

分散扩散模型

David McAllister 1 * Matthew Tancik 2 Jiaming Song 2 Angjoo Kanazawa 1 †

1 加州大学伯克利分校 2 Luma AI

摘要

大规模人工智能模型训练需要在数以千计的 GPU 上进行分工，然后每一步都要在它们之间同步梯度。这带来了巨大的网络负担，只有集中式的单体集群才能支持，从而导致基础设施成本上升，电力系统不堪重负。我们提出了分散式扩散模型（Decentralized Diffusion Models），这是一个可扩展的框架，通过消除对集中式高带宽网络结构的依赖，在独立集群或数据中心之间分配扩散模型训练。我们的方法在数据集的分区上训练一组专家扩散模型，每个分区之间完全隔离。推理时，专家们通过轻量级路由器进行组合。我们的研究表明，与在整个数据集上训练的单个模型相比，该集合能共同优化相同的目标。这意味着我们可以在多个“计算岛”之间分担训练负担，从而降低基础设施成本，提高对局部 GPU 故障的恢复能力。分散式扩散模型使研究人员能够利用更小、更具成本效益、更容易获得的计算，如按需 GPU 节点，而不是中央集成系统。我们在 ImageNet 和 LAION Aesthetics 上进行了大量实验，结果表明分散式扩散模型的 FLOP 对 FLOP 性能优于标准差异融合模型。最后，我们将我们的方法扩展到 240 亿个参数，证明现在只需 8 个独立的 GPU 节点，就能在不到一周的时间内训练出高质量的扩散模型。

1. 导言

扩散模型在图像生成[38, 39]、视频建模[4]和通过扩散策略控制机器人[10]方面取得了突破性成果。然而，这些模型仍然需要越来越多的训练计算。即使是早期在图像扩散方面取得的成功也凸显了巨大的计算需求，稳定扩散 1.5 的训练消耗了超过 6000 A100 GPU 天[7, 33]。

* 作者在 Luma AI 实习时完成了部分工作 † 作者是 Luma AI 的顾问

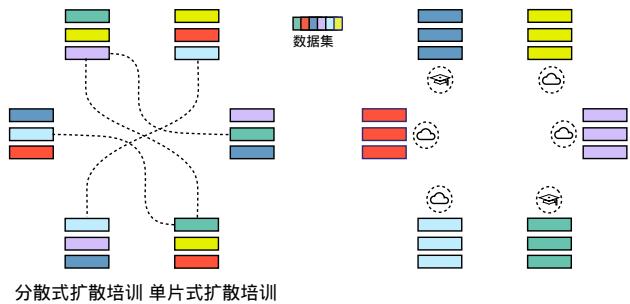


图 1. 分散式扩散模型 (DDM)。左图：现有的扩散模型（单体）需要在数千个 GPU 上进行同步集中训练，这使得高质量的训练系统既昂贵又难以使用。右图 DDM 将扩散模型划分为专家模型集合，每个专家模型在完全独立的数据集群上进行训练。该模型集合与在所有数据上训练的单一模型一样，都能优化相同的扩散目标。这样就能在不同的云计算或学术计算设施上进行灵活的训练。在推理过程中，该集合以相同的 FLOP 成本提高了性能，从而使高质量的扩散模型训练更加高效和容易获得。大规模 DDM 模型样本见图 2。

视频扩散模型对这些资源的需求呈阶跃式增长。例如，Meta 的 “Movie Gen” 在多达 6,114 个 H100 GPU 上进行训练[34]—几乎达到 GPT-3 的规模[6]。随着这些模型的规模和能力不断增长，它们会遇到与 LLM 训练相同的重大系统级挑战。高带宽互联成为关键瓶颈，存储系统必须处理海量流数据集，持续的硬件故障会导致训练过程崩溃或无声无息地减慢[1, 14]。这就形成了一个脆弱的集成系统，其性能和可靠性取决于加速器阵列和网络硬件之间复杂的相互作用。这对大型工业实验室来说具有挑战性，而且成本高昂，对学术界来说则难以承受。我们提出了分散扩散模型，这是一种可扩展的方法，可将扩散建模的负担分配给独立的专家模型，每个专家模型都在自己的“计算岛”上训练，没有交叉通信。这样，就可以利用分散的资源进行训练，利用具有成本效益的云计算，或在多个专家模型之间整合资源。



图 2 分散式扩散模型在现成的硬件上进行训练，并生成高质量、多样化的图像。我们展示了 8x3B 参数模型的部分样本。

分散化也有助于训练 "基础" 规模的模型 [3]。分散化还有助于训练 "基础" 规模的模型 [3]，在这种情况下，越来越难以建立具有足够功率容量的数据中心来满足现代训练运行的规模 [30]。独立训练还能带来更多好处，比如可以在异构硬件上执行，从而可以在更新的基础设施上重复使用现有的训练集群。分散式扩散模型采用了一种新的训练目标--分散式流匹配 (DFM)，它将训练数据分成 K 组，并对每组训练一个专门的扩散模型。受专家混合 (MoE) 文献[16]的启发，我们将这些专用模型称为专家。一个独立训练的路由器模型会对这些专家进行控制，确定哪些专家在测试时最合适。我们的研究表明，这种专家模型的集合优化了与在整个数据集上训练的单一模型相同的全局目标。在每个推理步骤中，路由器都会预测每个前摄体与输入噪声和条件的相关性。然后，专家预测会根据相关性的相关性得分进行线性组合。专家们学会了专业化，因此许多专家与给定输入不相关，使用其中的一个子集会更有效。如果不是所有专家都被选中，它就会成为一个稀疏模型，利用类似于 MoE 的选择性计算。稀疏推理的计算效率很高，但仍然需要大量内存。我们证明，我们可以将专家分解为一个单一的密集模型，从而在保证样本质量的同时实现便捷的推理。我们在 ImageNet 和 LAION-Aesthetics 的过滤子集上评估了我们的方法，并将 DDM 与现有的单一扩散模型训练进行了比较。我们系统地改变了专家的数量，以分析其对下游生成任务的影响，结果发现有 8 位专家的 DDM 始终能达到最佳性能，甚至优于用相同计算方法训练的单一模型。这得益于多位专业专家提供的额外参数，使其性能优于通用模型。最后，我们用 8 个 30 亿参数专家训练了一个大型分散模型，从而生成了高清图像，展示了分散扩散模型的扩展能力，如图 2 所示。分散式扩散模型还能轻松集成到现有的扩散训练环境中。在实践中，分散扩散模型需要对数据集进行聚类，然后在每个聚类上训练一个标准扩散模型。这意味着现有扩散基础架构中的几乎所有内容都可以重复使用--训练代码、数据加载、系统优化、噪声调度和架构。有关可视化演示和伪代码，请参阅我们的博文。

2. 相关作品

加速扩散模型流程匹配 [26、
27] 对扩散模型进行了概括 [20, 43 - 45]，并使

连续归一化流 (CNFs) 的大规模训练 [8]。扩散和流匹配模型可以在高维连续空间中生成最先进的模型，但要训练出高质量的模型通常成本高昂。最近的研究提出了标记掩蔽 (token masking) [42] 和将内部状态与预训练的图像表征模型对齐 (aligning internal states with pretrained image representation models) [51] 来加速模型学习过程。PixArt- α [7] 表明，带有合成标题的数据集可以降低训练成本。我们探索了一个互补的方向，即在多个 GPU 集群上分散训练模型，无需交叉通信，从而提高训练的鲁棒性和效率。现有方法都使用数据并行训练，即在 GPU 上分配批次并同步梯度，以产生更大的有效批次规模。我们的工作引入了一种正交形式的并行性，将模型的训练分割到独立的计算中心，不进行梯度同步。我们的方法与现有技术相辅相成--事实上，我们在每个集群内采用了完全分片数据并行 (Fully Sharded Data Parallel, FSDP) [52] 训练，同时在实验中保持了集群之间的隔离。

