

流得又直又快： 用 Rectified Flow 学习数据的生成与迁移

Flow Straight and Fast: Learning to Generate and Transfer Data with Rectified Flow

Xingchao Liu*
University of Texas at Austin
xcliu@utexas.edu

Chengyue Gong*
University of Texas at Austin
cygong@cs.utexas.edu

Qiang Liu
University of Texas at Austin
lqiang@cs.utexas.edu

摘要

我们提出 Rectified Flow（整流流），这是一种出人意料地简单的方法，用于学习（神经）常微分方程（ODE）模型，从而在两个经验观测到的分布 π_0 与 π_1 之间进行传输，进而为生成式建模、领域迁移，以及其他各类涉及分布传输的任务提供一个统一的解决方案。Rectified Flow 的思想是：让 ODE 尽可能沿着连接 π_0 与 π_1 中样本点的直线路径行进。这一目标通过求解一个简单的非线性最小二乘优化问题即可实现，且除了标准监督学习之外无需引入额外参数，因而可以轻松扩展到大模型。直线路径之所以特殊且更受青睐，是因为它是两点之间的最短路径，并且无需时间离散化即可被精确模拟，从而得到计算上高效的模型。我们证明：从数据中学习一个 Rectified Flow 的过程（称为整流，rectification）会把 π_0 与 π_1 的任意耦合转化为一个新的确定性耦合，且其凸传输代价可证明是不增的。此外，递归地施加整流，可以得到一系列路径越来越直的流；在推理阶段，即便使用较粗的时间离散化，这些流也能被准确地模拟。在实验研究中，我们表明 Rectified Flow 在图像生成、图像到图像翻译以及领域自适应上都表现优异。特别是在图像生成与翻译任务上，我们的方法给出近乎笔直的流，即便只用单步欧拉离散化也能得到高质量的结果。

*XL and CG contributed equally to this work.

1 引言

相比监督学习，无监督学习的共同难题是缺乏配对的输入/输出数据，标准回归或分类任务难以适用。大多数无监督方法的目标是在两个分布间找到有意义的对应关系。例如，生成对抗网络（GAN）和变分自编码器（VAE）等生成模型 [e.g., 19, 32, 14] 将数据点映射到遵循简单初等（高斯）分布的潜在编码，进而生成和操纵数据。表示学习基于这样的想法：足够光滑的函数可将结构化数据分布映射到初等分布，进而赋予语义上有意义的解释，对下游学习任务有用。领域迁移方法在两个经验观察的数据分布间寻找映射，用于图像到图像翻译、风格转移和领域自适应 [e.g., 100, 16, 79, 59]。这些任务都可统一视为在两个分布间寻找传输映射：

传输映射问题 给定两个分布 $X_0 \sim \pi_0, X_1 \sim \pi_1$ 在 $\mathbb{R}^d$ 上的经验观察，找到一个传输映射 $T: \mathbb{R}^d \rightarrow \mathbb{R}^d$ （最好在某种意义上是好的或最优的），使得当 $Z_0 \sim \pi_0$ 时， $Z_1 := T(Z_0) \sim \pi_1$ ，即 (Z_0, Z_1) 是 π_0 和 π_1 的耦合（也称传输方案）。

根据如何表示和训练映射 T ，已发展出多条技术路线。在传统生成模型中， T 用神经网络参数化，通过 GAN 型极小极大算法或（近似）最大似然估计（MLE）进行训练。然而，GAN 已知存在数值不稳定性和模式崩溃问题，需要大量工程投入和人工调参，往往难以跨不同模型架构和数据集迁移。另一方面，MLE 对复杂模型往往难以处理，因此需要变分或 Monte Carlo 近似推理技术（如变分自编码器中使用的），或特殊模型结构（如标准化流和自回归模型），以获得可处理的似然，导致表达能力与计算代价之间的权衡困难。

最近的进展通过将传输方案表示为连续时间过程，实现了突破。例如，带神经常微分方程（ODE）的流模型 [e.g., 6, 56] 和通过随机微分方程（SDE）的扩散模型 [e.g., 73, 23, 80, 11, 82]。在这些模型中，神经网络学习表示过程的漂移力，数值 ODE/SDE 求解器在推理时模拟过程。关键思想是通过利用 ODE/SDE 的数学结构，连续时间模型可高效训练，无需采用极小极大或传统近似推理技术。最著名的例子是近期的基于分数的生成模型 [71–73] 和去噪扩散概率模型（DDPM）[23]，我们统称为去噪扩散方法。这些方法使我们能训练大规模扩散/SDE 基生成模型，在图像生成的质量和多样性上都超过 GAN，且无不稳定性和模式崩溃问题 [e.g., 12, 53, 61, 64]。学习到的 SDE 可通过概率流 ODE（PF-ODE）[73] 和 DDIM [70] 方法转换为确定性 ODE 模型，加快推理。

然而，相比 GAN 和 VAE 这样的传统单步模型，连续时间模型的一个关键缺点是推理时的计算代价高：生成单一样本（如图像）需要用数值求解器求解 ODE/SDE，该求解器需重复调用昂贵的神经漂移函数。此外，现有去噪扩散技术需在复杂设计空间进行大量超参搜索，且在实证和理论上理解有限 [29]。

在现有方法中，生成式建模和领域迁移通常单独处理。往往需要扩展或定制生成学习技术以解决领域迁移问题；例见 Cycle GAN [100] 和基扩散的图像到图像翻译 [e.g., 75, 97]。一个自然统一两个

领域的框架是最优传输（OT）[e.g., 85, 2, 15, 59]，它提供一套技术用于找最小传输代价的最优耦合，其形式为 $\mathbb{E}[c(Z_1 - Z_0)]$ w.r.t. 代价函数 $c: \mathbb{R}^d \rightarrow \mathbb{R}$ ，自然应用于生成和迁移学习。然而，现有 OT 技术对高维和大数据量问题求解缓慢 [59]。此外，由于传输代价与实际学习性能不完全对齐，忠实寻找最优传输映射的方法不一定有更好的学习性能 [34]。

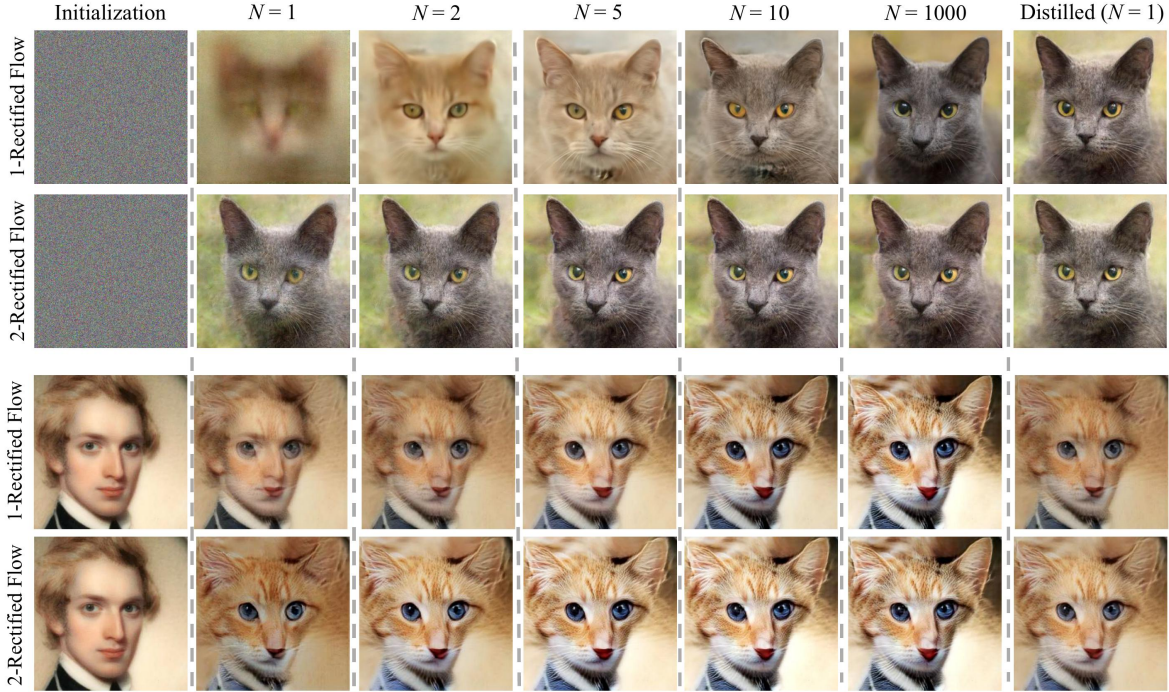


图 1: Rectified Flow 轨迹用于图像生成 (π_0 : 标准高斯噪声, π_1 : 猫脸图像, 上两行) 和人脸与猫脸间的图像迁移 (π_0 : 人脸, π_1 : 猫脸, 下两行), 采用欧拉方法模拟, 步长为 $1/N$, 共 N 步。从训练数据诱导的第一个 Rectified Flow (1-整流流) 即使只用少数几步 (例如 ≥ 2) 也能获得良好结果; 从 1-整流流诱导的平直化 Reflow (记为 2-整流流) 具有近似直线轨迹, 即使仅用一步离散化也能获得良好结果。

贡献

我们提出 *Rectified Flow*, 一个简明有效的传输映射问题求解方法, 统一解决生成式建模和领域迁移。Rectified Flow 是一个 ODE 模型, 通过 尽可能跟随直线路径 将分布 π_0 传输到 π_1 。直线路径之所以可取, 一方面理论上因为它是两端点间最短路径, 另一方面计算上因为它可不需时间离散化精确模拟。因此, 具有直线路径的流在单步和连续时间模型间架起桥梁。

从算法上, Rectified Flow 用简单可扩展的无约束最小二乘优化过程训练, 避免了 GAN 的不稳定性、MLE 方法的难处理似然以及去噪扩散模型的微妙超参数决策。从训练数据获得 Rectified Flow 的过程具有吸引人的理论性质: 1) 对所有凸代价函数 c 都产生非递增的传输代价, 2) 使流的路径越来越直, 降低数值求解器的误差。因此, 通过 *Reflow* 过程——迭代用先前得到的 Rectified Flow

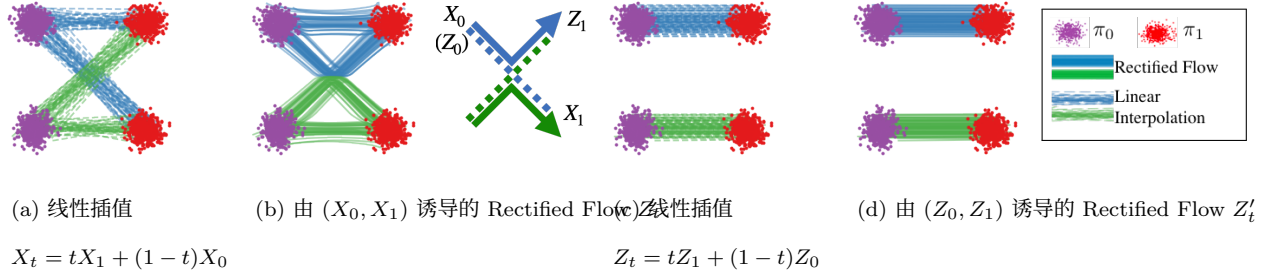


图 2: (a) 数据输入 $(X_0, X_1) \sim \pi_0 \times \pi_1$ 的线性插值。(b) 由 (X_0, X_1) 诱导的 Rectified Flow Z_t , 轨迹在交点处重新连接以避免交叉。(c) 流 Z_t 端点 (Z_0, Z_1) 的线性插值。(d) 由 (Z_0, Z_1) 诱导的 Rectified Flow, 沿直线路径。

