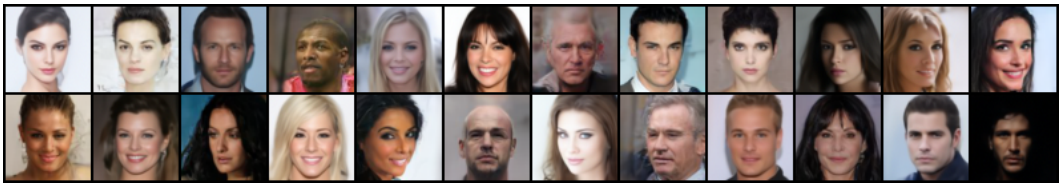
附录

在这份补充材料中，我们提供了消融研究（A 节）、实现细节（B 节）、更多实验和可视化（C 节），以及局限性与未来工作（G 节）。

A 消融实验

本节对 DPF 的关键设计选择和超参数进行消融实验。我们在 CelebA-HQ 64^2 数据集上训练所有模型变体（Karras 等人，2018）。

DPF 与 DDPM（Ho 等人，2020）之间的性能差距。 我们的目标是公平比较 DPF 与 DDPM（Ho 等人，2020）。为此，我们采用 Ho 等（2020）的 DDPM 分数网络实现[4]，并使用相同设置训练 DDPM 和 DPF：批次大小、学习率计划、训练轮次等。推断时，我们使用祖先采样并执行 1000 个去噪步数。在表 4 中，我们发现 DDPM（Ho 等人，2020）（一种特定领域的方法）与 DPF 之间的差距远小于表 2 中的差距。此外，当我们近似匹配 DDPM 分数网络与我们的分数场模型的参数数量（67.6M 对 62.4M）时，发现 DPF 在 FID 和召回率指标上均优于 DDPM。我们认为这是因为 DPF 能更好地捕捉全局一致的样本和长程交互。图 9 展示了 DDPM（Ho 等人，2020）和 DPF 生成的定性结果。



DDPM（Ho 等人，2020）（120M 个参数）



DPF（我们的方法，62.4M 个参数）

图 9：**DDPM（Ho 等人，2020）和 DPF 的结果可视化**，使用 CelebA-HQ（Karras 等人，2018）数据集。本图中的所有图像均由模型生成，并非直接从 CelebA-HQ（Karras 等人，2018）数据集中复制而来。

此外，我们还想了解 DPF 的主要瓶颈究竟在于编码上下文对，还是解码查询对（例如，在查询坐标处评估场）。为此，我们设计了两项消融实验。一项实验是用 UNet 替换 DPF 编码器，另一项则是用 UNet 替换 DPF 解码器。表 4 展示了定量结果。我们观察到，采用 UNet解码器的 DPF 在 FID、精确率和召回率方面表现最佳。这些实证结果表明，通过改进 DPF 对查询对的解码，可能会进一步提升性能。

无分辨率限制的生成。 重要的是，DPF 与经典 DDPM（Ho 等人，2020）的不同之处在于，DPF 生成的场可以使用查询对在连续点上进行评估，即。，训练期间使用的上下文对不同于推断时应用的查询对。这使 DPF 能够学习连续场，而不是简单地记忆离散化数据样本。为验证这一点，我们以不同于训练分辨率的分辨率对 DPF 进行采样。具体而言，我们在分辨率为 64^2 的 CelebA-HQ（Karras 等人，2018）上训练 DPF，同时使用 128^2 查询对生成高分辨率图像，如图 10 所示。从图 10 中可以观察到，在训练期间给定低分辨率上下文和查询对后，DPF 能够生成准确的高分辨率图像场。作为参考，我们比较

[4]代码可在 https://github.com/rosinality/denoising-diffusion-pytorch 获取。



图 10：**超分辨率图像生成的定性结果**。我们根据 64^2 上下文点（左上）生成 128^2分辨率的图像（第 *1* 行）。我们展示放大结果（第 *2* 行）。我们将结果与最近邻和双线性上采样进行比较。本图中的所有图像均由模型生成，*ie.*，也就是说，我们并未直接复现 CelebA-HQ 数据集中的图像（Karras 等人，2018）。