专家混合物 专家混合物 (MoE) 是一种流行而强大的方法，可在不相应增加计算成本的情况下提高模型容量。这是通过稀疏参数激活实现的：MoE 用一个轻量级路由器取代了转换器的每个密集前馈网络 (FFN) 层，该路由器从 N 个学习的 FFN 层中选择 k 个。这种方法在 Switch Transformers 中大放异彩 [16]，随后又通过各种路由策略 [53] 和系统改进 [17] 重新完善了 MoE。这些研究进展为语言建模带来了显著的成功，例如 Mixtral [23] 和 DeepSeek-V3 [11] 等模型，它们在保持计算效率的同时实现了强大的性能。虽然我们的工作重点是分散训练，但我们从测试时的 MoE 技术中汲取灵感，将通过稀疏激活专家来增加参数化的原理应用于扩散领域。MoE 和 DDM 都能在不增加计算成本的情况下提高模型总容量，但我们将通过将输入路由到不同的数据专家而不是将标记路由到不同的 FFN 来实现这一目标的。MoE 架构带来了巨大的系统挑战，尤其是专家负载平衡、容量因子管理问题，以及在训练和推理过程中在内存中保存多倍参数的挑战。与此相反，DDM 通过对专家和路由器进行单独训练，减轻了系统限制，从而将难度训练扩展到随时可用的硬件配置。

低交流学习 联合学习 [29] 建立了交流的基础技术。

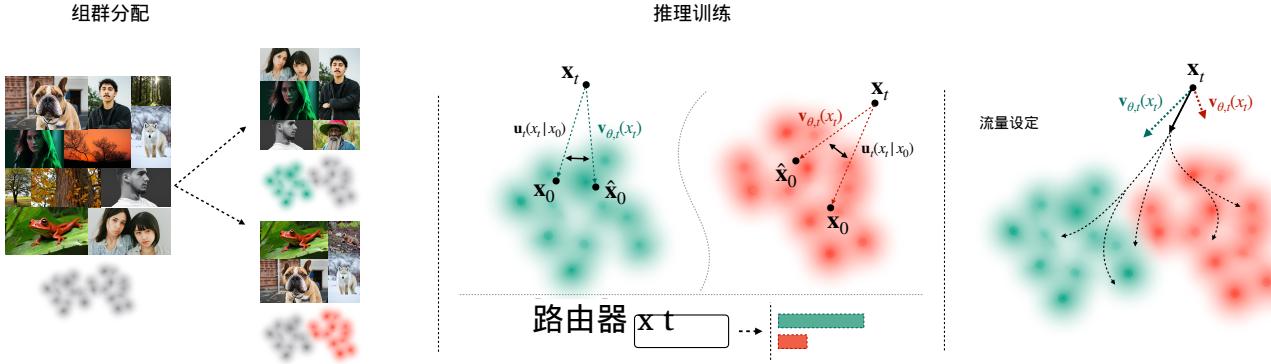


图 3. 分散式扩散模型 (DDM) 培训概述。DDM 的训练过程分为三个步骤。首先，我们使用现成的表征提取模型对数据集进行聚类。我们在每个聚类上训练一个扩散模型，并训练一个路由器，将任何输入 x_t 与其最可能的聚类相关联。测试时，给定一个有噪声的样本，每个专家（红色和绿色）预测自己的流量，并通过路由器预测的权重线性组合。合并后的流量会对整个分布进行采样，如右图所示。

受限训练，最初是出于对数据隐私的考虑。联邦平均[47]、自适应联邦优化[37]和 DiLoCo[13]等后续研究对这些想法进行了调整，以解决大规模训练的难题。DiLoCo 使用内部和外部优化循环来平衡局部训练和周期性全局同步。这与大型模型训练的发展相似，Branch-Train-Merge [25] 和 Diffusion Soup [2 , 49] 等方法已经探索了训练和组合数据专家模型的方法。虽然我们的方法在技术上与这些方法有相同之处，但我们的重点是利用分布式计算实现稳健性和灵活性，而不是数据主权或在移动设备上进行训练。事实上，我们的方法可以补充这些方法的不足，进一步实现去中心化，我们希望在未来的工作中看到这一点。

3.分散式流量匹配

我们引入了分散式流量匹配 (DFM) -- DDM 的训练目标--将扩散建模的负担分配给一组专家级去噪器，每个专家级去噪器在没有交叉通信的情况下单独训练。我们的研究表明，流量匹配目标很自然地融入了这种分散安排。由于扩散模型和整流可视为流量匹配的特例，因此 DFM 可直接应用于这些流行算法。具体来说，我们通过数据分布的一个预先确定和分片的子集来训练每个专家。然后，每个专家预测的流量会在测试时通过路由器进行组合，路由器会以简单的分类目标进行训练。这就产生了一个对整个数据分布进行采样的集合。

3.1.初步：流量匹配目标

流量匹配 [26] 定义了一个固定的正向损坏过程和一个学习的反向去噪过程。我们将此

时间步长为 t 的过程，插值 $t = 0$ 时的数据分布和 $t = 1$ 时的每个像素高斯分布。具体来说， $x_t = \alpha_t x_0 + \sigma_t \epsilon$ ，其中缩放系数 α_t 和 σ_t 是提前选择的，通常是为了保持跨时间步的方差 [45]。学到的模型 $v_{\theta,t}(x_t)$ 会逆转这一损坏过程，从噪声分布传输到数据分布。这些传输路径统称为边际流 $u_t(x_t)$ ，它是每个时间步的矢量场，可以在模型中回归以插值数据分布，或在数据集上分析计算，从而再现训练样本。流通过多步采样过程将潜变量 x_t 传递到数据分布中。在每一步中，模型都会预测 x_t 流向数据分布的加权平均值。在连续定义中，我们将其表示为一个积分：

$$u_t(x_t) = \int_{x_0} u_t(x_t|x_0) \frac{p_t(x_t|x_0)q(x_0)}{p_t(x_t)} dx_0 \quad (1)$$

其中 $u_t(x_t)$ 是边际流量， x_t 是我们的噪声潜变量， x_0 是数据样本， $p_t(x_t|x_0)$ 是高斯变量， $u_t(x_t|x_0)$ 是从 x_0 到 x_t 的条件流量。流量匹配通过参数模型对 u_t 的分析形式进行回归。由于我们对离散数据集进行训练，因此积分变成了总和、

$$u_t(x_t) = \frac{1}{p_t(x_t)} \sum_{x_0} u_t(x_t|x_0)p_t(x_t|x_0)q(x_0). \quad (2)$$

3.2.分散式流量匹配目标

DFM 将边际流量分解为一系列前流量。我们将数据划分为 K 个互不相交的簇 $\{S_1, S_2, \dots, S_K\}$ ，每个专家在指定的子集 (S_i) 上进行训练。每个专家对指定的子集 ($x_0 \in S_i$) 进行训练。这是一个自然的选择，因为经验表明

研究结果表明，图像数据位于流形的不连续结合部 [5 , 24]，这可能是扩散模式在图像领域有效的原因之一 [48]。对这些子集进行分割可以得到以下形式的边际流：

$$u_t(x_t) = \sum_{k=1}^K \frac{1}{p_t(x_t)} \sum_{x_0 \in S_k} u_t(x_t|x_0) p_t(x_t|x_0) q(x_0). \quad (3)$$

最后，我们发现边际流量可以写成专家流量的线性组合、

$$u_t(x_t) = \underbrace{\sum_{k=1}^K \frac{p_{t,S_k}(x_t)}{p_t(x_t)}}_{\text{Router}} \underbrace{\sum_{x_0 \in S_k} \frac{u_t(x_t|x_0) p_t(x_t|x_0) q(x_0)}{p_{t,S_k}(x_t)}}_{\text{Data-Expert Flow}}, \quad (4)$$