模拟的数据训练新的 Rectified Flow——我们得到近似直线的流，即使只用一步欧拉离散化也能获得良好结果。我们的方法完全基于 ODE，在概念上比基于 SDE 的方法 [23, 73, 70] 更简洁，推理也更快。

实证上，用较少的欧拉步数（见图 1 上行），Rectified Flow 就能生成高质量图像。仅需一步 Reflow，流即变得近似直线，用单个欧拉步就能获得良好结果（图 1 第二行）。这大幅改进了标准去噪扩散方法。在单步快速扩散/流模型中，我们在 CIFAR10 上达到最先进的 FID (4.85) 和回召 (0.51) 指标 [5, 48, 91, 99, 47]。同一算法在领域迁移任务（如图像到图像翻译，见图 1 下两行）和迁移学习上也取得优异结果。

2 方法

本节首先在 2.1 节快速概览方法，然后在 2.2 节给出进一步讨论和说明。2.3 节介绍方法的非线性推广，澄清与概率流 ODE (Probability Flow ODE) [73] 和 DDIM [70] 方法的联系与优势。

2.1 方法概览

Rectified Flow (整流流) 给定 $X_0 \sim \pi_0, X_1 \sim \pi_1$ 的经验观测，由 (X_0, X_1) 诱导的 Rectified Flow 是时间 $t \in [0, 1]$ 上的常微分方程 (ODE) 模型：

$$dZ_t = v(Z_t, t)dt,$$

它将 Z_0 从 π_0 转换为服从 π_1 的 Z_1 。速度场 $v: \mathbb{R}^d \rightarrow \mathbb{R}^d$ 的设计目标是推动流尽可能沿着从 X_0 指向 X_1 的线性路径方向 $(X_1 - X_0)$ 运动。这通过求解一个简单的最小二乘回归问题实现：

$$\min_v \int_0^1 \mathbb{E} \left[\|(X_1 - X_0) - v(X_t, t)\|^2 \right] dt, \quad \text{其中} \quad X_t = tX_1 + (1-t)X_0, \quad (1)$$

其中 X_t 是 X_0 和 X_1 的线性插值。形式上, X_t 满足 $dX_t = (X_1 - X_0)dt$, 这是非因果的 (或称预见的), 因为 X_t 的更新需要知道终点 X_1 的信息。通过用 $X_1 - X_0$ 拟合速度场 v , Rectified Flow 将线性插值路径 X_t 转换为具有因果性的流, 得到可以无需观察未来而精确模拟的 ODE。

实践中, 我们用神经网络或其他非线性模型参数化 v , 并用随机梯度下降等现成优化器求解 (1) (利用对 (X_0, X_1) 的经验采样)。参见算法 1。得到 v 后, 我们从 $Z_0 \sim \pi_0$ 求解 ODE 将 π_0 转换为 π_1 , 或从 $Z_1 \sim \pi_1$ 反向求解将 π_1 转换为 π_0 。具体地, 对于反向采样, 我们从 $\tilde{X}_0 \sim \pi_1$ 求解 $d\tilde{X}_t = -v(\tilde{X}_t, t)dt$, 并设置 $X_t = \tilde{X}_{1-t}$ 。前向和反向采样在训练算法中同等重要, 因为 (1) 中的目标函数关于时间对称, 即交换 X_0 和 X_1 并翻转 v 的符号会得到等价问题。

流不相交 理解本方法的关键在于流的不相交性质: 遵循定义明确的 ODE $dZ_t = v(Z_t, t)dt$ 的不同路径 (该 ODE 有存在且唯一解) 在任何时刻 $t \in [0, 1)$ 都不能相交。具体地, 不存在位置 $z \in \mathbb{R}^d$ 和时刻 $t \in [0, 1)$ 使得两条路径沿不同方向在 z 处相交, 否则 ODE 的解不唯一。相反, 插值过程 X_t 的路径可能相互相交 (见图 2a), 这使其非因果。因此, 如图 2b 所示, Rectified Flow 在交点处重新连接各轨迹以避免相交, 同时由于 (1) 的优化, 保持与线性插值路径相同的密度分布。

我们可以将线性插值 X_t 看作建造连接 π_0 和 π_1 的道路 (或隧道), 将 Rectified Flow 看作粒子以短视、无记忆、不相交的方式通过这些道路, 从而能够忽略 X_0 和 X_1 配对的全局路径信息, 并重建 (Z_0, Z_1) 的更确定性配对。

Rectified Flow 降低传输代价 若 (1) 精确求解, 则 Rectified Flow 的配对 (Z_0, Z_1) 保证是 π_0, π_1 的有效耦合 (定理 3.3), 即当 $Z_0 \sim \pi_0$ 时 Z_1 服从 π_1 。此外, (Z_0, Z_1) 对所有凸代价函数 c 同时保证给出不大于数据配对 (X_0, X_1) 的传输代价 (定理 3.5)。数据配对 (X_0, X_1) 可以是 π_0, π_1 的任意耦合, 通常是独立的 (即 $(X_0, X_1) \sim \pi_0 \times \pi_1$), 这是由于实际问题中缺乏有意义的配对观测。相比之下, Rectified 耦合 (Z_0, Z_1) 具有确定性依赖, 因为它由 ODE 模型构造。记 $(Z_0, Z_1) = \text{Rectify}((X_0, X_1))$ 为从 (X_0, X_1) 到 (Z_0, Z_1) 的映射。因此, $\text{Rectify}(\cdot)$ 将任意耦合转化为传输代价更低的确定性耦合。

直线路径实现快速模拟 按照算法 1, 记 $Z = \text{RectFlow}((X_0, X_1))$ 为由 (X_0, X_1) 诱导的 Rectified Flow。递归应用该算子得到 Rectified Flow 序列 $Z^{k+1} = \text{RectFlow}((Z_0^k, Z_1^k))$, 其中 $(Z_0^0, Z_1^0) = (X_0, X_1)$, Z^k 称为第 k 个 Rectified Flow (简称 k -Rectified Flow)。

这一 Reflow (重整流) 过程不仅降低传输代价, 还具有平直化 Rectified Flow 路径的重要效应, 即使流路径逐步变得更平直。这在计算上极具吸引力, 因为近似平直的路径在数值模拟中导致时间离散化误差较小。实际上, 完全平直的路径可用单步欧拉法精确模拟, 实际上是单步模型, 这解决了现有连续时间 ODE/SDE 模型推理代价高昂的瓶颈问题。

算法 1 Rectified Flow: 主算法

过程: $Z = \text{RectFlow}((X_0, X_1))$:

输入: 来自 π_0 和 π_1 的耦合 (X_0, X_1) 的采样; 速度模型 $v_\theta: \mathbb{R}^d \rightarrow \mathbb{R}^d$, 参数为 θ 。

训练: $\hat{\theta} = \arg \min_{\theta} \mathbb{E} \left[\|X_1 - X_0 - v(tX_1 + (1-t)X_0, t)\|^2 \right]$, 其中 $t \sim \text{Uniform}([0, 1])$ 。

采样: 从 $Z_0 \sim \pi_0$ 开始 (或从 $Z_1 \sim \pi_1$ 反向) 按 $dZ_t = v_{\hat{\theta}}(Z_t, t)dt$ 采样 (Z_0, Z_1) 。

返回: $Z = \{Z_t: t \in [0, 1]\}$ 。

Reflow (可选): $Z^{k+1} = \text{RectFlow}((Z_0^k, Z_1^k))$, 从 $(Z_0^0, Z_1^0) = (X_0, X_1)$ 开始。

蒸馏 (可选): 学习神经网络 $\hat{T}$ 以蒸馏第 k 个 Rectified Flow, 使得 $Z_1^k \approx \hat{T}(Z_0^k)$ 。

2.2 主要结果与性质

我们对整流流的主要性质提供更深入的讨论。为突出本节直观的洞察, 我们保持讨论的非正式风格, 将完整的理论分析推迟到第 3 节。

首先, 对于给定的输入耦合 (X_0, X_1) , 容易看出 (1) 的精确最小值通过以下方式达到:

$$v^X(x, t) = \mathbb{E}[X_1 - X_0 \mid X_t = x], \quad (2)$$

即通过 x 在时刻 t 的线方向 $X_1 - X_0$ 的期望。以下讨论整流流 $dZ_t = v^X(Z_t, t)dt$ 的性质, 其中 $Z_0 \sim \pi_0$, 假设常微分方程有唯一解。

保边缘性 [定理 3.3] 对 (Z_0, Z_1) 是 π_0 和 π_1 的耦合。实际上, 在每个时刻 t , Z_t 的边缘分布等于 X_t 的边缘分布, 即 $\text{Law}(Z_t) = \text{Law}(X_t), \forall t \in [0, 1]$ 。

直观上, 这是因为根据 (2) 中 v^X 的定义, 在所有位置和时刻通过每个无穷小体积的质量流入和流出的期望在 X_t 和 Z_t 的动力学下相等, 确保它们保持相同的边缘分布:

$$\text{流入与流出} \left(\text{Diagram} \right) = \text{流入与流出} \left(\text{Diagram} \right), \forall \text{时刻与位置} \implies \text{Law}(Z_t) = \text{Law}(X_t), \forall t.$$

另一方面, 整个轨迹 Z_t 和 X_t 的联合分布通常不同。特别地, X_t 通常是非因果的、非马尔可夫过程, 具有随机耦合 (X_0, X_1) , 而 Z_t 在保持所有时刻边缘分布的同时, 对 X_t 进行了因果化、马尔可夫化和去随机化。

降低传输代价 [定理 3.5] 耦合 (Z_0, Z_1) 对任何凸代价函数 $c: \mathbb{R}^d \rightarrow \mathbb{R}$, 产生不超过输入 (X_0, X_1) 的凸传输代价, 即 $\mathbb{E}[c(Z_1 - Z_0)] \leq \mathbb{E}[c(X_1 - X_0)]$ 。

传输代价衡量根据耦合指定的赋值关系将一个分布的质量传输到另一个分布所需的代价, 是最优传输的中心课题 [例如 84, 85, 65, 59, 15]。常见的例子是 $c(\cdot) = \|\cdot\|^\alpha$, 其中 $\alpha \geq 1$ 。因此, $\text{Rectify}(\cdot)$

在所有凸传输代价的集合上产生帕累托下降，而不针对任何特定的 c 。这区别于典型的最优传输优化方法，后者明确针对给定的 c 进行优化。因此， $\text{Rectify}(\cdot)$ 的递归应用不能保证对任何给定的 c 都能达到 c -最优耦合，除了一维情况，此时 $\text{Rectify}(\cdot)$ 的不动点与同时最小化所有非负凸代价 c 的唯一单调耦合一致；见第 3.4 节。

直观上，凸传输代价保证下降是因为整流流 Z_t 的路径是连接 (X_0, X_1) 的直线路径的重新连接。为说明此点，考虑 $c(\cdot) = \|\cdot\|$ 的简单情况，此时传输代价 $\mathbb{E}[\|X_0 - X_1\|]$ 和 $\mathbb{E}[\|Z_0 - Z_1\|]$ 是连接端点的直线长度的期望。不等式可用如下图解证明：

$$\mathbb{E}[\|Z_0 - Z_1\|] = \text{长度} \left(\begin{array}{c} \text{---} \text{---} \text{---} \\ \text{---} \text{---} \end{array} \right) \stackrel{(*)}{\leq} \text{长度} \left(\begin{array}{c} \text{---} \text{---} \text{---} \\ \text{---} \text{---} \end{array} \right) \stackrel{(**)}{=} \text{长度} \left(\begin{array}{c} \text{---} \text{---} \text{---} \\ \text{---} \text{---} \end{array} \right) = \mathbb{E}[\|X_0 - X_1\|],$$