CelebA-HQ 64^2	# 参数.	# 通道	FID ↓	精确率 ↑	召回率 ↑
DDPM（Ho 等人，2020）	120M	128	11.26	0.916	0.384
DDPM（Ho 等人，2020）	67.6M	96	24.93	0.927	0.213
DPF（我们的方法）	62.4M	-	13.21	0.866	0.347
DPF（使用 UNet 编码器）	127M	128	20.79	0.916	0.323
DPF（使用 UNet解码器）	62.5M	128	12.27	0.877	0.457

表4：**DDPM（Ho 等人，2020）与 DPF 在相同设置下的对比**。DDPM 模型采用 UNet架构（Ronneberger等人，2015）实现，卷积通道数分别为 128（第 *1* 行）和 96（第 *2* 行）。我们还使用 UNet架构替换 DPF 的编码器（第 *4* 行）或解码器（第 *5* 行）。

了我们的结果与标准的最近邻或双线性上采样结果。在图 10 中，我们展示了与上采样结果相比，我们的结果在纹理细节方面具有相当甚至更高的质量。

DPF超参数。我们对不同的 DPF超参数进行了消融实验。在表 5 中，我们研究了训练期间所用上下文对数量变化的影响。在表 6 中，我们评估了不同的 PerceiverIO（Jaegle 等，2022）超参数。对于表 5 和表 6 中的对比，我们训练模型 180 个训练轮次，并使用 $1k$ 样本计算 FID指标。

如表 5 所示，减少上下文对的数量会导致 FID 分数下降。这是预期结果，因为较少的上下文对会降低场表示的保真度。还值得注意的是，我们没有针对减少后的上下文对数量优化分数网络超参数，即，这些超参数仅根据使用完整上下文对集合训练的模型性能进行选择。

表 6 中报告的结果展示了通过不同手段减少 PerceiverIO 架构参数数量的效果。我们观察到：降低潜变量容量（无论是减少潜变量数量还是降低其维度）的影响，比降低模型深度（编码器、解码器中的块数，或编码器块中的自注意力层数）的影响更大。

CelebA-HQ 64^2	FID
上下文对数量 $= 1028$（50%）	103.86
上下文对数量 $= 2048$（75%）	91.12
上下文对数量 $= 4096$（100%）	74.89

Tab表 5： **使用不同数量上下文对训练的 DPF 的性能**。上下文对为 e
ran训练过程中随机抽取样本。经过 180 个训练轮次后，评估 $1k$ 个样本的 FID 分数。

CelebA-HQ 64^2	# 参数。	FID
# 潜变量 $= 256$	62.3M	93.54
# 潜变量维度 $= 256$	22.8M	96.63
# 解码器块 $= 2$	59.7M	77.29
# 块 $= 6$	35.4M	83.87
# S.A.块 $= 1$	43.5M	75.52
Base	62.4M	74.89

表 6：**使用不同 PerceiverIO（Jaegle 等人，2022）超参数时 DPF 的性能**。我们在实验中使用的基础超参数为：512 个维度为 512 的潜变量、12 个包含 2 个自注意力模块的编码器块，以及 4 个解码器块。FID 分数在经过 180 个训练轮次后针对 $1k$ 个样本进行评估。有关超参数和实现细节的更多信息，请参阅 B 节。

B 实现细节

DPF 中分数场的实现设计空间涵盖了所有能够处理非规则采样数据的架构，例如 Transformer（Vaswani 等，2017）或多层感知机（Tolstikhin 等，2021）（见 D 节）。除非另有说明，我们使用 PerceiverIO（Jaegle 等，2022）实现 DPF，这是一种高效的编码器—解码器 Transformer 架构。选择 PerceiverIO 的原因是，它提供了一种直接处理大量上下文与查询对的方法，并能通过注意力机制自然地对上下文与查询对之间的交互进行编码。在图 11 中，我们展示了上下文与查询对在 PerceiverIO 架构中的用法。具体而言，编码器通过交叉注意力层将上下文对映射到潜变量数组（例如 一组可学习向量）。类似地，解码器通过交叉注意力块将查询对映射到潜变量数组。关于 PerceiverIO 架构的更多讨论，请参阅 Jaegle 等人（2022）。为了使分数计算以时间步 t 为条件，我们只需将 t 的位置嵌入表示与上下文与查询对进行拼接。所有进行定量评估的实验所采用的 PerceiverIO 设置见表 6。实际上，DPF 编码器包含 12 个 Transformer 块，每个块包含 1 个交叉注意力层，后接 2 个自注意力层。交叉注意力层和自注意力层均具有 4 个注意力头。对于 64^2 分辨率的图像，潜在数组的维度为 512×512；对于 32^2 分辨率的图像，维度为 1024×1024。我们使用频率分别为 10 和 64 的傅里叶特征位置编码来表示坐标和时间步。训练期间，我们使用 Adam（Kingma 与 Ba，2015）优化器。我们将学习率设为 $1e-4$。对于所有图像数据集，我们将批次大小设为 16。所有实验均使用 8 个 A100 图形处理器。最后，对于 PerceiverIO，我们使用了公开代码库的一个修改版本。[5]