其中，外部总和是所有专家的分类概率分布，我们称之为 "路由器"，内部总和是每个子集的边际流量，我们称之为数据-专家流量。关于基于分数匹配的另一种推导方法，请参阅附录。这种划分对于大规模训练非常方便。我们可以训练多个独立的专家去噪器，而不是一次单一的训练运行。我们以标准的流量匹配目标和零交叉模型通信来训练每一个模型。另外，我们训练一个路由器，预测 x_t 从每个数据子集中提取的概率，这可以表述为我们下面描述的分类任务。与 MoE 类似，这个学习到的路由器在推理过程中将计算委托给不同的参数。一个关键的不同点是，我们的路由器可以单独进行明确的训练，而不是在梯度流经整个模型的情况下进行端到端的训练。在推理时，专家们会组合在一起，如上所示，他们会共同优化与单体训练相同的全局目标。

3.3. 路由器培训

路由器会预测给定 x_t 从每个数据分区中提取的概率。

$$p(k|x_t, t) = \frac{p_{t,S_k}(x_t)}{p_t(x_t)}, \quad k \in [K] \quad (5)$$

为了学习这一点，我们对输入 x_t 进行采样，然后利用与每个数据样本相关的聚类标签的交叉熵损失进行监督。当模型 $r_\theta(x_t, t)$ 能准确预测 $p(k|x_t, t)$ 时，损失最小。详见算法 1。请注意，路由器的训练与去噪器无关。我们用一个小小的扩散转换器 (DiT) [32] 来设置 $r_\theta(x_t, t)$ 的参数，并使用与 DiT 去噪器相同的调节机制。我们将学习到的分类标记[12]添加到我们的解码器中，并按照惯例通过线性头对 logits 进行聚类。

算法 1 流量路由器训练

要求 训练数据 { x_0, k } 时间表 { α_t, σ_t } $T t = 1$

1: 当未收敛时 do

2: $x_0, k \sim D$

4: $\epsilon \sim \mathcal{N}(0, \mathbf{I})$

5: $x_t = \alpha_t x_0 + \sigma_t \epsilon$

6: $\mathbf{z} = r_\theta(x_t, t)^{|K|}$

7: 使用 $\nabla \theta \text{ LCE}(\mathbf{z}, \text{OneHot}(\mathbf{k}))$ 更新 θ

8: 同时结束

3.4. 专家培训

DFM 目标将训练任务划分为学习一系列专家去噪器。方便的是，这些专家采用与标准流量匹配相同的目标：

$$\mathcal{L}_{\text{CFM}}(\theta) = \mathbb{E}_{t, q(x_0), p_t(x_t|x_0)} \|\mathbf{v}_{\theta,t}(x_t) - \mathbf{u}_t(x_t|x_0)\|^2. \quad (6)$$

我们将每个专家参数化为自己的 DiT，但需要注意的是，我们的方法与架构无关。只要能优化上述目标，我们甚至可以为每个去噪器分配不同的架构。

3.5. Inference Strategy

测试时，我们按照路由器预测的权重合并专家的预测：

$$u_t(x_t) = \underbrace{\sum_{k=1}^K r_\theta(x_t, t)}_{\text{Router}} \underbrace{v_{\theta,t}(x_t)}_{\text{Expert}}. \quad (7)$$

虽然要完全匹配全局流量匹配目标，必须要有完整的专家集合，但在每次测试时间预测中使用专家子集或甚至只使用一个专家的成本要低得多。模型的 FLOP 成本与每个前向传递中的有效专家数量成线性比例。这就需要在理论正确性和计算效率之间做出权衡。我们可以将这种选择理解为模型稀疏性的实例化，即模型在每个前向传递中只激活其参数的一个子集。在 MoE 中，这被证明可以在相同的计算预算下提高性能[16]。我们对不同的推理策略进行了实证比较，概述了这种效率与正确性的权衡，发现只选择顶级专家是最有效的方法，而且不会牺牲质量。

3.6. Distillation

在生产系统中，很难部署参数数量极高的模型。在测试时进行单个专家选择的 DDM 模型，每次前向传递的 FLOPs 几乎与单体模型相同，尽管它具有以下特点

$|K|$ 倍的参数。不过，它所需的内存仍是最初加载的单体模型的 $|K|$ 倍。随着模型规模超过当今加速器的内存预算，这将成为一个具有挑战性的分布式系统问题。虽然这在很多情况下是可以实现的，但我们提出了一种蒸馏方法作为便捷的替代方案。具体来说，稀疏 DDM 模型可以被蒸馏为密集模型。蒸馏后，推理就可以像其他以非分散方式训练的扩散模型一样进行。需要注意的是，这里蒸馏的目的是将 K 个专家缩减为一个模型，而不是像其他扩散蒸馏方法那样减少去噪步骤的数量 [40, 46, 50]。我们遵循师生训练流程 [18]。具体来说，我们用教师模型的预测替换条件流 $u_t(x_t | x_0)$ 监督目标：

$$L_{\text{distill}}(\theta) = E_{t,q(x_0)} p_t(x_t | x_0) \| v_{\theta,t}(x_t) - v_{\text{teacher},t}(x_t) \|_2.$$

在我们的案例中，我们使用分配给每个数据点的聚类标签来为每个训练实例选择一名教师专家。

4. 实验

我们在学术界的 ImageNet 数据集和 LAION [41] 数据集的一个子集上评估了分散扩散模型的有效性，该数据集的审美分数为 5 分或更高，更接近真实世界的训练场景。我们的目标是分析适用于广泛应用环境的十级扩散模型的设计空间，以便从业人员能够自信地调整方法，满足他们的需求。因此，我们尽可能使用标准架构、超参数和数据。

4.1. 实施细节

数据集分区 我们以高效的方式对数据集进行分区，以促进分散式流量匹配目标的学习。自然 k-means 算法的规模与输入数据的大小成二次方关系，这对于海量互联网数据集来说是不可行的。Ma 等人 [28] 提出了一种多阶段算法，可有效地将大量细粒度聚类合并为少量分区。我们采用这种方法，将工作重点放在生成建模上。按照 [28] 的方法，我们通过汇集 DINOv2 [31] 的输出结果来计算数据集中图像的图像特征，将像素特征和语义结合起来，将这些特征聚类为 1024 个细粒度中心点，然后进一步聚类为 k 个粗粒度中心点。我们将每个数据点分配给最接近的粗中心点，以生成最终的分区集。与训练相比，这个分区过程的计算量可以忽略不计。

评估 我们对 FID 计算中的已知敏感性进行了控制，这些敏感性来自于不同的实现方式和评估分割。我们从每个训练数据集中指定了一个包含 50,000 个样本的固定评估集，用于计算特征

推理策略	GFLOPs ↓	FID ↓	CLIP FID ↓
Monolith	308	12.81	5.58
Oracle	308	10.46	5.83
Full	2490	10.52	5.83
Top-1	334	9.84	5.48
Top-2	642	10.31	5.74
Top-3	950	10.37	5.77
Sample-1	334	157.05	51.17
Sample-2	642	10.27	5.73
Sample-3	950	10.44	5.78
Threshold-0.01	-	10.46	5.81
Threshold-0.05	-	10.37	5.75
Threshold-0.1	-	10.15	5.72
Nucleus ($T = 0.5$)	334	188.66	60.09
Nucleus ($T = 1.0$)	334	152.16	48.37
Nucleus ($T = 2.0$)	334	33.9	14

表 1. 测试时的组合策略 我们分析了测试时从集合中采样的策略，发现简单地选择最优秀的专家比选择更复杂的策略更有效。

并对所有模型进行评估。我们的 FID 数据与已发表的结果非常吻合，而且我们一致的实施方法可对不同方法进行精确的相对比较。这种标准化消除了通常使生成模型比较复杂化的混杂变量。

训练细节 在 ImageNet 实验中，我们采用了扩散变换器的超参数和架构 [32]，使用的批次大小为 256，EMA 衰减率为 0.9999，学习率为 $1e-4$ ，无预热或衰减。我们的目标是在 LAION 实验中复制真实世界中可信的训练运行，因此我们将批量大小扩展到 1024。