其中 $\stackrel{(*)}{\leq}$ 使用三角不等式， $\stackrel{(**)}{=}$ 成立是因为 Z_t 的路径是 X_t 直线路径的重新连接，遵循 (2) 中 v^X 的构造。对于一般的凸 c ，类似的证明使用詹森不等式在第 3.2 节中给出。

Reflow、平直化与快速模拟 如图 3 所示，当我们递归应用过程 $Z^{k+1} = \text{RectFlow}((Z_0^k, Z_1^k))$ 时，第 k 个整流流 Z^k 的路径随着 k 的增加而越来越笔直，因此在数值上更容易模拟。这种平直化趋势在理论上可以保证。

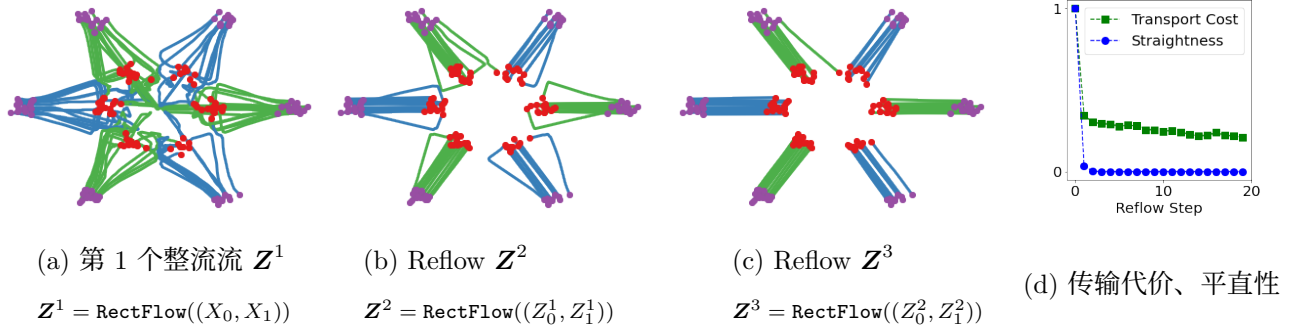


图 3: (a)-(c) 玩具例子中从 reflow 采样的轨迹样本 (π_0 : 紫色点, π_1 : 红色点; 绿色和蓝色线是连接 π_0, π_1 不同模式的轨迹)。 (d) reflow 步数与平直性和相对 L2 传输代价的关系; 值被缩放到 $[0, 1]$, 其中 0 对应直线和 L2 最优传输; 详见第 5.1 节。我们使用 (5) 中带宽 $h = 0.1$ 的非参数模型。

具体而言, 我们说流 $dZ_t = v(Z_t, t)dt$ 是直的, 如果对所有 $\forall t \in [0, 1]$ 几乎必然有 $Z_t = tZ_1 + (1-t)Z_0$, 或等价地 $v(Z_t, t) = Z_1 - Z_0 = \text{const}$ 沿每条路径成立。(更准确地说, 这里”直”是指以常速运动的直线。) 这样的直流在计算上极具吸引力, 因为它实际上是单步模型: 单个欧拉步更新 $Z_1 = Z_0 + v(Z_0, 0)$ 从 Z_0 精确计算出 Z_1 。注意线性插值 $X = \{X_t\}$ 按此定义是直的, 但它不是 (因果的) 流, 因此不能在没有任何预言机访问 π_0 和 π_1 的样本的情况下模拟。相比之下, 使流 $dZ_t = v(Z_t, t)dt$ 笔直是非平凡的, 因为如果是这样, v 必须满足无粘伯格方程 $\partial_t v + (\partial_z v)v = 0$:

$$\frac{d}{dt} v(Z_t, t) = \partial_z v(Z_t, t) \dot{Z}_t + \partial_t v(Z_t, t) = \partial_z v(Z_t, t) v(Z_t, t) + \partial_t v(Z_t, t) = 0.$$

更一般地，我们可以通过以下方式测量任何连续可微过程 $\mathbf{Z} = \{Z_t\}$ 的平直性：

$$S(\mathbf{Z}) = \int_0^1 \mathbb{E} \left[\left\| (Z_1 - Z_0) - \dot{Z}_t \right\|^2 \right] dt. \quad (3)$$

$S(\mathbf{Z}) = 0$ 表示精确的直。平直性较小的流具有近似笔直的路径，因此可以用较少的离散化步数用数值求解器精确模拟。第 3.3 节表明递归应用整流可以证明地将 $S(\mathbf{Z})$ 减少至零。

[定理 3.7] 设 $\mathbf{Z}^k$ 是从 (X_0, X_1) 导出的第 k 个整流流。则

$$\min_{k \in \{0 \dots K\}} S(\mathbf{Z}^k) \leq \frac{\mathbb{E}[\|X_1 - X_0\|^2]}{K}.$$

如图 1 所示，只需应用一步 reflow 就可以提供近似笔直的流，即使用单个欧拉步模拟时也能产生良好的性能。不建议应用太多 reflow 步数，因为这可能会积累 v^X 的估计误差。

蒸馏 获得第 k 个整流流 $\mathbf{Z}^k$ 后，我们可以通过将 (Z_0^k, Z_1^k) 的关系蒸馏到神经网络 $\hat{T}$ 中来进一步改进推理速度，以直接从 Z_0^k 预测 Z_1^k ，而无需模拟流。由于流已经近似笔直（因此由单步更新很好地近似），蒸馏可以高效完成。特别地，如果取 $\hat{T}(z_0) = z_0 + v(z_0, 0)$ ，则蒸馏 $\mathbf{Z}^k$ 的损失函数是 $\mathbb{E} \left[\left\| (Z_1^k - Z_0^k) - v(Z_0^k, 0) \right\|^2 \right]$ ，这正是 $t = 0$ 时 (1) 中的项。

我们应当强调蒸馏和整流之间的区别：蒸馏试图忠实地近似耦合 (Z_0^k, Z_1^k) ，而整流产生一个不同的耦合 (Z_0^{k+1}, Z_1^{k+1}) ，具有更低的传输代价和更笔直的流。因此，蒸馏只应在最后阶段应用，当我们想要微调模型以实现快速单步推理时。

关于速度场 v^X 如果 X_0 在条件 $X_1 = x_1$ 下给出条件密度函数 $\rho(x_0 | x_1)$ ，那么最优速度场 $v^X(z, t) = \mathbb{E}[X_1 - X_0 | X_t = z]$ 可以表示为：

$$v^X(z, t) = \mathbb{E} \left[\frac{X_1 - z}{1 - t} \eta_t(X_1, z) \right], \quad \eta_t(X_1, z) = \rho \left(\frac{z - tX_1}{1 - t} \mid X_1 \right) / \mathbb{E} \left[\rho \left(\frac{z - tX_1}{1 - t} \mid X_1 \right) \right], \quad (4)$$

其中期望 $\mathbb{E}[\cdot]$ 关于 $X_1 \sim \pi_1$ 求取。这可以通过注意到当条件 $X_t = z$ 成立时， $X_0 = \frac{z - tX_1}{1 - t}$ 和 $X_1 - X_0 = \frac{X_1 - z}{1 - t}$ 来看出。因此，如果 ρ 在任何地方都是正的且连续的，则 v^X 在 $\mathbb{R}^d \times [0, 1)$ 上是良好定义且连续的。进一步，如果 $\log \eta_t$ 关于 z 连续可微，我们可以证明：

$$\nabla_z v^X(z, t) = \frac{1}{1 - t} \mathbb{E} [((X_1 - z) \nabla_z \log \eta_t(X_1, z) - 1) \eta_t(X_1, z)].$$

注意如果 v^X 在 $[0, a]$ 上对任何 $a < 1$ 均匀李普希茨连续，则 $dZ_t = v^X(Z_t, t)dt$ 保证有唯一解。

如果 $X_0 | X_1 = x_1$ 不产生条件密度函数， $v^X(z, t)$ 可能未定义或不连续，使常微分方程 $dZ_t = v^X(Z_t, t)dt$ 行为不良。简单的修正是向 X_0 加入独立于 (X_0, X_1) 的高斯噪声 $\xi \sim \mathcal{N}(0, \sigma^2 I)$ 以得到平滑变量 $\tilde{X}_0 = X_0 + \xi$ ，然后使用整流流将 $\tilde{X}_0$ 传输到 X_1 。这实际上给出了形式为 $T(X_0 + \xi)$ 的随机映射，将 π_0 传输到 π_1 。

光滑函数逼近 按照 (4), 如果条件密度函数 $\rho(\cdot|x_1)$ 存在且已知, 且 π_1 是有限个点的经验测度 (其期望可以精确计算), 我们可以精确计算 v^X 。在此情况下, 向前运行整流流将精确恢复 π_1 中的点。然而, 在大多数情况下这在实践中并不有用, 因为它完全过拟合数据。因此, 用神经网络或非参数模型等光滑函数逼近器拟合 v^X 既是必要的也是有益的, 以获得包含实用新样本的光滑分布。

对于大规模问题, 深度神经网络无疑是最好的函数逼近器。对于低维问题, 以下简单的纳达亚-沃森风格的非参数估计器可以在不知道条件密度 ρ 的情况下对精确整流流产生良好的逼近:

$$v^{X,h}(z, t) = \mathbb{E} \left[\frac{X_1 - z}{1 - t} \omega_h(X_t, z) \right], \quad (5)$$

其中 $\omega_h(X_t, z) = \frac{\kappa_h(X_t, z)}{\mathbb{E}[\kappa_h(X_t, z)]}$, 而 $\kappa_h(x, z)$ 是带宽参数 $h > 0$ 的平滑核, 衡量 z 和 x 之间的相似性。取高斯 RBF 核 $\kappa_h(x, z) = \exp(-\|x - z\|^2 / 2h^2)$, 当 $h \rightarrow 0^+$ 时, 可以证明 $v^{X,h}(z, t)$ 在能被 X_t 达到的点 z 上收敛到 $v^X(z, t) = \mathbb{E} \left[\frac{X_1 - z}{1 - t} \mid X_t = z \right]$ (即条件期望 $\mathbb{E}[\cdot \mid X_t = z]$ 存在)。在 X_t 无法达到的点 z 上, $v^{X,h}(z, t)$ 通过找接近 z 的 X_t 进行外推。在实践中, 我们用经验平均替换 (5) 中的期望。我们发现 $v^{X,h}$ 在实践中表现良好, 因为它是总指向 π_1 支撑中某点的线性函数的混合。

2.3 非线性扩展

我们提出 Rectified Flow 的一种非线性扩展, 其中线性插值 X_t 被任意时间可微曲线取代, 该曲线连接 X_0 和 X_1 。这类广义整流流仍能将分布 π_0 传输至 π_1 (定理 3.3), 但不再保证凸传输代价的减少, 也不具备平直化效应。重要的是, 概率流方法 [73] 和 DDIM[70] 可 (大致) 视为该框架的特殊情形, 这使我们能够澄清与这些方法的联系及优势。

设 $\mathbf{X} = \{X_t : t \in [0, 1]\}$ 为任意时间可微随机过程, 连接 X_0 和 X_1 。设 $\dot{X}_t$ 为 X_t 关于时间的导数。由 $\mathbf{X}$ 诱导的 (非线性) 整流流定义为

$$dZ_t = v^{\mathbf{X}}(Z_t, t)dt, \quad \text{其中} \quad Z_0 = X_0, \quad \text{且} \quad v^{\mathbf{X}}(z, t) = \mathbb{E} \left[\dot{X}_t \mid X_t = z \right].$$