C 附加实验

去噪过程的可视化。图 12 展示了在 CelebA-HQ（Karras 等人，2018）和 CIFAR-10（Krizhevsky，2009）数据上，去噪过程随时间从左到右的变化。
其他图像数据集上的视觉效果。图 16 展示了 DPF 在其他图像数据集上的定性结果，包括分辨率为 32×32 的 AHFQ（Choi 等，2020）、FFHQ（Karras 等人，2019）和 LSUN 教堂（Yu 等人，2015）数据。图 17 展示了在球面几何上生成的场的视频（使用 Adobe Acrobat 查看动画）。

D DPF 中分数场网络 ϵ_θ 的架构消融实验

DPF 的构造与分数网络 ϵ_θ 的具体实现无关，分数模型的设计空间涵盖了大量选项。尤其是，所有架构

[5] https://huggingface.co/docs/transformers/model_doc/perceiver

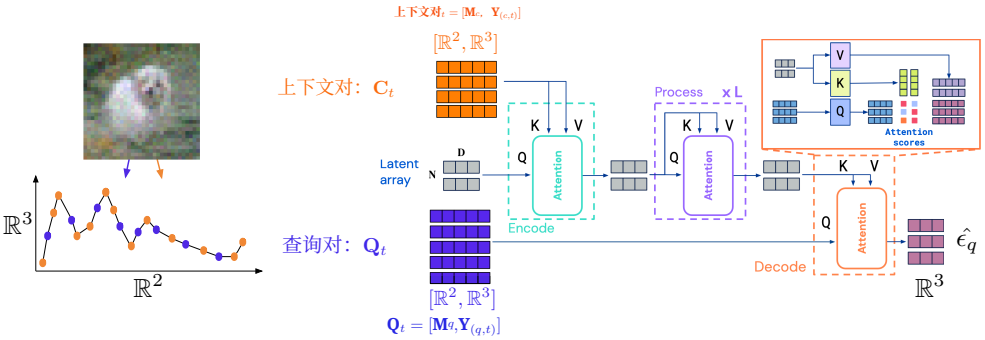


图 11：PerceiverIO 架构中上下文与查询对的交互。上下文对$\mathbf{C}_t$ 通过交叉注意力机制关注由可学习参数构成的潜在数组。随后，潜在数组经过多个自注意力模块。最后，查询对 $\mathbf{Q}_t$ 与潜在数组进行交叉注意力计算，从而生成最终的噪声预测 $\hat{\epsilon}_q$。

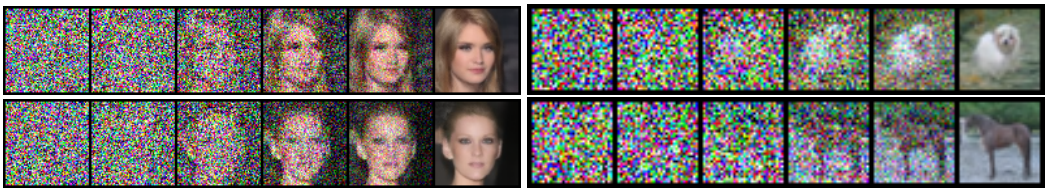


图 12：CelebA-HQ 上从 $t = 1000$ （噪声）到 $t = 0$ （图像）的**去噪过程可视化**（Karras 等人，2018）（左），以及 CIFAR-10 上的结果（Krizhevsky，2009）（右）。