我们为去噪模型重新实现了 DiT XL/2 架构，每个模型包含 895M 个参数。在分散扩散模型中，总参数数与专家数量成线性关系。然而，在使用单一专家选择时，推理过程中的计算成本保持不变。

在 LAION 的实验中，我们按照 Pix-Art Alpha [7] 架构实现了文本调节，并使用 SDXL 的 CLIP [9, 35] 模型通过交叉关注将文本纳入其中。路由器使用较小的 DiT B/2 架构（1.58 亿个半径），并增加了学习的 CLS 标记，该标记可线性解码为集群上的概率分布。

我们通过在分散式扩散模型和基线单体模型之间保持一致的计算来确保公平的比较。为此，我们只需将总批次大小平均分配给专家。例如，如果有 8 位专家，256 个单体模型的批次大小相当于 8 个 32 个专家的批次大小。这样，DDM 和基线之间的总训练 FLOP 就相等了。路由器会带来额外 4% 的训练 FLOPs 开销，我们在比较时也会考虑到这一点。

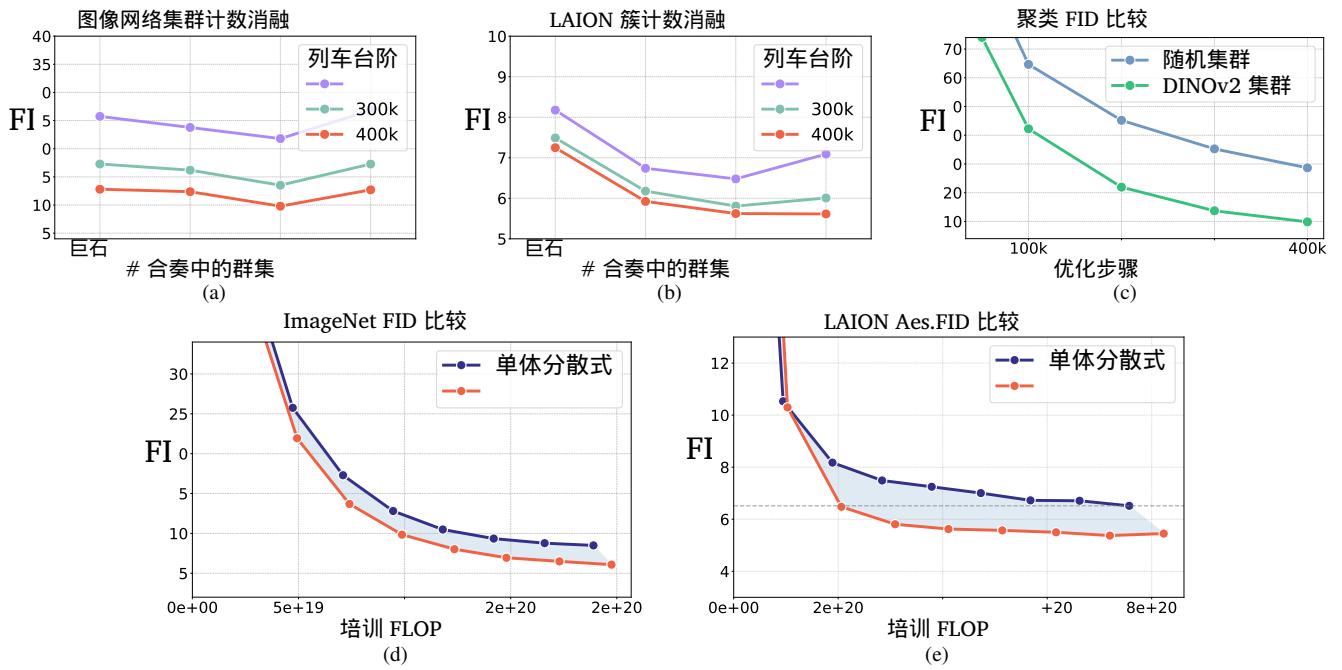


图 4. DiT XL 模型规模的消融情况。八专家 DDM 在 ImageNet (a) 和 LAION Aesthetics (b) 上表现出最佳的一致性。与随机聚类相比，我们展示了 ImageNet 上基于图像的聚类的重要性 (c)。最后，在两个数据集上，分散扩散模型的 FLOP 对 FLOP 性能均优于整体扩散模型 (d、e)。

4.2. 测试时的集合

我们首先比较了在测试时组合专家预测的不同策略。对边际流量的全面估计涉及将所有专家预测进行线性组合：

$$u_t(x_t) = \sum_{k=1}^K r_\theta(x_t, t) v_{\theta, k}(x_t). \quad (8)$$

在实践中，只选择重要的专家可以节省计算工作。我们对以下策略进行了评估：

- 全。计算所有专家预测的加权组合。该策略的 FLOP 成本与专家数量成线性关系。
- 样本。从路由器预测的 softmax 分布中取样，选择一个专家。这是对边际流量的无偏蒙特卡洛估计。
- 顶 k。只需使用预测路由器概率最大的 k 个专家即可。Top-1 选择是测试时最有效的方案。
- 核心。根据大型语言模型中常用的核心 (top-p) 采样策略 [21]，对一名专家进行采样。我们在表 1 中使用 $p = 0.9$ 和消减 softmax 温度。
- 甲骨文。根据与评估图像相关的群组标签选择一位专家。仅用于评估学习路由器的有效性。



图 5. DDM 优化了全局扩散目标。我们使用带有匹配随机种子的确定性采样器 (左图) 对来自单一和 DDM ImageNet 模型的样本进行平均处理，并将它们与使用随机噪声样本生成的输出结果 (右图) 进行比较。左侧样本高度相关，看起来不那么模糊。

在表 1 中，我们在 ImageNet 上对这些推理策略进行了评估。我们发现 top-1 选择优于所有其他选择，同时产生的 FLOP 成本也最低。在图 4 和图 6 中的所有其他比较中，我们使用 top-1 选择，因为它几乎与密集模型的计算成本相匹配。

4.3. 选择合适数量的专家

DFM 目标在理论上支持任意数量的专家，但在实践中，我们发现这是 DDM 的一个重要超参数。这一选择决定了系统的分散程度和总参数数。在理论极限中，当专家数量与训练样本数量相近时，系统将简化为近邻查找，只能复制

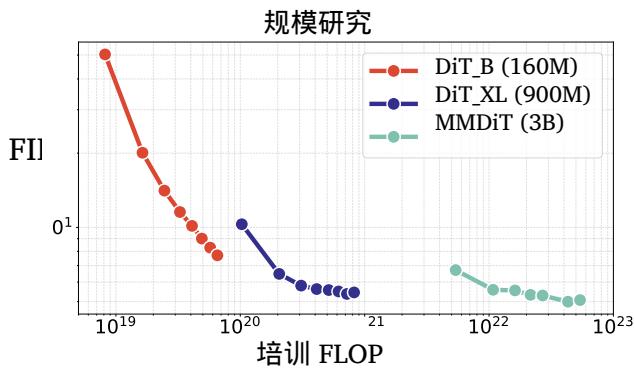


图 6. 分散式扩散模型可以优雅地扩展到数十亿个参数。在整个训练过程中，我们绘制了 LAION Aesthetics 的 FID 与训练计算量的函数关系图。我们发现，专家模型容量和训练计算量的增加可预测地提高性能。

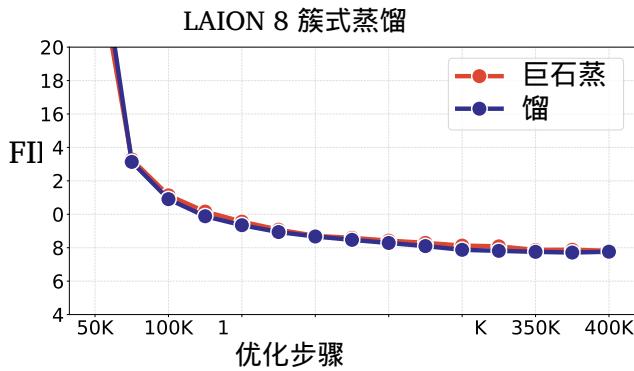


图 7. 将 DDM 提炼为密集模型。在生产环境中，密集模型通常比稀疏模型更方便使用。将分散扩散模型蒸馏为密集模型，其性能与从头开始训练单体的性能相当，而 FLOP 成本仅为单体的三分之一（批量大小为单体的 1/4）。