可通过求解如下问题估计 $v^{\mathbf{X}}$:

$$\min_v \int_0^1 \mathbb{E} \left[w_t \left\| v(X_t, t) - \dot{X}_t \right\|^2 \right] dt, \quad (6)$$

其中 $w_t: (0, 1) \rightarrow (0, +\infty)$ 是正加权序列 (默认 $w_t = 1$)。当使用线性插值 $X_t = tX_1 + (1 - t)X_0$ 时, 我们有 $\dot{X}_t = X_1 - X_0$, (6) ($w_t = 1$ 时) 退化为 (1)。

如定理 3.3 所示, 该方法得到的流 $\mathbf{Z}$ 仍保持 $\mathbf{X}$ 的边缘律, 即 $\text{Law}(Z_t) = \text{Law}(X_t)$ ($\forall t \in [0, 1]$), 因此 (Z_0, Z_1) 仍为 π_0, π_1 的耦合。但若 $\mathbf{X}$ 非直线, 则 (Z_0, Z_1) 不再保证相对于 (X_0, X_1) 凸传输代价的减少。更关键的是, Reflow 过程不再能平直 Z_t 的轨迹。

一类简单的插值过程为 $X_t = \alpha_t X_1 + \beta_t X_0$ ，其中 α_t 和 β_t 是两个可微序列，满足 $\alpha_1 = \beta_0 = 1$ 和 $\alpha_0 = \beta_1 = 0$ 以保证过程在起点和终点分别等于 X_0 和 X_1 。此时，在 (6) 中有 $\dot{X}_t = \dot{\alpha}_t X_1 + \dot{\beta}_t X_0$ ，其中 $\dot{\alpha}_t$ 和 $\dot{\beta}_t$ 分别为 α_t 和 β_t 的时间导数。曲线形状由 α_t 和 β_t 的关系控制。若取 $\beta_t = 1 - \alpha_t$ (对所有 t)，则 X_t 具有直线路径但不以恒定速度运动；它可视为标准情形 $X_t = tX_1 + (1-t)X_0$ 的时间重参数化变体 (当 t 重参数化为 α_t 时)。当 $\beta_t \neq 1 - \alpha_t$ 时， X_t 的路径除了某些特殊情形 (如 $\dot{\alpha}_t X_1 = 0$ 或 $\dot{\beta}_t X_0 = 0$ ，或 $X_1 = aX_0$ (某 $a \in \mathbb{R}$) 外，不是直线。

2.3.1 概率流 ODE 与 DDIM

概率流 ODE (PF-ODE) [73] 和去噪扩散隐模型 (DDIM) [70] 是从球形高斯初始分布 π_0 学习 ODE 型生成模型 π_1 的方法。它们通过将去噪扩散方法学习的 SDE 转换为具有等价边缘律的 ODE 而得到。在 [73] 中，从三类 SDE (学习为基于分数的生成模型) 推导出三类 PF-ODE，包括方差爆炸 (VE) SDE、方差保存 (VP) SDE 和 sub-VP SDE，我们分别记为 VE ODE、VP ODE 和 sub-VP ODE。VP ODE 对应 DDIM 的连续时间极限，后者由去噪扩散概率模型 (DDPM) [23] 推导而来。由于 PF-ODE 和 DDIM 的推导需要高等随机微积分工具，我们仅讨论 [73, 23] 中建议的最终算法过程，这将在第 3.5 节总结。读者可参考 [73, 70] 了解详情。

[命题 3.11] 所有 PF-ODE 变体在使用 $X_t = \alpha_t X_1 + \beta_t \xi$ (某些 α_t, β_t 满足 $\alpha_1 = 1, \beta_1 = 0$ ，其中 $\xi \sim \mathcal{N}(0, I)$ 为标准高斯随机变量) 时，都可视为 (6) 的特例。

我们需要用 ξ 代替 X_0 是因为 [73, 70, 23] 中建议的 α_t 和 β_t 选择不满足 $t = 0$ 时的边界条件 $\alpha_0 = 0$ 和 $\beta_0 = 1$ ，因此 $X_0 \neq \xi$ 。相反，在这些方法中，初始分布 $X_0 \sim \pi_0$ 隐式定义为 $X_0 = \alpha_0 X_1 + \beta_0 \xi$ ，通过令 $\alpha_0 X_1 \ll \beta_0 \xi$ 而近似为 $X_0 \approx \beta_0 \xi$ 。因此在这些方法中， π_0 被设定为 $\mathcal{N}(0, \beta_0^2 I)$ 。从我们的框架看，没有理由将 ξ 限制为 $\mathcal{N}(0, \beta_0^2 I)$ ，也没有必要设定 $\alpha_0 = 0, \beta_0 = 1$ 来避免近似。

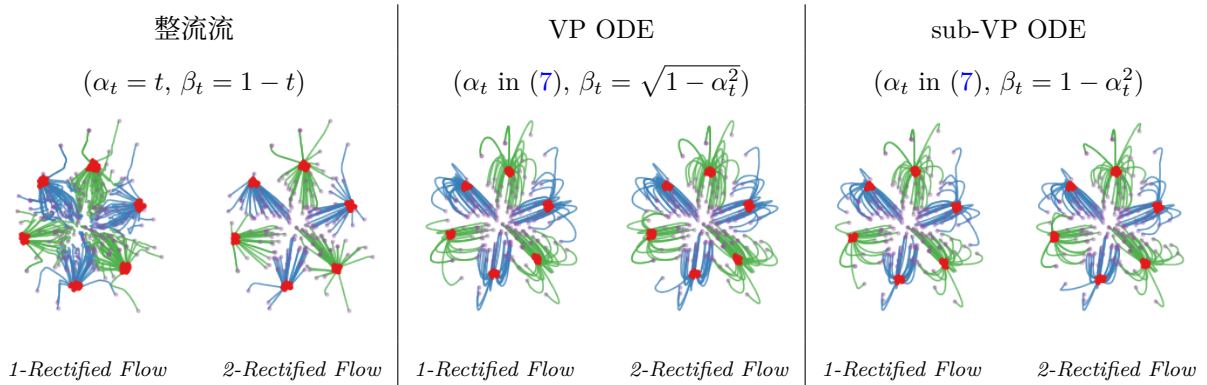


图 4: 整流流与 VP ODE、sub-VP ODE 在 $\pi_0 = \mathcal{N}(0, I)$ (紫色点) 和 π_1 为低方差高斯混合 (红色点) 时的比较。线性整流流经过一步 Reflow 后产生近似直线轨迹。但 VP ODE 和 sub-VP ODE 的轨迹呈曲线，无法通过 Reflow 平直。

时间离散化 步数 N	整流流 $\alpha_t = t, \beta_t = 1 - t$	VP ODE α_t in (7), $\beta_t = \sqrt{1 - \alpha_t^2}$	sub-VP ODE α_t in (7), $\beta_t = 1 - \alpha_t^2$	VP ODE (恒定速度) $\alpha_t = t, \beta_t = \sqrt{1 - \alpha_t^2}$
$N = 1$				
$N = 2$				
$N = 5$				
$N = 100$				

图 5: 不同方法在离散化步数 N 变化时的轨迹 (紫色点: π_0 ; 红色点: π_1 ; 橙色点: 中间步; 蓝色曲线: 流轨迹)。整流流沿直线运动且在时间上均匀推进; 当用单个欧拉步模拟时生成 π_1 的均值, 更多步数下迅速覆盖整个分布 π_1 (此例 $N = 2$ 足矣)。相比之下, VP ODE 和 sub-VP ODE 以非均匀速度沿曲线运动: 它们在初始阶段缓慢, 后期加速 (大部分更新发生在 $t \gtrsim 0.5$ 时)。设定 $\alpha_t = t$ 可避免非均匀速度 (见最后一列)。

VP ODE 和 sub-VP ODE [73] 中的 VP ODE 和 sub-VP ODE 使用如下共同的 α_t :

$$\text{(sub-)VP ODE: } \alpha_t = \exp\left(-\frac{1}{4}a(1-t)^2 - \frac{1}{2}b(1-t)\right); \quad \text{默认值: } a = 19.9, b = 0.1, \quad (7)$$

其中 a, b 的默认值选择以匹配 DDIM 和 DDPM 共同训练过程的连续时间极限。VP ODE 和 sub-VP ODE 的区别在于 β_t 的选择, 如下所示:

$$\text{VP ODE: } \beta_t = \sqrt{1 - \alpha_t^2}, \quad \text{sub-VP ODE: } \beta_t = 1 - \alpha_t^2. \quad (8)$$

由于在 VP ODE 和 sub-VP ODE 中 $\beta_0 \approx 1$, 两种情况下的 π_0 都取为 $\mathcal{N}(0, I)$ 。

上述 α_t, β_t 的选择是 [73] 中基于 SDE 的推导结果。但在检查诱导 ODE 的路径性质时, 它们的动机不充分:

- 非直线路径: 由于 (8) 中 β_t 的选择, VP ODE 和 sub-VP ODE 的轨迹一般是曲线, 无法通过 Reflow 过程平直。应选择 $\beta_t = 1 - \alpha_t$ 以诱导直线路径。

- 非均匀速度: (7) 中 α_t 的指数形式是在推导 SDE 模型时使用 Ornstein-Uhlenbeck 过程的结果 [73, 23]。但对于 ODE, 使用 (7) 没有明显优势。如图 5 所示, VP ODE 和 sub-VP ODE 的 α_t 和 β_t 在早期阶段 ($t \lesssim 0.5$) 变化缓慢。因此, 流在初期运动缓慢, 大部分更新集中在后期。这种非均匀更新速度加上非直线路径, 使 VP ODE 和 sub-VP

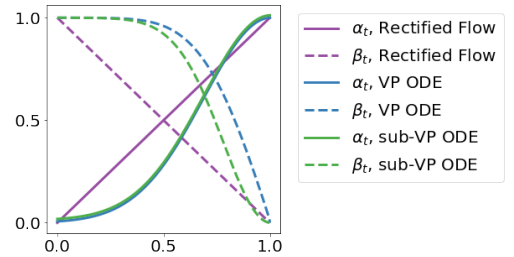


图 6: 不同方法的 t vs. α_t, β_t

ODE 在使用大步长时性能不佳，即使是在简单球形高斯分布间的传输（见图 5）。如图 5 最后一列所示，将 VP ODE 中的指数 α_t 改为线性函数 $\alpha_t = t$ 可获得均匀更新速度，同时保持相同的连续时间轨迹。

VE ODE [73] 中的 VE ODE 使用 $\alpha_t = 1$ 和 $\beta_t = \sigma_{\min} \sqrt{r^{2(1-t)} - 1}$ ，其中 $\sigma_{\min} = 0.01$ 是默认值， r 设定使得 $\sigma_{\max} := r\sigma_{\min}$ 等于 π_1 中所有训练数据点对间的最大欧氏距离（[72] 技巧 1）。假设 $\sigma_{\max}^2$ 远大于 $\sigma_{\min}^2$ 和 X_1 的方差，则 $X_0 = X_1 + \beta_0 \xi \approx \sigma_{\max} \xi$ ，我们可将初始分布设定为 $\pi_0 \sim \mathcal{N}(0, \sigma_{\max}^2 I)$ ，其方差远大于 π_1 。因此 VE ODE 无法应用于（也未显示在）图 4 和图 5 中的玩具示例。与 (sub-)VP ODE 情况类似，对 ξ 的限制实际上多余， $\sigma_{\max}$ 的设定从我们框架的角度看并无必要。另一方面，VE ODE 中 X_t 的轨迹确实为直线，因为 $\dot{X}_t = \dot{\beta}_t \xi$ 的方向始终与 ξ 相同。但 β_t 的选择导致类似于 (sub-)VP ODE 的非均匀速度问题。