能够处理不规则数据的模型，例如 Transformer 或多层感知机，都是可行的候选方案。为支持这一观点，我们在 32×32 分辨率下对 CelebA-HQ 进行了评估，并比较了 3 种不同的架构：PerceiverIO（Jaegle 等人，2022）、原始 Transformer 编码器（Vaswani 等人，2017）和 MLP-mixer（Tolstikhin 等，2021）。我们使用推断过程中获得的 1000 个样本，报告 FID 和 KID 指标；推断过程详见算法 2。为进行公平比较，我们大致统一了各模型的参数量（约为 55M）和设置（例如例如块数、每块的参数等），并训练了 500 个训练轮次。为简化评估，我们使用相同数量的上下文与查询对。Transformer 编码器和 MLP-mixer 均使用各自的架构处理作为输入的上下文对，随后将得到的潜变量与相应的查询对拼接，并馈入线性投影层以进行最终预测。

在表 8 中，我们可以看到，DPF 的公式设计适用于分数场模型的多种不同架构实现。特别是，我们观察到不同架构的整体性能相当接近，从基于变换器的架构到多层感知机皆是如此。查看图 14 所示的定性样本时，我们也观察到了相同的现象。这支持了我们的主张：DPF 的优势源于其公式设计，而非分数场模型的某种特定实现。每种架构都有其自身的优点，例如，MLP混合器可实现高吞吐量，变换器编码器易于实现，而 PerceiverIO 则支持数量庞大且可变的上下文与查询对。我们认为，结合这些不同架构的优势，是 DPF 未来极具前景的发展方向。最后请注意，这些实证结果无法与论文其他章节报告的结果直接比较，因为这些模型的参数量通常只有其他章节所用模型的大约 50%。

此外，图 13 还定性比较了在第 4 节所用球面 MNIST 数据集上，MLP混合器与 PerceiverIO 的结果。可以看到，PerceiverIO 和 MLP-mixer 都成功捕捉了球面上数字的分布。这支持了这样一种观点：DPF 所引入的与数据无关的生成建模能力，并非取决于分数场的具体实现，而是源于其公式化方式。

超参数	CelebA-HQ（Karras 等人，2018）	CIFAR-10（Krizhevsky，2009年）	ShapeNet（Chang 等人，2015年）
训练结果	64^2	32^2	32^3
评估结果	64^2	32^2	64^3
#上下文对	4096	2048	32768
#查询对	4096	2048	32768
#维度坐标	2	2	3
#维度信号	3	3	1
#频率位置嵌入	10	10	10
#频率位置嵌入 t	64	64	64
#潜变量	1024	512	512
#潜变量维度	1024	512	1024
#块	12	12	12
#解码器块	1	1	1
#每个块的自注意力次数	2	2	2
#自注意力头	4	4	4
#交叉注意力头	4	4	4
批次大小	128	128	16
lr	$1e-4$	$1e-4$	$1e-4$
训练轮次	1300	1300	1000

表 7：在不同数据集上使用 DPF 的超参数和设置。

CelebA-HQ 32^2	FID ↓	KID ↓ $\times 10^3$
PceiverIO（Jaegle 等，2022）	64.8	7.10
Transformer 编码器（Vaswani 等人，2017年）	70.1	10.27
MLP-mixer（Tolstikhin 等，2021）	66.8	7.89