训练示例--相当于流量匹配的分析形式。我们还发现，由于全局批量规模划分过大，单个专家的训练效果很差，如图 4a 中的 16 个专家（每个专家的批量规模为 16）实验。我们在图 4a 中比较了 4、8 和 16 个专家的 DDM

我们发现，在 ImageNet 和 LAION 上，八位专家始终能达到最佳性能。这种配置既能实现强大的去中心化，又能保持合理的测试时间内存要求。八位专家在保持数据分布一致性的同时，似乎还进行了有意义的专业化。根据经验，我们发现这是模型容量、去中心化和实际部署等竞争因素的最佳平衡点。

4.4. DDMs vs. Monoliths

在这两个数据集上，我们将分散扩散模型与公平的单一基线进行了比较。我们发现，由八位专家组成的分散式扩散模型始终优于由四位专家组成的分散式扩散模型。

在 FLOP 对 FLOP 的基础上形成标准扩散模型。在图 4d 中，我们比较了 ImageNet 上高达 800k 训练步数的 FID。在 800k 步时，分散扩散模型的 FID 降低了 28% (6.081 对 8.494)。请注意，我们绘制了 FID 与训练 FLOPs 的函数关系图，以考虑路由器 4% 的额外训练成本。我们发现，图 4e 中的 LAION 在有字幕的互联网数据上也有显著的性能提升。事实上，与单片机在 800k 步时的 6.52 FID 相比，LAION 在 200k 优化步时实现了更低的 6.48 FID。这意味着训练速度提高了 4 倍，收敛 FID 也降低了。我们还直观地验证了 DFM 目标的正确性。标准的扩散目标可预测噪声样本和数据样本的配对，因此训练有素的 DDM 在相同的输入噪声下，应能采样出与单片机相似的图像。我们在图 5 中验证了这一点。有关 DDM 的更多样本和分析，请参阅补充资料。

4.5. 数据聚类消融

DFM 对聚类大小或聚类位置没有明确限制，因此我们采用了两种聚类策略。我们发现，所选策略对模型的性能有很大影响。使用 DINO 进行基于特征的聚类与随机聚类分配进行比较（随机聚类分配将在各分区之间保持 i.i.d. 属性），发现基于特征的聚类能显著改善结果。我们假设，与随机聚类相比，特征聚类内部存在更多的相互信息，这意味着专家模型可以更有效地压缩和专门化其分配的子分布。当数据具有语义或低级特征相似性时，专家们就可以自由地学习更有针对性的表征。

4.6. Distillation

虽然我们的方法通过在推理时选择排名第一的专家实现了计算效率，但许多专家模型的总内存占用可能很大。我们通过知识提炼来解决这一局限性，将集合能力整合到一个密集模型中。我们的方法是用专家集合的预测结果监督学生模型，根据每个训练实例的聚类标签选择合适的专家。这可以看作是对前 1 个稀疏模型的提炼，其成本与标准的师生提炼模型相同。我们的蒸馏模型与直接在数据集上进行训练的性能不相上下，尽管只使用了批量大小的四分之一，因此也只使用了训练 FLOP 的三分之一（假设后向传递的成本是前向传递的两倍）。经过 40 万步训练后，蒸馏模型的 FID 为 7.76，与基线模型的 FID 7.82 相当（图 7）。许多扩散蒸馏工作都侧重于减少采样步数，而我们的目标只是在密集模型中复制集合。我们将

探索如何将我们的方法与以取样为重点的蒸馏技术相结合，是未来大有可为的工作。

4.7. 缩放实验

我们对分散式扩散模型进行了扩展研究。在每个规模上，我们都遵循从消融中总结出的最佳实践，并训练一个由八个专家模型组成的系统，每个模型都基于 FLUX MMDiT 架构[15]。我们使用的隐藏维度为 2560，深度为 30，并分别使用文本和视觉标记流，每个专家共使用 3B 个参数。我们使用单个 T5 XL 模型[36]对文本提示进行编码，并通过自我关注将其特征与图像特征混合在一起。最重要的是，每个专家都可以在现成的硬件上进行独立训练。在每个专家使用 16 个 GPU 的情况下，我们以每次迭代 0.28 秒的速度进行 100 万步预训练。使用梯度累积法，这相当于在一个按需云 GPU 节点上对每个专家进行了六天半的训练。这表明，我们的方法无需专门的基础设施或大型集成计算集群就能训练大规模扩散模型。我们在图 6 中评估了我们的大规模集合与较小的 DiT B 和 XL [32] 集合的对比情况。DDM 的性能随着专家参数化的增加而提高，并且在我们尝试的任何规模下都没有达到饱和。最后，我们在高分辨率数据上对最大集合进行了 60k 步的微调，并在图 2 中显示了一些选定样本。

5. 讨论

分散扩散模型能够在孤立的计算集群中进行高质量的生成模型训练，从而大大拓宽了扩散模型训练的硬件配置。虽然我们关注的是计算资源的分布，但从理论上讲，DFM 也允许数据的分散，这一特性对医疗成像等领域具有潜在的隐私影响。专家可以在敏感数据所在的地方进行训练，路由器可以根据这些专家的样本而不是原始数据进行训练。这些想法使得 DDM 能够保护数据隐私和主权，因为私人数据永远不会离开其原始位置。此外，将 DDM 与低带宽训练方法相结合，还能推动去中心化的发展--或许能在真正的商品硬盘上进行大规模模型训练。虽然我们对图像建模进行了实验，但 DDM 提出的原则可应用于其他领域，如医疗成像、机器人策略和视频建模。我们期待着未来在这些方向上的工作。

致谢。我们要感谢 Alex Yu 在整个项目中的指导以及他的分数匹配推导。我们还要感谢 Daniel Mendelevitch、Songwei Ge、Dan Kondratyuk、Haiwen Feng、Terrance De-Vries、Chung Min Kim、Hang Gao、Justin Kerr 和 Luma AI 研究团队的讨论。

参考资料

- [1] Wei An, Xiao Bi, Guanting Chen, Shanhua Chen, Chengqi Deng, Honghui Ding, Kai Dong, Qiushi Du, Wenjun Gao, Kang Guan, et al. Fire-flyer ai-hpc: A cost-effective software-hardware co-design for deep learning. *arXiv preprint arXiv:2408.14158*, 2024. 1
- [2] Benjamin Biggs, Arjun Seshadri, Yang Zou, Achin Jain, Aditya Golatkar, Yusheng Xie, Alessandro Achille, Ashwin Swaminathan, and Stefano Soatto. Diffusion soup: Model merging for text-to-image diffusion models. *arXiv preprint arXiv:2406.08431*, 2024. 4
- [3] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021. 3
- [4] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, et al. Video generation models as world simulators. 2024. URL <https://openai.com/research/video-generation-models-as-world-simulators>, 3, 2024. 1
- [5] Bradley CA Brown, Anthony L Caterini, Brendan Leigh Ross, Jesse C Cresswell, and Gabriel Loaiza-Ganem. Verifying the union of manifolds hypothesis for image data. *arXiv preprint arXiv:2207.02862*, 2022. 5
- [6] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and Sandhini Agarwal et al. Language models are few-shot learners, 2020. 1
- [7] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart- α : Fast training of diffusion transformer for photorealistic text-to-image synthesis, 2023. 1, 3, 6
- [8] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018. 3
- [9] Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Itsev. Reproducible scaling laws for contrastive language-image learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2818–2829, 2023. 6
- [10] Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, page 02783649241273668, 2023. 1
- [11] DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao,