继 [73, 23] 之后，已有多项工作试图改进 α_t, β_t 的选择，但仍受制于 SDE-to-ODE 推导的基本设计空间；见例如 [54, 29, 95]。

总结一下，(6) 中的简单非线性整流流框架既简化又扩展了现有框架，揭示了多个重要观点：

- 学习 ODE 可直接独立进行，无需诉诸扩散/SDE 方法；
- 学习 ODE 的路径可由 X_0 和 X_1 的任意光滑插值曲线 X_t 指定；
- 初始分布 π_0 可任意选择，独立于插值 X_t 的选择。
- 标准线性插值 $X_t = tX_1 + (1-t)X_0$ 应作为默认推荐选择。

另一方面， X_t 的非线性选择可在如下情形有用：当我们想融入变量的某些非欧几里得几何结构，或想对 ODE 的轨迹施加某些约束时。我们将此留作未来工作。

3 理论分析

我们为整流流提供理论分析。结果总结如下。

- [第 3.1 节] 所有具有任意插值 X_t 的非线性整流流都保持边缘分布。
- [第 3.2 节] 整流流（使用标准线性插值）降低凸传输代价。
- [第 3.3 节] Reflow（重整流）保证平直化（线性）整流流。
- [第 3.4 节] 我们澄清平直耦合与 c -最优耦合之间的关系。
- [第 3.5 节] 我们将 PF-ODE 建立为非线性整流流的实例。

3.1 保边缘性

边缘分布保持性质 $\text{Law}(Z_t) = \text{Law}(X_t)$ 对所有 t 成立，是通式 (6) 中所有非线性整流流的一般性质，无论插值 X_t 是否平直。

定义 3.1. 对于路径上处处连续可微的随机过程 $\mathbf{X} = \{X_t: t \in [0, 1]\}$ ，其期望速度 $v^{\mathbf{X}}$ 定义为

$$v^{\mathbf{X}}(x, t) = \mathbb{E}[\dot{X}_t \mid X_t = x], \quad \forall x \in \text{supp}(X_t).$$

对于 $x \notin \text{supp}(X_t)$ ，条件期望未定义，我们任意设定 $v^{\mathbf{X}}$ ，比如 $v^{\mathbf{X}}(x, t) = 0$ 。

定义 3.2. 如果 $v^{\mathbf{X}}$ 局部有界且下列积分方程的解存在唯一，则称 $\mathbf{X}$ 可整流：

$$Z_t = Z_0 + \int_0^t v^{\mathbf{X}}(Z_t, t) dt, \quad \forall t \in [0, 1], \quad Z_0 = X_0. \quad (9)$$

此时， $\mathbf{Z} = \{Z_t: t \in [0, 1]\}$ 称为由 $\mathbf{X}$ 诱导的整流流。

定理 3.3. 假设 $\mathbf{X}$ 可整流， $\mathbf{Z}$ 是其整流流。则对所有 $t \in [0, 1]$ 有 $\text{Law}(Z_t) = \text{Law}(X_t)$ 。

Proof. 对任意紧支撑的连续可微测试函数 $h: \mathbb{R}^d \rightarrow \mathbb{R}$ ，有

$$\frac{d}{dt} \mathbb{E}[h(X_t)] = \mathbb{E}[\nabla h(X_t)^\top \dot{X}_t] = \mathbb{E}[\nabla h(X_t)^\top v^{\mathbf{X}}(X_t, t)], \quad (10)$$

其中我们用了 $v^{\mathbf{X}}(X_t, t) = \mathbb{E}[\dot{X}_t \mid X_t]$ 。按定义，这等价于 $\pi_t := \text{Law}(X_t)$ 以分布意义下满足具有漂移 $v_t^{\mathbf{X}} := v^{\mathbf{X}}(\cdot, t)$ 的连续性方程：

$$\dot{\pi}_t + \nabla \cdot (v_t^{\mathbf{X}} \pi_t) = 0. \quad (11)$$

为了看到 (10) 和 (11) 的等价性，我们可以用 h 乘以 (11) 两边后积分：

$$0 = \int h(\dot{\pi}_t + \nabla \cdot (v_t^{\mathbf{X}} \pi_t)) = \int h \dot{\pi}_t - \nabla h^\top v_t^{\mathbf{X}} \pi_t = \frac{d}{dt} \mathbb{E}[h(X_t)] - \mathbb{E}[\nabla h(X_t)^\top v^{\mathbf{X}}(X_t, t)],$$

其中我们使用分部积分 $\int h \nabla \cdot (v_t^{\mathbf{X}} \pi_t) = - \int \nabla h^\top (v_t^{\mathbf{X}} \pi_t)$ 。

由于 Z_t 由相同的速度场 $v^{\mathbf{X}}$ 驱动，其边缘分布 $\text{Law}(Z_t)$ 满足同样的方程和同样的初值条件 ($Z_0 = X_0$)。因此， $\text{Law}(Z_t)$ 和 $\text{Law}(X_t)$ 的等价性成立，只要 (11) 的解唯一，这根据 Kurtz [37] 的推论 1.3（参见 Ambrosio and Crippa [1] 定理 4.1）等价于 $dZ_t = v^{\mathbf{X}}(Z_t, t)$ 的解的唯一性。□

3.2 降低凸传输代价

(Z_0, Z_1) 的凸传输代价不大于 (X_0, X_1) 的这一事实是由特殊线性插值 $X_t = tX_1 + (1-t)X_0$ （欧氏空间的测地线）的结果。

定义 3.4. 如果耦合 (X_0, X_1) 的线性插值过程 $\mathbf{X} = \{tX_1 + (1-t)X_0 : t \in [0, 1]\}$ 可整流, 则称该耦合可整流。此时, (9) 中的 $\mathbf{Z} = \{Z_t : t \in [0, 1]\}$ 称为耦合 (X_0, X_1) 的整流流, 记为 $\mathbf{Z} = \text{RectFlow}((X_0, X_1))$; (Z_0, Z_1) 称为 (X_0, X_1) 的整流耦合, 记为 $(Z_0, Z_1) = \text{Rectify}((X_0, X_1))$ 。

定理 3.5. 假设 (X_0, X_1) 可整流, $(Z_0, Z_1) = \text{Rectify}((X_0, X_1))$ 。则对任意凸函数 $c: \mathbb{R}^d \rightarrow \mathbb{R}$, 有

$$\mathbb{E}[c(Z_1 - Z_0)] \leq \mathbb{E}[c(X_1 - X_0)].$$

Proof. 证明基于 Jensen 不等式的基本应用。

$$\begin{aligned} \mathbb{E}[c(Z_1 - Z_0)] &= \mathbb{E}\left[c\left(\int_0^1 v^{\mathbf{X}}(Z_t, t) dt\right)\right] && // \text{因为 } dZ_t = v^{\mathbf{X}}(Z_t, t)dt \\ &\leq \mathbb{E}\left[\int_0^1 c(v^{\mathbf{X}}(Z_t, t)) dt\right] && // c \text{ 的凸性, Jensen 不等式} \\ &= \mathbb{E}\left[\int_0^1 c(v^{\mathbf{X}}(X_t, t)) dt\right] && // X_t \text{ 和 } Z_t \text{ 有相同的边缘分布} \\ &= \mathbb{E}\left[\int_0^1 c(\mathbb{E}[(X_1 - X_0) | X_t]) dt\right] && // v^{\mathbf{X}} \text{ 的定义} \\ &\leq \mathbb{E}\left[\int_0^1 \mathbb{E}[c(X_1 - X_0) | X_t] dt\right] && // c \text{ 的凸性, Jensen 不等式} \\ &= \int_0^1 \mathbb{E}[c(X_1 - X_0)] dt && // \mathbb{E}[\mathbb{E}[(X_1 - X_0) | X_t]] = \mathbb{E}[(X_1 - X_0)] \\ &= \mathbb{E}[c(X_1 - X_0)]. \end{aligned}$$

□

若 X_t 平直但速率非常数 (即 $X_t = \alpha_t X_1 + \beta_t X_0$, 其中 $\beta_t = 1 - \alpha_t$ 且 $\dot{\alpha}_t \geq 0$), 则当 c 是凸且 m -齐次的 (即 $c(ax) = |a|^m c(x)$ 对所有 $a \in \mathbb{R}, x \in \mathbb{R}^d$ 和某个常数 $m \in (0, 1]$), 仍有 $\mathbb{E}[c(Z_1 - Z_0)] \leq \mathbb{E}[c(X_1 - X_0)]$ 。

3.3 平直化效应

一个耦合 (X_0, X_1) 称为平直 (或完全整流), 如果它是 $\text{Rectify}(\cdot)$ 映射的不动点。获得平直耦合是可取的, 因为其整流流是平直的, 因此可以用数值求解器在一步内精确模拟。在本节中, 我们首先描述平直耦合的基本性质, 说明耦合是平直的当且仅当其线性插值路径不相交。然后, 我们证明递归整流使耦合及相关流以 $O(1/k)$ 的速率平直, 其中 k 是整流步数。

定理 3.6. 假设 (X_0, X_1) 可整流。令 $X_t = tX_1 + (1-t)X_0$, $\mathbf{Z} = \text{RectFlow}((X_0, X_1))$ 。则 (X_0, X_1) 是平直耦合当且仅当以下等价陈述成立。

1. 存在严格凸函数 $c: \mathbb{R}^d \rightarrow \mathbb{R}$, 使得 $\mathbb{E}[c(Z_1 - Z_0)] = \mathbb{E}[c(X_1 - X_0)]$ 。

2. (X_0, X_1) 是 $\text{Rectify}(\cdot)$ 的不动点，即 $(X_0, X_1) = (Z_0, Z_1)$ 。

3. 整流流与线性插值过程重合： $\mathbf{X} = \mathbf{Z}$ 。

4. 线性插值 $\mathbf{X}$ 的路径不相交：

$$V((X_0, X_1)) := \int_0^1 \mathbb{E} \left[\|X_1 - X_0 - \mathbb{E}[X_1 - X_0 \mid X_t]\|^2 \right] dt = 0, \quad (12)$$

其中 $V((X_0, X_1)) = 0$ 表示当 $t \sim \text{Uniform}([0, 1])$ 时， $X_1 - X_0 = \mathbb{E}[X_1 - X_0 \mid X_t]$ 几乎处处成立，这意味着经过每个 X_t 的直线唯一，因此没有线性插值路径相交。

Proof. $3 \rightarrow 2 \rightarrow 1$ ：显然。

$1 \rightarrow 4$ ：若 $\mathbb{E}[c(Z_1 - Z_0)] = \mathbb{E}[c(X_1 - X_0)]$ ，定理 3.5 证明中的两个 Jensen 不等式应用都是紧的。因 c 严格凸，第二个 Jensen 不等式的等号条件表明 $X_1 - X_0 = \mathbb{E}[X_1 - X_0 \mid X_t]$ 对 X 和 $t \sim \text{Uniform}([0, 1])$ 几乎处处成立，从而 $V(\mathbf{X}) = 0$ 。