表 8：**CelebA-HQ（Karras 等人，2018）上不同分数场实现的图像生成**定量评估，分辨率为 32x32 ϵ_θ。

E 与 Functa（Dupont 等人，2022a）和 GASP（Dupont 等，2022b）在 $\mathbb{S}^2$上的定性比较。

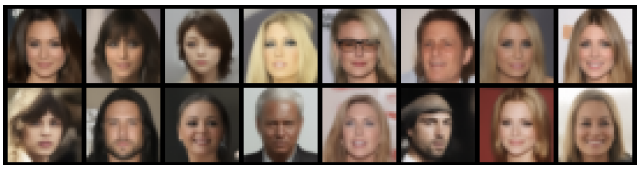
我们对 Functa（Dupont 等，2022a）和 GASP（Dupont 等，2022b）在球面上定义的场进行了定性比较。为此，我们使用了随 GASP 一同发布的 ERA5 温度数据集 https://github.com/EmilienDupont/ neural-function-distributions，该数据集的球面分辨率为 46×90。需要注意的是，该数据集不同于用于训练 Functa（Dupont 等，2022a）的 ERA5 数据集；后者采样分辨率更高（181×360），且未在其代码库 https://github.com/deepmind/functa 中公开。在图 15 中，我们展示了 Functa（Dupont 等，2022a）、GASP（Dupont 等，2022b）和 DPF 生成的样本。我们观察到，DPF 生成的样本与 Functa 或 GASP 生成的样本在视觉上难以区分。DPF 捕捉到了与 Functa 和 GASP 相同的模式：两极温度较低（例如）.如深紫色调），而赤道和热带地区的温度则明显较高，以较浅的黄色显示。

F 采样上下文与查询对

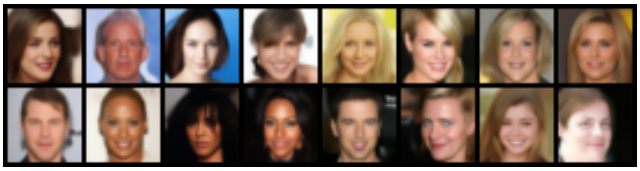
通过上下文与查询对来表述 DPF，自然会引出以下问题：从场中采样哪些上下文与查询对，有什么直观的依据吗？

对于查询对，答案很简单：应根据问题定义的需要采样尽可能多的查询对。例如，如果任务是在 32×32 分辨率下对图像的场表示进行采样，就应采样 $32 \times 32 = 1024$ 个查询对。从这个意义上说，查询对的数量和分布完全取决于具体问题，具有灵活性，而 DPF 也支持这种灵活性。

选择要采样的上下文对 是一个复杂得多的问题。为讨论这一问题，我们利用 DPF 的表述与高斯过程（GP）之间的密切关系。具体而言，可以将选择上下文对的问题视为等同于选择诱导变量的问题，这属于稀疏高斯过程中的组合优化问题（Quinonero-Candela 和 Rasmussen，2005），需要通过近似方法求解。尽管已有多种近似方法用于



(a) PerceiverIO（Jaegle 等，2022）。

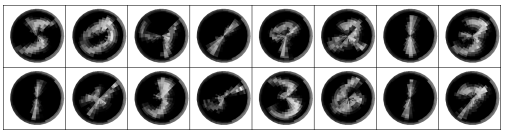


(b) Transformer 编码器（Vaswani 等，2017）



(c) MLP-mixer（Tolstikhin 等，2021）

Fig图 13：**不同架构的定性比较**，用于实现分数场模型 ϵ_θ .



(a) PerceiverIO（Jaegle 等，2022）



(b) MLP-mixer（Tolstikhin 等，2021）。

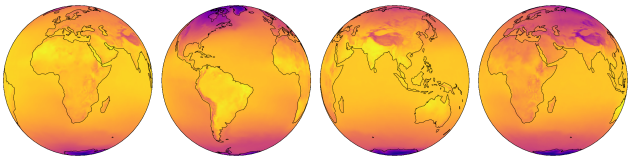
Fig图 14：**定性比较**不同实现的分数场，用于函数 s defi义在球面上。(a) PerceiverIO（Jaegle 等，2022）。(b) MLP-mixer（Tolstikhin 等，2021).

选择诱导变量的问题，我们重点介绍三种经典方法，它们分别代表了更广泛的几类方法。Titsias，2009 提出了一种变分近似，通过最大化真实对数边际似然的下界，联合推断诱导输入和核超参数。这一表述的一个关键特性是将诱导输入定义为变分参数。另一种方法是 Williams 和 Seeger（2000）提出的 Nyström 近似，它通过诱导变量（即上下文对）对格拉姆矩阵进行特征分解。最后，Tipping（1999）的研究提出了相关向量机（RVM），可以将其视为嵌入贝叶斯学习框架中的支持向量机稀疏版本。需要注意的是，由于 RVM 是一个权重服从高斯先验的有限线性模型，因此可以将其视为高斯过程。