- 李慧、曲慧、J. L.Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jiawei Wang, Jin Chen, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang、张明川、张明华、唐明辉、李明明、田宁、黄盼盼、王培毅、张鹏、王前程、朱启豪、陈钦宇、杜秋实、R.J. Chen, R. L. Jin, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runxin Xu, Ruoyu Zhang, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan Zhou, Shanhua Chen, Shaoqing Wu, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Shuting Pan, T. Wang, Tao Yun, Tian Pepper, T. T. T. Wang, Tao Yun, Tian Pei, Tianyu Sun, W. L. Xiao, Wangding Zeng, Wanjia Zhao, Wei An, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, X. Q.Q.Li, Xiangyue Jin, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaojin Shen, Xiaokang Chen, Xiaokang Zhang, Xiaosha Chen, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingkai Yu, Xinnan Song, Xinxia Shan, Xinyi Zhou, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, Y. K. Li, Y. Q. Wang, Y. K. Wang, Y. Q.Wang, Y.X.Wei, Y. X.X.Zhu, Yang Zhang, Yanhong Xu, Yanhong Xu, Yanping Huang, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Li, Yaohui Wang, Yi Yu, Yi Zheng, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Ying Tang, Yishi Piao, Yisong Wang, Yixuan Tan, 马逸阳、刘逸媛、郭永强、吴昱、欧媛、朱雨辰、王宇端、龚玥、邹宇恒、何雨佳、查玉坤、熊云帆、马云贤、严玉婷、罗玉香、游玉香、刘玉璇、周玉阳、Z.F. Wu, Z.Z.Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhen Huang, Zhen Zhang, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhipeng Xu, Zhiyu Wu, Zhongyu Zhang, Zhuoshu Li, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Ziyi Gao, and Zizheng Pan.Deepseek-V3 技术报告 , 2024 年。 3
- [12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 5
- [13] Arthur Douillard, Qixuan Feng, Andrei A Rusu, Rachita Chhaparia, Yani Donchev, Adhiguna Kuncoro, Marc'Aurelio Ranzato, Arthur Szlam, and Jiajun Shen. Diloco: Distributed low-communication training of language models. *arXiv preprint arXiv:2311.08105*, 2023. 4
- [14] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. 1
- [15] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entzari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first International Conference on Machine Learning*, 2024. 9, 12
- [16] William Fedus, Barret Zoph, and Noam Shazeer. Switch变电器：以简单高效的稀疏性扩展到万亿参数模型 机器学习研究期刊》, 23 (120) :1-39 , 2022。 3 , 5
- [17] Trevor Gale, Deepak Narayanan, Cliff Young, and Matei Zaharia. Megablocks: Efficient sparse training with mixture-of-experts. *Proceedings of Machine Learning and Systems*, 5: 288–304, 2023. 3
- [18] Geoffrey Hinton. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 6
- [19] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance, 2022. 12
- [20] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 3
- [21] Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration, 2020. 7
- [22] Emiel Hoogeboom, Jonathan Heek, and Tim Salimans. Simple diffusion: End-to-end diffusion for high resolution images, 2023. 12
- [23] Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024. 3
- [24] Hamidreza Kamkari, Brendan Leigh Ross, Rasa Hosseinzadeh, Jesse C Cresswell, and Gabriel Loaiza-Ganem. A geometric view of data complexity: Efficient local intrinsic dimension estimation with diffusion models. *arXiv preprint arXiv:2406.03537*, 2024. 5
- [25] Margaret Li, Suchin Gururangan, Tim Dettmers, Mike Lewis, Tim Althoff, Noah A Smith, and Luke Zettlemoyer. Branch-train-merge: Embarrassingly parallel training of expert language models. *arXiv preprint arXiv:2208.03306*, 2022. 4
- [26] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022. 3, 4, 12
- [27] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022. 3
- [28] Jiawei Ma, Po-Yao Huang, Saining Xie, Shang-Wen Li, Luke Zettlemoyer, Shih-Fu Chang, Wen-Tau Yih, and Hu Xu. Mode: Clip data experts via clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26354–26363, 2024. 6
- [29] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017. 3
- [30] Mark Morey. Data center owners turn to nuclear as potential electricity source - u.s. energy information administration (eia), 2024. 3
- [31] Maxime Oquab, Timothée Darcret, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 6

- [32] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023. 5, 6, 9
- [33] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 1
- [34] Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, et al. Movie gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024. 1
- [35] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 6
- [36] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020. 9
- [37] Sashank Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečný, Sanjiv Kumar, and H Brendan McMahan. Adaptive federated optimization. *arXiv preprint arXiv:2003.00295*, 2020. 4
- [38] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 1
- [39] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022. 1
- [40] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022. 6
- [41] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022. 6
- [42] Vikash Sehwag, Xianghao Kong, Jingtao Li, Michael Spranger, and Lingjuan Lyu. Stretching each dollar: Diffusion training from scratch on a micro-budget. *arXiv preprint arXiv:2407.15811*, 2024. 3
- [43] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015. 3
- [44] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.
- [45] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. 3, 4, 12
- [46] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. *arXiv preprint arXiv:2303.01469*, 2023. 6
- [47] Tao Sun, Dongsheng Li, and Bao Wang. Decentralized federated averaging. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):4289–4301, 2022. 4
- [48] Peng Wang, Huijie Zhang, Zekai Zhang, Siyi Chen, Yi Ma, and Qing Qu. Diffusion models learn low-dimensional distributions via subspace clustering. *arXiv preprint arXiv:2409.02426*, 2024. 5
- [49] Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, et al. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International conference on machine learning*, pages 23965–23998. PMLR, 2022. 4
- [50] Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6613–6623, 2024. 6
- [51] Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation alignment for generation: Training diffusion transformers is easier than you think. *arXiv preprint arXiv:2410.06940*, 2024. 3
- [52] Yanli Zhao, Andrew Gu, Rohan Varma, Liang Luo, Chien-Chin Huang, Min Xu, Less Wright, Hamid Shojanazeri, Myle Ott, Sam Shleifer, et al. Pytorch fsdp: experiences on scaling fully sharded data parallel. *arXiv preprint arXiv:2304.11277*, 2023. 3
- [53] Yanqi Zhou, Tao Lei, Hanxiao Liu, Nan Du, Yanping Huang, Vincent Zhao, Andrew M Dai, Quoc V Le, James Laudon, et al. Mixture-of-experts with expert choice routing. *Advances in Neural Information Processing Systems*, 35:7103–7114, 2022. 3

A. 得分匹配推导

我们提供了基于分数匹配[45]而非流量匹配[26]的分散流量匹配的另一种推导。我们从分数开始，分数是对数似然的梯度，即 $p_t(x_t)$ 。

$$\nabla_{x_t} \log p_t(x_t) \quad (9)$$

应用链式法则，这可以用可能性本身的导数来表示：

$$\nabla_{x_t} \log p_t(x_t) = \frac{1}{p_t(x_t)} \cdot \nabla_{x_t} p_t(x_t). \quad (10)$$

假设我们的数据 x_0 是以双层方式生成的：

$$x_0 \sim p_0(x_0|v), \quad v \sim p_v(v), \quad (11)$$

其中， v 是 DDM 方法中讨论的聚类标签， $p_v(v)$ 遵循聚类程序定义的分布。边际似然 $p_t(x_t)$ 可以通过对 v 进行积分来求得：

$$\nabla_{x_t} \log p_t(x_t) = \frac{1}{p_t(x_t)} \cdot \nabla_{x_t} \sum_v p(v) \cdot p_t(x_t|v). \quad (12)$$

通过线性微分，我们将梯度分布在求和上：

$$= \frac{1}{p_t(x_t)} \cdot \sum_v p(v) \cdot \nabla_{x_t} p_t(x_t|v). \quad (13)$$

由于对数概率的梯度可以表示为概率乘以其对数的梯度、

$$= \frac{1}{p_t(x_t)} \cdot \sum_v p(v) p_t(x_t|v) \cdot \nabla_{x_t} \log p_t(x_t|v). \quad (14)$$