$4 \rightarrow 3$ ：若 $V(\mathbf{X}) = 0$ ，对 $s \in (0, 1]$ 有 $\int_0^s (X_1 - X_0) dt = \int_0^s \mathbb{E}[X_1 - X_0 \mid X_t] dt = \int_0^s v^X(X_t, t) dt$ 。因此

$$X_t = X_0 + \int_0^t (X_1 - X_0) dt = X_0 + \int_0^t v^X(X_t, t) dt.$$

因为 $\mathbf{Z}$ 满足相同的方程 (9)，由解的唯一性有 $\mathbf{X} = \mathbf{Z}$ 。

□

$O(1/K)$ 收敛速率 现在我们展示，当递归应用整流时，整流流变得越来越平直，耦合的线性插值变得越来越不相交。

定理 3.7. 令 $\mathbf{Z}^k$ 为 (X_0, X_1) 的第 k 整流流，即 $\mathbf{Z}^{k+1} = \text{RectFlow}((Z_0^k, Z_1^k))$ 且 $(Z_0^0, Z_1^0) = (X_0, X_1)$ 。假设对 $k = 0, \dots, K$ ，每个 (Z_0^k, Z_1^k) 都可整流。

则

$$\sum_{k=0}^K S(\mathbf{Z}^{k+1}) + V((Z_0^k, Z_1^k)) \leq \mathbb{E} \left[\|X_1 - X_0\|^2 \right].$$

因此，若 $\mathbb{E}[\|X_1 - X_0\|^2] < +\infty$ ，则 $\min_{k \leq K} (S(\mathbf{Z}^k) + V((Z_0^k, Z_1^k))) = O(1/K)$ 。

Proof. 在定理 3.5 的证明中取 $c(x) = \|x\|^2$ ，我们可以得到

$$\mathbb{E}[\|X_1 - X_0\|] - \mathbb{E}[\|Z_1 - Z_0\|] = S(\mathbf{Z}) + V((X_0, X_1)). \quad (13)$$

对每个整流步应用它得到

$$\mathbb{E} \left[\left\| Z_1^k - Z_0^k \right\|^2 \right] - \mathbb{E} \left[\left\| Z_1^{k+1} - Z_0^{k+1} \right\|^2 \right] = S(Z^{k+1}) + V((Z_0^k, Z_1^k)).$$

对 $k = 0, \dots, K$ 做 telescoping 求和得到结果。

□

3.4 平直耦合与最优耦合

耦合 (X_0, X_1) 称为 c -最优，如果它在所有具有相同边缘分布的耦合中实现 $\mathbb{E}[c(X_1 - X_0)]$ 的最小值。理解和计算最优耦合一直是最优传输的主要焦点 [例如 84, 2, 15, 59]。平直耦合是一个不同的可取性质。在下面，我们展示平直性是严格凸函数 c 的 c -最优性的必要但非充分条件，除了一维情况，在一维情况下两个概念重合。因此，找到平直耦合”比较容易”，而找 c -最优耦合更困难。

定理 3.8. 若可整流的耦合 (X_0, X_1) 对某个严格凸的代价函数 c 是 c -最优的，则 (X_0, X_1) 是平直耦合。

Proof. 令 $(Z_0, Z_1) = \text{Rectify}((X_0, X_1))$ 。若 (X_0, X_1) 是 c -最优的，则必有 $\mathbb{E}[c(Z_1 - Z_0)] = \mathbb{E}[c(X_1 - X_0)]$ 。这对应于定理 3.6 第 1 条，故 (X_0, X_1) 平直。 □

一维情况 对于 $\mathbb{R}$ 上的任意 π_0, π_1 ，存在唯一的耦合 (X_0^*, X_1^*) 对所有非负凸代价函数 c 同时最优。这个耦合通过单调性性质唯一刻画：对于支撑中的任意 (x_0, x_1) 和 (x'_0, x'_1) ，若 $x_0 < x'_0$ ，则 $x_1 \leq x'_1$ 。进一步，若 π_0 对 Lebesgue 测度绝对连续，则 (X_0^*, X_1^*) 必须是确定性的，即存在映射 $T: \mathbb{R} \rightarrow \mathbb{R}$ 使得 $X_1^* = T(X_0^*)$ 。参见 [65]。

在下面，我们展示 $\mathbb{R}$ 上的平直耦合与确定性单调耦合 (X_0^*, X_1^*) 重合，因此当 π_0 绝对连续时，平直耦合唯一且对所有凸 c 同时最优。在 $\mathbb{R}$ 上，耦合单调当且仅当其线性插值路径不相交——这正是平直耦合的特征。

引理 3.9. $\mathbb{R}$ 上的耦合是平直的当且仅当它是确定性且单调的。

定理 3.10. 对于 $\mathbb{R}$ 上的任意 π_0, π_1 ，存在要么不存在平直耦合，要么恰好一个平直耦合。进一步，若存在，唯一的平直耦合是确定性且单调的，且关于所有凸代价函数 $c: \mathbb{R}^d \rightarrow [0, +\infty)$ (其中 $\mathbb{E}[c(X_1 - X_0)]$ 的最小值存在且有限) 是联合最优的。

引理 3.9 的证明. 若 $\mathbb{R}$ 上的 (X_0, X_1) 是平直的，则它与其整流耦合 $(Z_0, Z_1) = \text{Rectify}((X_0, X_1))$ 重合。但 (Z_0, Z_1) 由整流流 $dZ_t = v^X(Z_t, t)dt$ 诱导，故显然是确定性的。由流的非穿交性也是单调的。具体地，若 (Z_0, Z_1) 不单调，则支撑中存在 (z_0, z_1) 和 (z'_0, z'_1) 满足 $z_0 < z'_0$ 但 $z_1 > z'_1$ 。则必有某个 $t_0 \in (0, 1)$ 使得 $z_{t_0} = z'_{t_0}$ 。但由解的唯一性，对 $t \geq t_0$ 有 $z_t = z'_t$ ，矛盾。

假设 (X_0, X_1) 是确定性且单调的。由单调性，不存在 x_0, x'_0 在 π_0 的支撑中满足 $x_0 \neq x'_0$ 且某个 $t_0 < 1$ 处有 $x_{t_0} = x'_{t_0}$ 。从而 $X_1 - X_0 = \mathbb{E}[X_1 - X_0 \mid X_t] = v^X(X_t, t)$ 对 $t \in (0, 1)$ 成立，即 $dX_t = (X_1 - X_0)dt = v^X(X_t, t)dt$ 是整流流的 ODE。显然 X_t 是该 ODE 的唯一解，根据定理 3.6 第 3 条， (X_0, X_1) 可整流且平直。 $\square$

定理 3.10 的证明。这是引理 3.9 和单调耦合是唯一的且对所有凸 c （其最优耦合存在）联合最优这一事实相结合的结果，根据 [65] 的引理 2.8 和定理 2.9。 $\square$

多维情况 另一方面，在 $\mathbb{R}^d$ ($d \geq 2$) 上，不同的代价函数 c 一般不共享一个公共的最优耦合，平直耦合不保证优化特定的 c ；这是预期的，因为 `Rectify(·)` 程序不依赖于 c 的特定选择。因此，必须修改 `Rectify(·)` 程序以使其适应感兴趣的特定 c 。

在最近的工作 [30] 中，有人推测 VP ODE（等价地 DDIM）诱导的耦合 (Z_0, Z_1) 关于二次损失产生最优耦合，但这被 [39, 78] 证明为假。在这里，我们展示即使平直耦合也不保证最优，更不用说 VP ODE 按设计不沿着平直路径。

我们在另一项专门致力于修改整流流以找 c -最优耦合的工作 [42] 中探索这一点；来自 [42] 可以容易陈述的一个结果是，关于二次代价 $c(\cdot) = \|\cdot\|^2$ 的最优耦合可以作为 `Rectify(·)` 的不动点达到，若 v 被限制为梯度场形式 $v(x, t) = \nabla f(x, t)$ 。将 v 限制为梯度场移除了速度场 v^X 导致次优传输代价的旋转分量。

3.5 去噪扩散模型与概率流 ODE

我们证明 [73] 中的概率流 ODE (PF-ODE) 可视为 (6) 中的非线性 Rectified Flow，其中 $X_t = \alpha_t X_1 + \beta_t \xi$ 。我们首先介绍去噪扩散模型与 PF-ODE 的算法流程，详细的理论推导见原始文献 [73, 23, 70]。

去噪扩散方法通过构造由标准布朗运动 W_t 驱动的 SDE 模型来学习生成模型：

$$dU_t = b(U_t, t)dt + \sigma_t dW_t, \quad U_0 \sim \pi_0, \quad (14)$$

其中 $\sigma_t: [0, 1] \rightarrow [0, +\infty)$ 是（通常为固定的）扩散系数， b 是可训练的神经网络，初始分布 π_0 受限于由算法超参数决定的球形高斯分布。该方法的思想是：首先用扩散过程（主要是 Ornstein-Uhlenbeck (OU) 过程）将数据坍缩至（近似）高斯分布，然后将生成扩散过程 (14) 理解为坍缩过程的时间反演 [e.g., 3]。

不深入推导细节, [73] 中 VE、VP、sub-VP SDE 对 b 的训练损失可总结为:

$$\min_v \int_0^1 \mathbb{E} \left[w_t \|v(V_t, t) - Y_t\|_2^2 \right] dt, \quad V_t = \alpha_t X_1 + \beta_t \xi_t, \quad Y_t = -\eta_t V_t - \frac{\sigma_t^2}{\beta_t} \xi_t, \quad (15)$$

其中 ξ_t 是满足 $\xi_t \sim \mathcal{N}(0, I)$ 的扩散过程, η_t, σ_t 是算法的超参数序列, α_t, β_t 由 η_t, σ_t 通过以下关系确定:

$$\alpha_t = \exp \left(\int_t^1 \eta_s ds \right), \quad \beta_t^2 = \int_t^1 \exp \left(2 \int_t^s \eta_r dr \right) \sigma_s^2 ds. \quad (16)$$

(16) 中的关系经推导得出, 使得 $\tilde{V}_t = V_{1-t} = \alpha_{1-t} X_1 + \beta_{1-t} \xi_t$ 遵循 Ornstein-Uhlenbeck (OU) 过程 $d\tilde{V}_t = \eta_{1-t} \tilde{V}_t dt + \sigma_{1-t} dW_t$ 。

VE SDE 等价于 [71, 72] 中的 SMLD, 取 $\eta_t = 0$ 因而 $\alpha_t = 1$ 。(sub-)VP SDE 取 η_s 为 s 的线性函数, 从而得到 (7) 中的指数 α_t 。VP SDE (等价于 DDPM [23]) 取 $\eta_t = -\frac{1}{2}\sigma_t^2$, 此时满足 $\alpha_t^2 + \beta_t^2 = 1$ (见 (8))。在 DDPM 中, 常采用 $b(x, t) = -\eta_t x - \frac{\sigma_t^2}{\beta_t} \epsilon(x, t)$ 的形式, 其中 ϵ 是神经网络, 从 (V_t, t) 预测 ξ_t 。

从理论上讲, (14) 中求解 (15) 的 SDE 在从 $U_0 = \alpha_0 X_1 + \beta_0 \xi_0$ 初始化时, 保证得到 $\text{Law}(U_1) = \text{Law}(X_1) = \pi_1$; 当 $\alpha_0 X_1 \ll \beta_0 \xi_0$ 时, 可近似为 $U_0 \approx \beta_0 \xi_0$ 。

利用 Fokker-Planck 方程的性质, [73, 70] 观察到: (14) 中用 (15) 训练的 SDE 可转化为保持相同边缘分布的 ODE:

$$dZ_t = \tilde{b}(Z_t, t) dt, \quad \text{with} \quad \tilde{b}(z, t) = \frac{1}{2}(b(z, t) - \eta_t z), \quad \text{starting from } Z_0 = U_0 = \alpha_0 X_1 + \beta_0 \xi_0. \quad (17)$$