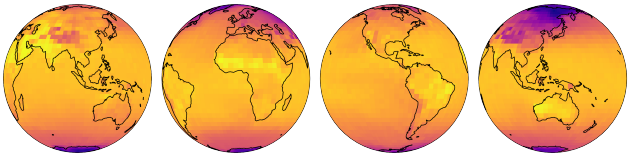
未来工作的一个有趣方向是借鉴稀疏高斯过程近似，构建一种学习目标，以分层方式联合学习 DPF 中的生成模型以及上下文对的分布。

G 局限性与未来工作

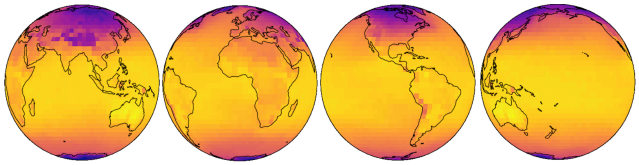
尽管 DPF 在学习场分布方面取得了进展，但仍存在一些局限，也有值得未来探索的方向。首先，将分数网络直接实现为 Transformer，即使在低分辨率下，计算开销也会大到难以承受，因为对上下文与查询对计算注意力机制的成本随其数量呈二次增长。为缓解这一问题，我们采用了 PerceiverIO（Jaegle 等，2022）架构，其计算复杂度随查询对和上下文对的数量线性增长。为了进一步解决这一问题，未来可以探索其他高效 Transformer 架构（Zhai 等，2022）。此外，与 DDPM（Ho 等人，2020）类似，DPF 在采样过程中会遍历所有时间步，以便在推断时生成场，因此速度较慢



(a) Functa（Dupont 等，2022a）（181×360 分辨率）

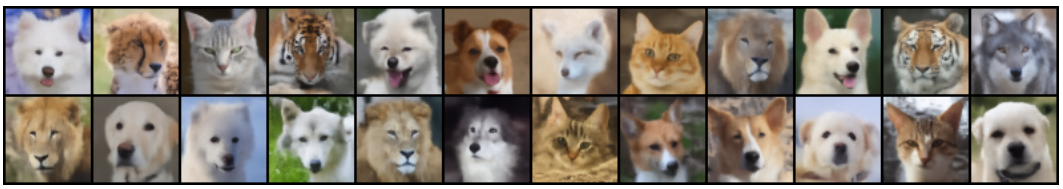


(b) GASP（Dupont 等，2022b）（46×90 分辨率）

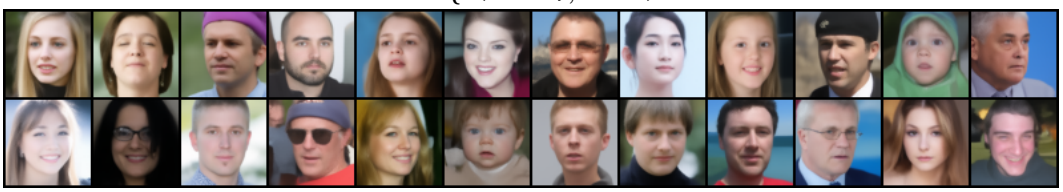


(c) DPF（本文方法）（46×90 分辨率）

图 15：**分数场的定性比较**，比较了为定义在 $\mathbb{S}^2$上的场所实现的不同分数场。(a) Functa（Dupont 等，2022a）。(b) GASP（Dupont 等，2022b）。(c) DPF（本文方法）。



AFHQ（Choi 等，2020）



FFHQ（Karras 等人，2019）



LSUN 教堂（Yu 等人，2015年）

图 16：**图像数据集上的更多结果**。我们展示了使用 DPF 在 AHFQ（Choi 等，2020）、FFHQ（Karras 等人，2019）和 LSUN 教堂（Yu 等人，2015年）数据上生成的结果，分辨率为 32×32。人脸图像由 DPF 生成，即，我们并未直接复现 FFHQ（Karras 等人，2019）中的人脸图像。

与生成对抗网络相比。现有研究通过牺牲样本质量和多样性来加速采样（Song 等人，2021a）。值得指出的是，像（Song 等人，2021a）这样的改进推理方法可以直接集成到 DPF 中。

Fi图 17：**生成的球面结果视频**。请使用 Adobe Acrobat 查看动画。