最后，我们引用贝叶斯定理：

$$= \sum_v p_t(v|x_t) \cdot \nabla_{x_t} \log p_t(x_t|v) \quad (15)$$

这一结果反映了流量匹配的推导，表明对整体数据分布的得分预测可以重塑为每个数据集群得分预测的线性组合。每个学习到的专家都会预测自己的条件得分，而条件得分的组合是根据给定潜在 x_t 的专家标签的后验概率确定的。

B. 其他培训和评估细节

我们拆分的 LAION Aesthetics 包含 1.536 亿个图像-标题对。我们在通过 Huggingface 的微调稳定扩散 VAE (sd-vae-ft-mse) 编码的 256x256 平方裁剪图像上对所有扩散模型进行预训练。该编码器

采用 8 倍空间降采样因子。在整个训练过程中，为了获得最佳质量，我们将补丁大小保持在 2，因此训练前的上下文长度为 256。为了进行高分辨率微调，我们选择了五个宽高比桶来处理不同的图像尺寸，同时保持一致的标记化序列长度（3600）。这些比例桶如下 - 1280 × 720 (16:9 横向) - 1200 × 768 (3:2 横向) - 960 × 960 (正方形) - 768 × 1200 (2:3 纵向) - 720 × 1280 (9:16 纵向) 图像按纵横比映射到最接近的匹配桶。根据 Stable Diffusion 3 的最佳实践[15]，我们调整了高分辨率训练和推理的时间步计划，应用了 3 的对数-SNR 移位[22]。我们还修改了 MMDiT 架构中的旋转位置嵌入 (RoPE)，对中央方形区域内的 RoPE 输入进行内插，对外围区域进行外推。在评估中，我们使用了标准的无分类器引导尺度：LAION 文本条件生成为 7.5，ImageNet 类条件生成为 3。所有评估均使用 50 个采样步骤，以确保比较结果的一致性。我们在固定的评估分段上计算 FID、CLIP-FID 和 DINO-FID 指标，以实现评估的标准化。

C. 补充定量分析

我们的附加定量分析探讨了测试时间 DDM 的关键超参数。我们在表 2 中测试了各种集合组合，包括 LLM 解码中常见的核采样。Top-1 取样始终能提供最佳性能，同时计算效率最高。这一结论在不同的专家数量、路由器温度和阈值概率下都成立。我们的无分类器引导 (CFG) [19] 规模实验 (图 8) 显示，DDM 与单体模型的响应类似，这表明标准 CFG 规模可直接应用于 DDM。此外，我们的训练效率分析 (图 8e 和图 8f) 证实，蒸馏法的生成质量 (FID) 相当，而批量大小 (256 对 1024) 仅为整体模型的四分之一。

D. 其他定性结果

我们在图 9 中提供了从最大 DDM 组合中选取的其他样本，并在图 10 至图 17 中提供了不同文本提示的随机样本。

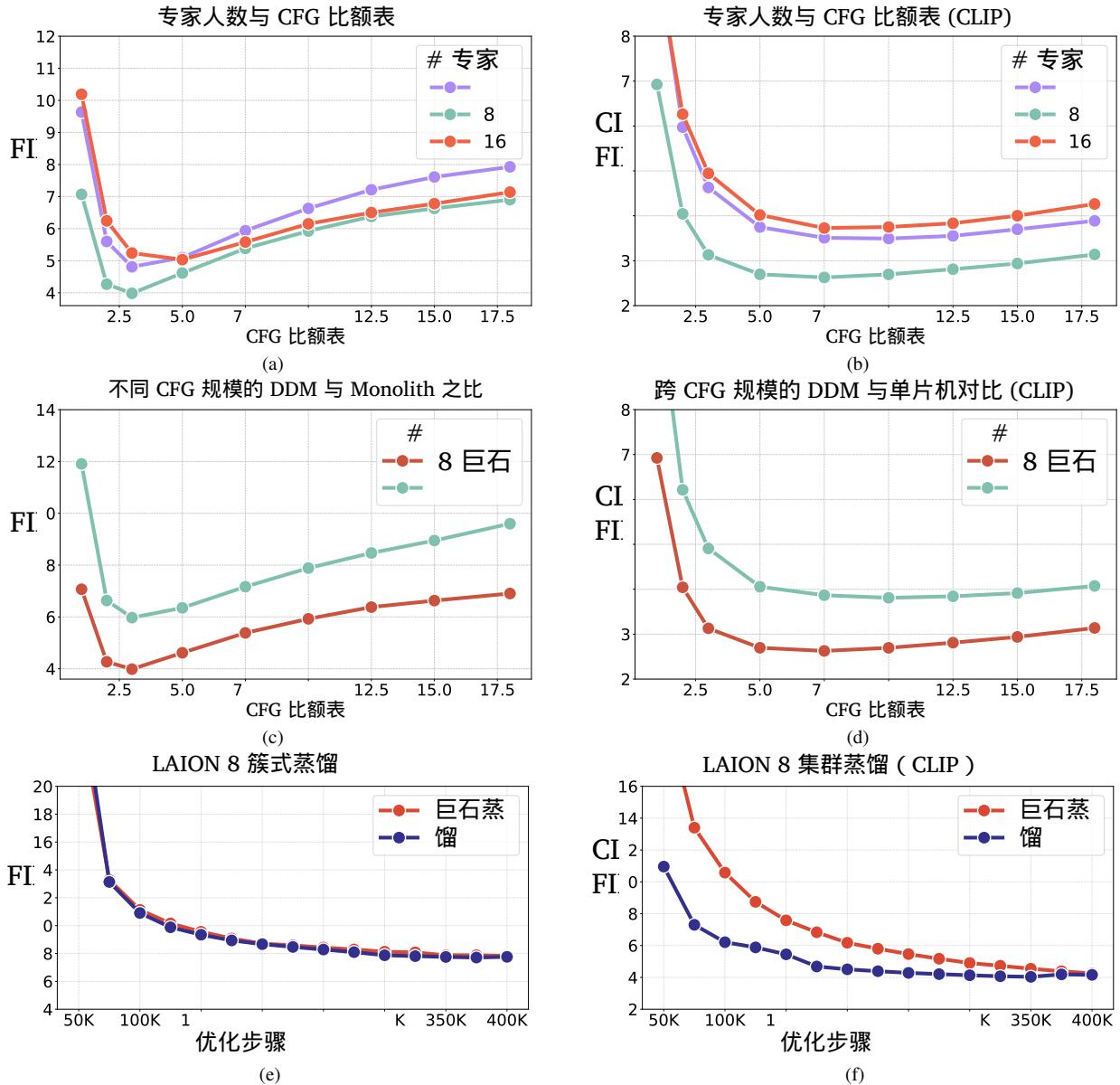


图 8. 附加定量分析。我们横扫了在 LAION Aesthetics (a, b, c, d) 上训练的分散和单体扩散模型的 CFG 规模，发现不同模型的最佳 CFG 规模是一致的。蒸馏法的性能与根据原始数据训练单块模型的性能相当，而 FLOP 成本仅为后者的一小部分 (e, f)。