等价地, 可将 $\tilde{b}$ 视为以下优化问题的解:

$$\min_v \int_0^1 \mathbb{E} \left[w_t \|v(V_t, t) - \tilde{Y}_t\|_2^2 \right] dt, \quad V_t = \alpha_t X_1 + \beta_t \xi_t, \quad \tilde{Y}_t = -\eta_t V_t - \frac{\sigma_t^2}{2\beta_t} \xi_t, \quad (18)$$

这与 (14) 仅在 Y_t 第二项相差 $1/2$ 的系数。此等价性仅在 (14) 和 (17) 采用特定初始化 $Z_0 = U_0 = \alpha_0 X_1 + \beta_0 \xi_0$ 时成立。

下面证明 (18) 可视为非线性 Rectified Flow 目标 (6), 其中 $X_t = \alpha_t X_1 + \beta_t \xi$, $\xi \sim \mathcal{N}(0, I)$ 。主要工作是利用 (16) 中的关系消去 η_t 和 σ_t , 证明 $\tilde{Y}_t$ 等价于 $\dot{X}_t$ 。

命题 3.11. 假设 (16) 成立。则 (18) 等价于 (6), 其中 $X_t = \alpha_t X_1 + \beta_t \xi$ 。

Proof. 首先, 注意对所有时刻 t 可取 $\xi_t = \xi$, 因为 ξ_t 的相关结构不影响结果。因此 $V_t = X_t =$

$\alpha_t X_1 + \beta_t \xi$ 。为证明 (18) 与 (6) 的等价性，只需验证 $\dot{X}_t = \tilde{Y}_t$ ：

$$\begin{aligned}\tilde{Y}_t &= -\eta_t X_t + \frac{\sigma_t^2}{2\beta_t^2}(\alpha_t X_1 - X_t) \\ &= -\dot{\eta}_t (\alpha_t X_1 + \beta_t \xi) + \frac{\sigma_t^2}{2\beta_t} \xi \\ &= -\dot{\eta}_t \alpha_t X_1 + \left(-\dot{\eta}_t \beta_t + \frac{\sigma_t^2}{2\beta_t}\right) \xi \\ &\stackrel{(*)}{=} \dot{\alpha}_t X_1 + \dot{\beta}_t \xi \\ &= \dot{X}_t.\end{aligned}$$

其中在 $\stackrel{(*)}{=}$ 处利用了 $\eta_t = -\frac{\dot{\alpha}_t}{\alpha_t}$ 和 $\sigma_t^2 = 2\beta_t^2 \left(\frac{\dot{\alpha}_t}{\alpha_t} - \frac{\dot{\beta}_t}{\beta_t}\right)$ ，这些都可从 (16) 推导得出。 $\square$

4 相关工作与讨论

单步模型学习 GAN [19, 4, 43]、VAE [32] 和（离散时间）标准化流 [62, 13, 14] 是深度生成模型的三种经典学习方法。GAN 在生成质量上最为成功（特别是图像生成），但因对抗式优化导致训练不稳定和模式崩溃。VAE 和标准化流都基于最大似然估计（MLE）原理，需要在模型架构上引入约束或采用特殊的近似技巧来保证似然计算的可行性：VAE 通常使用条件高斯分布和变分近似；标准化流则要求使用特殊设计的可逆架构，并需处理昂贵的雅可比矩阵计算。

本工作的 Reflow（重整流）+ 蒸馏方法提供了训练单步模型的另一条有前景的路径，既规避了 GAN 的对抗式优化问题，也避免了基于似然方法的不可行性问题。

常微分方程学习：最大似然和概率流 ODE 学习神经 ODE 有两条主要路径：第 2.3 节讨论的 PF-ODE/DDIM 方法，以及 [6] 的更经典的基于 MLE 的方法。

- 最大似然方法。在 [6] 中，神经 ODE 通过最大化时刻 $t = 1$ 处 ODE 结果 Z_1 在数据分布 π_1 下的似然来训练生成模型。具体地，给定来自 π_1 的观测，通过以下方式估计 ODE $dZ_t = v(Z_t, t)dt$ 的神经漂移 v ：

$$\max_v \mathbb{D}(\pi_1; \rho^{v, \pi_0}), \quad (19)$$

其中 $\mathbb{D}(\cdot; \cdot)$ 表示 KL 散度（或其他差异度量）， ρ^{v, π_0} 是从 $Z_0 \sim \pi_0$ 遵循 $dZ_t = v(Z_t, t)dt$ 得到的 Z_1 的密度； π_0 的密度应该已知且易于计算。

利用瞬时变量变换公式，[6] 观察到神经 ODE 的似然比离散时间标准化流更易计算，且无需对模型结构加约束。但这种 MLE 方法对大规模模型仍然计算成本高昂，因为每个训练步骤都需要反复

模拟 ODE。此外，MLE 的优化过程需要跨时间反向传播，容易产生梯度消失或爆炸问题，除非加入适当的正则化。

一个根本问题在于式 (19) 的神经 ODE MLE 在理论上不足以确定唯一解，因为 MLE 仅约束最终分布 Z_1 与数据分布 π_1 的匹配，而可实现同一输出分布的 ODE 路径有无穷多条。为解决这一问题，已有许多工作提出加入正则化项（如基于传输代价的项）来偏好更短的路径；见 [54, 55]。加入正则化项后，MLE 学到的 ODE 会由优化算法的初始化和其他超参数隐式决定。

- **概率流 ODE**。PF-ODE [73] 和 DDIM [70] 方法提供了学习 ODE 的另一条路径，避免了 MLE 方法的主要缺点，包括昂贵的似然计算、训练时 ODE 模型的模拟以及跨时间反向传播的需要。然而，因为 PF-ODE 和 DDIM 是作为学习更复杂的扩散/SDE 模型的副产品推导出来的，它们的理论和算法形式被设计得不必要地受限和复杂。非线性 Rectified Flow 框架表明，ODE 学习可以以极其简洁的方式直接进行，使我们能够识别线性 Rectified Flow 的规范情况，并通过灵活、解耦的插值曲线 X_t 和初始分布 π_0 选择来探索进一步的改进方向。

从一般非线性 Rectified Flow 框架看，MLE 的计算和理论缺陷可以被规避，因为我们可以通过指定插值曲线 X_t 预先确定 ODE 应该经过的“路径”，而不是让算法隐式地找出它。预先指定任意插值 X_t 在理论上是有效的，因为神经 ODE 作为生成模型是高度过参数化的：当 v 是通用近似器且 π_0 绝对连续时，给定任意固定的插值曲线 X_t ， Z_1 的分布可以近似任何分布。Rectified Flow 的思想在于为 X_t 选择最简单的测地线。

使用去噪扩散学习随机微分方程 虽然本工作的范围限于学习 ODE，但基于分数的生成模型 [71–74] 和去噪扩散概率模型 (DDPM) [23] 的相关性很高，因为它们是 PF-ODE 和 DDIM 的基础。用这些方法训练的扩散/SDE 模型在图像合成的质量和多样性上都超过了 GAN [12]。值得注意的是，得益于稳定可扩展的基于优化的训练过程，扩散模型已成功应用于大规模文本到图像生成，取得了令人惊艳的结果 [如 53, 61, 64]。它已迅速推广到其他领域，如视频 [如 24, 92, 21]、音乐 [51]、音频 [如 33, 40, 60] 和文本 [41, 88]，以及图像编辑等任务 [97, 50]。围绕改进去噪扩散模型的推理速度已形成一个快速增长的文献体系，其中 PF-ODE/DDIM 方法通过将 SDE 转为 ODE 来加速的案例就是一个例子。下面列举一些近期工作的例子，虽不穷尽但具有代表性。

- **改进训练和推理**。一类工作致力于改进去噪扩散模型的推理和采样过程。例如，[54] 提出了 DDPM 的若干简单修改，以改进似然、采样速度和生成质量。[29] 系统地检视扩散生成模型的设计空间，通过实证研究找出了多项训练和推理方案以获得更好的生成质量和更少的采样步数。[94] 提出了扩散指数积分器采样器用于快速采样。[46] 为 PF-ODE 提供了定制的高阶求解器。[5] 给出了最优扩散系数的解析估计。
- **与其他方法结合**。另一条路径是通过将扩散模型与 GAN 和其他生成模型结合来加速。DDPM 蒸

馏 [47] 通过将扩散模型的轨迹蒸馏到一系列条件 GAN 中来加速推理。截断扩散概率模型 (TDPM) [99] 训练一个 GAN 模型作为 π_0 , 使扩散过程可被截断以提高速度; 类似思想在 [48, 18] 中被探索, [18] 提供了最优截断时刻的分析。[68, 89, 81] 在潜在空间中学习去噪扩散模型并将其与变分自编码器相结合。这些方法有可能被应用到 Rectified Flow 中以获得类似的加速收益。

- 非配对图像到图像翻译。标准去噪扩散和 PF-ODE 方法聚焦于将高斯噪声 (π_0) 传输到数据 (π_1) 的生成任务。已有许多工作被提出以将其适配到任意源-目标领域对之间的数据传输。例如, SDEdit [50] 通过先对输入添加噪声, 再通过预训练的 SDE 模型去噪来合成由输入图像引导的逼真图像。[8] 提出了一种方法来指导 DDPM 的生成过程基于给定的参考图像生成逼真图像。[75] 利用两条 PF-ODE, 一条将源图像传输到潜变量, 另一条从潜变量构造目标图像。[97] 提出能量引导方法, 用预训练的源-目标能量函数指导 SDE 推理以改进图像翻译。相比之下, 我们的框架表明领域传输可以通过本质上与生成式建模相同的算法来实现, 仅需将 π_0 设为源领域。

- 扩散桥。最近的一些工作 [57, 44] 表明去噪扩散模型的设计空间可以通过使用在末时刻固定到某个数据点的扩散桥过程来变得高度灵活。这将去噪扩散方法的设计简化为构造适当的桥过程。在 Song et al. [73] 中的桥通过时间反演技术构造, 这可以等价地通过 Doob 的 h 变换实现, 如 [57, 44] 所示, 更一般的构造技术在 [44, 90] 中讨论。尽管设计空间显著扩展, 一个未回答的问题是应该偏好哪种扩散桥过程。这个问题具有挑战性, 因为扩散噪声的存在和高等随机微积分工具的需要使得直觉理解这些方法的工作原理困难重重。通过去除扩散噪声, 我们的工作明确表明直线路径应该被偏好。我们期望这个思想可以扩展以为学习 SDE 的最优桥过程设计提供指导。

- 薛定谔桥。另一类工作 [87, 11, 7, 82] 利用薛定谔桥 (SB) 作为学习扩散生成模型的另一条路径。这些方法在理论上具有吸引力, 但在求解薛定谔桥问题方面提出了重大计算挑战。

重新审视扩散噪声的作用 扩散噪声的引入曾被认为至关重要, 因为它在成功方法的推导中起关键作用 [73, 23]。然而, 因为 Rectified Flow 能以纯 ODE 框架达到更好或相当的结果, 扩散机制的作用应该被重新审视和明确地从去噪扩散模型的其他优点中解耦出来。去噪扩散模型的成功主要来自简单且稳定的基于优化的训练过程, 这避免了 GAN 的不稳定性和逐案例调优的繁琐, 而非扩散噪声本身。