推理策略		专家数量	Temp.	活跃专家 p	GFLOPs ↓	FID ↓	CLIP FID ↓	DINO FID ↓
Monolith	1	-	1	0.00	308	12.81	5.58	343.96
Full	4	-	4	0.00	1245	12.75	6.82	386.3
Top-1	4	-	1	0.00	334	12.54	6.72	378.4
Top-2	4	-	2	0.00	642	12.76	6.75	384.12
Top-3	4	-	3	0.00	950	12.88	6.8	385.79
Sample	4	0.5	1	0.00	334	117.02	36.25	1321.17
Sample	4	1.0	1	0.00	334	89.14	28.01	1042.09
Sample	4	2.0	1	0.00	334	15.08	7.53	425.29
Sample	4	0.5	2	0.00	642	12.67	6.71	380.85
Sample	4	1.0	2	0.00	642	12.67	6.73	382.93
Sample	4	2.0	2	0.00	642	13.51	7.05	400.08
Sample	4	0.5	3	0.00	950	12.57	6.76	381.44
Sample	4	1.0	3	0.00	950	12.84	6.81	385.18
Sample	4	2.0	3	0.00	950	13.59	7.08	400.23
Nucleus	4	0.5	1	0.90	334	15.74	9.99	412.49
Nucleus	4	1.0	1	0.90	334	15.72	10.04	411.31
Nucleus	4	2.0	1	0.90	334	17.35	10.31	432.46
Threshold	4	1.0	-	0.01	-	12.82	6.81	385.44
Threshold	4	1.0	-	0.05	-	12.67	6.73	382.54
Threshold	4	1.0	-	0.10	-	12.63	6.76	382.86
Full	8	-	4	0.00	2490	10.52	5.85	354.15
Top-1	8	-	1	0.00	334	9.85	5.54	339.56
Top-2	8	-	2	0.00	642	10.33	5.73	349.28
Top-3	8	-	3	0.00	950	10.45	5.77	351.91
Sample	8	1.0	1	0.00	334	190.95	59.03	2105.79
Sample	8	2.0	1	0.00	334	184.06	50.55	1790.24
Sample	8	0.5	2	0.00	642	9.93	5.57	343.51
Sample	8	1.0	2	0.00	642	10.28	5.72	348.39
Sample	8	2.0	2	0.00	642	17.11	8.09	471.18
Sample	8	0.5	3	0.00	950	10.04	5.62	342.86
Sample	8	1.0	3	0.00	950	10.42	5.78	350.91
Sample	8	2.0	3	0.00	950	12.06	6.38	380.54
Nucleus	8	0.5	1	0.90	334	188.66	60.09	2110.22
Nucleus	8	1.0	1	0.90	334	152.16	48.37	1609.23
Nucleus	8	2.0	1	0.90	334	33.9	14	682.31
Threshold	8	1.0	-	0.01	-	10.51	5.82	351.17
Threshold	8	1.0	-	0.05	-	10.32	5.73	349.86
Threshold	8	1.0	-	0.10	-	10.18	5.7	346.9
Full	16	-	4	0.00	4980	15.43	7.57	440.54
Top-1	16	-	1	0.00	334	12.51	6.6	397.99
Top-2	16	-	2	0.00	642	148.26	41.13	1535.85
Top-3	16	-	3	0.00	950	91.76	29.53	1105.92
Sample	16	1.0	1	0.00	334	232.1	71.88	2557.41
Sample	16	2.0	1	0.00	334	259	81.49	2797.76
Sample	16	0.5	2	0.00	642	161.29	47.76	1732.88
Sample	16	1.0	2	0.00	642	174.23	54.41	1941.49
Sample	16	0.5	3	0.00	950	119.84	40.18	1510.78
Sample	16	1.0	3	0.00	950	44.62	20.04	772.02
Sample	16	2.0	3	0.00	950	26.01	10.92	603.31
Threshold	16	1.0	-	0.01	-	14.92	7.44	431.89
Threshold	16	1.0	-	0.05	-	12.62	6.61	399.09
Threshold	16	1.0	-	0.10	-	12.69	6.61	398.33

表 2. 测试时的组合策略。我们删除了测试时从 ImageNet DDM 组合中采样的策略和相关超参数。在多次实验中，我们发现简单地选择顶级专家的效果优于更复杂的替代方案。

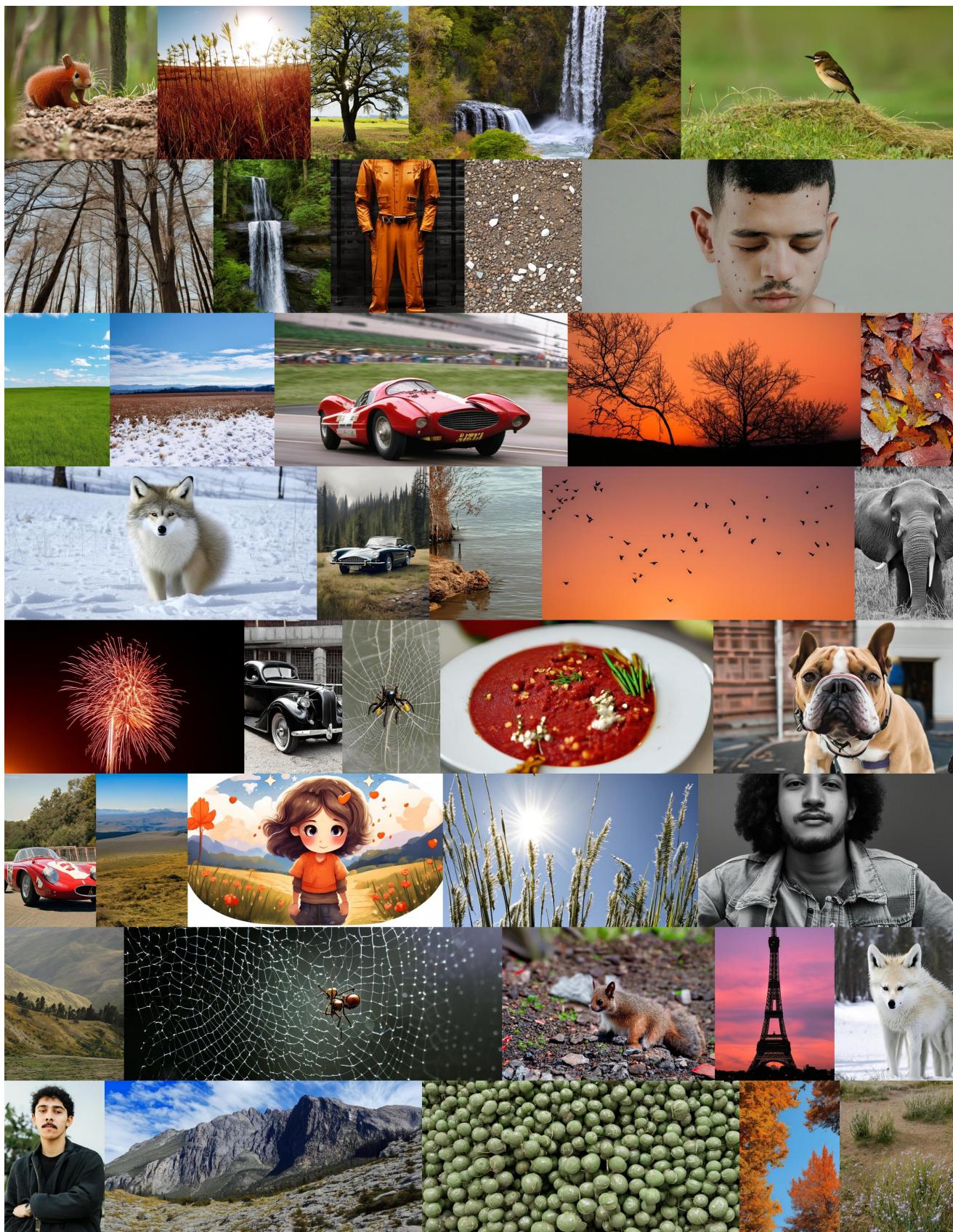


图 9. 其他选定样本。



图 10. 白云岩照片



图 11. 随机样本，固定提示。 1969 年 Polaris Colt，修复后的陈列室，雪中静态展示，冬日日出



图 12. 极端条件下的气象研究站、监测设备、自然因素



图 13. 随机样本，固定提示。古老的硬毛松森林，扭曲的树木，高海拔的光线，崎岖的山脉背景



图 14. 随机样本，固定提示。沙漠深槽峡谷，砂岩纹理，光轴，自然色彩渐变



图 15. 随机样本，固定提示。哥特式大教堂尖顶刺破晨雾，欧洲古城屋顶景观



图 16. 随机样本，固定提示。历史悠久的纺织厂内部，保存完好的机器，阳光穿过工业窗户



图 17. 随机样本，固定提示。排练中的交响乐团，指挥视角，历史悠久的音乐厅