因为我们的工作表明学习 ODE 无需使用 SDE 工具, 剩下的问题是对于给定问题是应该学习 ODE 还是 SDE。如已有多项工作论证的 [73, 70, 29], 通常应该偏好 ODE 而非 SDE。下面是 ODE 和 SDE 之间的详细对比。

- 概念简洁性和数值速度。SDE 在数学上更复杂, 更难理解。ODE 的数值模拟比 SDE 更简洁和更快。

- 时间可逆性。对 ODE 进行前向和后向求解同样容易。相比之下, SDE 的时间反演 [如 3, 22, 17]

在理论上更复杂，可能在计算上不可行。

- 潜在空间。ODE 的耦合 (Z_0, Z_1) 是确定性的，在 Rectified Flow 情况下产生低传输代价，因此为表示和操作输出提供了良好的潜在空间。引入扩散噪声使 (Z_0, Z_1) 更随机，因此效用较低。实际上，DDPM [23] 和 [73] 的 SDE 给出的 (Z_0, Z_1) 对潜在表示无用。
- 训练难度。没有理由相信训练 ODE 比训练共享相同边缘分布的 SDE 更难，如果不是更容易的话：两种情况的训练损失会共享协变量分布，仅在目标上不同。在 [73] 的设定中，两个损失函数 (15) 和 (18) 在线性重参数化意义下是等价的。
- 表达能力。如 [70, 73] 中的技术所示（也见 [84]），每个 SDE 都可以转换为具有相同边缘分布的 ODE，因此 ODE 对表示边缘分布与 SDE 一样强大，这正是本工作所考虑的传输映射问题所需的。另一方面，如果需要捕捉更丰富的时间相关结构，SDE 可能更为优选。
- 流形数据。当配备神经网络漂移时，ODE 的输出倾向于落在光滑的低维流形上，这是图像和文本等 AI 中结构化数据的关键归纳偏置。相比之下，使用 SDE 建模流形数据时，需要仔细退火扩散噪声以得到光滑的输出，这导致计算缓慢和超参数调整的负担。SDE 可能在金融和经济学等高噪声数据领域，以及涉及物理扩散过程的领域（如分子模拟）更有用。

最优传输与直线传输 最优传输在机器学习中已被广泛探索，是比较和传输概率测度之间的强大方式。对于本工作所考虑的传输映射问题，一条自然的方法是找到最小化给定 c 的传输代价 $\mathbb{E}[c(Z_1 - Z_0)]$ 的最优耦合 (Z_0, Z_1) 。最常见的 c 选择是二次代价 $c(\cdot) = \|\cdot\|^2$ 。

然而，找到最优耦合，特别是对于高维连续测度，计算上高度困难，是活跃研究的主题；见例如 [67, 34, 35, 49, 63, 10]。此外，虽然已知最优耦合具有良好的光滑性和其他正则性质，但不必精确找到最优耦合，因为传输代价与个别问题的学习表现并不完全一致；见例如 [34]。

相比之下，我们的 Reflow 过程找到直线耦合，这对给定的 c 而言并非最优（见 3.4 节）。从快速推理的角度，所有直线耦合都同样好，因为它们都产生直线 Rectified Flow，因此可以用单步欧拉法模拟。

5 实验

我们首先研究 Reflow 在玩具例子上的影响。之后，我们展示通过多次 Reflow，Rectified Flow 在 CIFAR-10 上达到最先进的性能。此外，它还能在高分辨率图像数据集上生成高质量图像。超越无条件图像生成，我们将该方法应用于非配对图像到图像翻译任务以生成视觉上高质量的图像对。

算法 我们遵循算法 1 中的过程。我们从抽取 $(X_0, X_1) \sim \pi_0 \times \pi_1$ 开始，并通过最小化 (1) 得到第一个 Rectified Flow Z^1 。第二个 Rectified Flow Z^2 通过相同的过程获得，除了数据被替换为从第一个 Rectified Flow Z^1 模拟得到的 (Z_0^1, Z_1^1) 的抽取。这个过程重复 k 次以得到 k -整流流 (k -rectified flow) Z^k 。最后，我们可以通过在 (Z_0^k, Z_1^k) 的抽取上进行拟合，进一步将 k -整流流 (k -rectified flow) Z^k 蒸馏到单步模型 $z_1 = z_0 + v(z_0, 0)$ 。

默认情况下，ODE 使用常数步长 $1/N$ 和 N 步的范数欧拉方法进行模拟，即 $\hat{Z}_{t+1/N} = \hat{Z}_t + v(\hat{Z}_t, t)/N$ ，其中 $t \in \{0, \dots, N\}/N$ 。我们使用 Scipy 的 5 阶（4 阶）龙格-库塔方法 [86]，记为 RK45，它根据用户指定的相对和绝对容差自适应地决定步长和步数 N 。在我们的实验中，我们遵循与 [73] 相同的参数。

5.1 玩具例子

为了精确地说明理论性质，我们在图 2, 3, 4, 5 的玩具例子中使用非参数估计器 $v^{X,h}(z, t)$ (5)。在实践中，我们用最近邻估计器近似 (5) 中的期望：给定从 (X_0, X_1) 抽取的样本 $\{x_0^{(i)}, x_1^{(i)}\}_i$ ，我们通过以下方式估计 v^X ：

$$v^{X,h}(z, t) \approx \sum_{i \in \text{knn}(z, m)} \frac{x_1^{(i)} - z}{1 - t} \omega_h(x_t^{(i)}, z) / \sum_{i \in \text{knn}(z, m)} \omega_h(x_t^{(i)}, z), \quad x_t^{(i)} = tx_1^{(i)} + (1 - t)x_0^{(i)},$$

其中 $\text{knn}(z, m)$ 表示 z 在 $\{x_t^{(i)}\}_i$ 中的前 m 个最近邻。我们发现结果对 m 的选择和带宽 h 不敏感（见图 7）。我们默认使用 $h = 1$ 和 $m = 100$ 。使用欧拉方法和 N 步的常数步长 $1/N$ 模拟流。我们使用 $N = 100$ 步，除非另有说明。

或者， v^X 可以参数化为神经网络并通过随机梯度下降或 Adam 进行训练。图 7 显示了当 v^X 参数化为 2 层隐藏层、两个隐藏层各 64 个神经元的全连接神经网络时的例子。我们看到神经网络与线性插值轨迹的拟合不如完美（在这个玩具例子中应该是分段线性的）。如图 7 所示，我们发现增强神经网络的平滑性（通过在训练期间增加 L2 正则化系数）可以帮助平直化流，除了整流效应之外。

在 2.2 节的图 3 中，平直性根据模拟的轨迹计算为 (3) 的经验估计。相对传输代价根据通过模拟流从 (Z_0, Z_1) 抽取的 $\{z_0^{(i)}, z_1^{(i)}\}_{i=1}^n$ 计算，形式为 $\frac{1}{n} \sum_{i=1}^n \|z_1^{(i)} - z_0^{(i)}\|^2 - \|z_1^{(i^*)} - z_0^{(i)}\|^2$ ，其中 $z_1^{(i^*)}$ 是通过解 $\{z_0^{(i)}\}$ 和 $\{z_1^{(i)}\}$ 之间的离散 L2 最优传输问题得到的 $z_0^{(i)}$ 的最优 L2 分配。应该注意的是，这个度量仅在低维中 useful，因为在高维情况中，即使 v^X 被设置为随机神经网络，它也倾向于为 0。这种误导性现象导致 [30] 做出了 DDIM 产生 L2 最优传输的错误假设。

5.2 无条件图像生成

我们在 CIFAR-10 和多个高分辨率数据集上测试 Rectified Flow 进行无条件图像生成。通过 Fréchet 初始距离（FID）和初始分数（IS）评估生成的图像的质量，以及通过召回分数评估生成的图像的

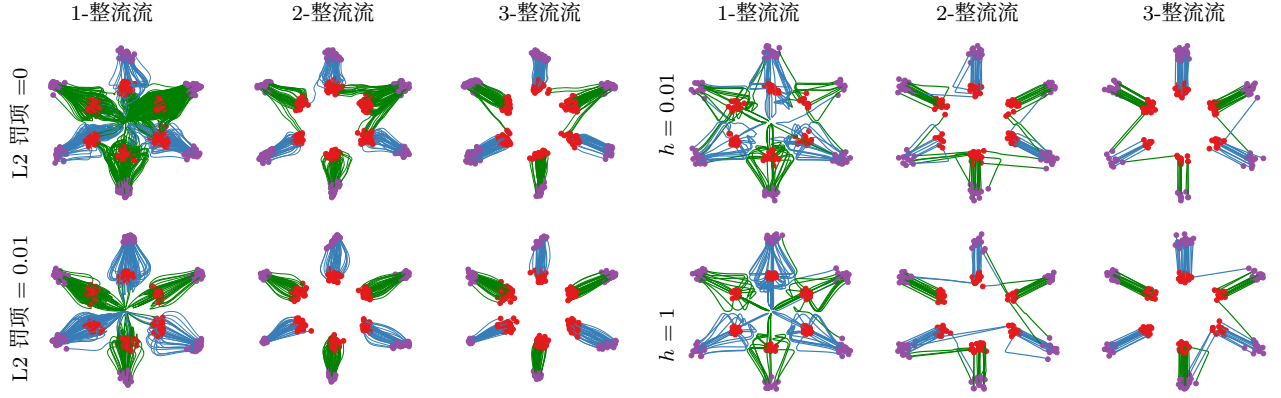


图 7: 通过不同 L2 罚项的神经网络拟合的 Rectified Flow (左), 以及不同带宽 h 的核估计器 (右)。 π_0 : 红点; π_1 : 紫点。

多样性, 遵循 [38]。

实验设置 为了生成式建模, 我们设置 π_0 为标准高斯分布, π_1 为数据分布。我们的 Rectified Flow 实现是基于 [73] 的开源代码修改的。我们采用 DDPM++ 的 U-Net 架构 [73] 来表示漂移 v^X , 并在表 1 (a) 和图 8 中报告我们的方法和 [73] 中的 (sub) -VP ODE 使用相同架构的结果。使用不同网络架构的其他最近结果在表 1 (b) 中作参考显示。更详细的设置可以在附录中找到。

方法	函数求值次数 (NFE) ($\downarrow$)	初始分数 (IS) ($\uparrow$)	FID ($\downarrow$)	召回 (Recall) ($\uparrow$)
常微分方程 (ODE) 单步生成 (欧拉求解器, $N=1$)				
1-Rectified Flow (+ 蒸馏)	1	1.13 (9.08)	378 (6.18)	0.0 (0.45)
2-Rectified Flow (+ 蒸馏)	1	8.08 (9.01)	12.21 (4.85)	0.34 (0.50)
3-Rectified Flow (+ 蒸馏)	1	8.47 (8.79)	8.15 (5.21)	0.41 (0.51)
VP ODE [73] (+ 蒸馏)	1	1.20 (8.73)	451 (16.23)	0.0 (0.29)
sub-VP ODE [73] (+ 蒸馏)	1	1.21 (8.80)	451 (14.32)	0.0 (0.35)
常微分方程 (ODE) 完整模拟 (龙格-库塔方法 (RK45), 自适应 N)				
1-Rectified Flow	127	9.60	2.58	0.57
2-Rectified Flow	110	9.24	3.36	0.54
3-Rectified Flow	104	9.01	3.96	0.53
VP ODE [73]	140	9.37	3.93	0.51
sub-VP ODE [73]	146	9.46	3.16	0.55
随机微分方程 (SDE) 完整模拟 (欧拉求解器, $N=2000$)				
VP SDE [73]	2000	9.58	2.55	0.58
sub-VP SDE [73]	2000	9.56	2.61	0.58

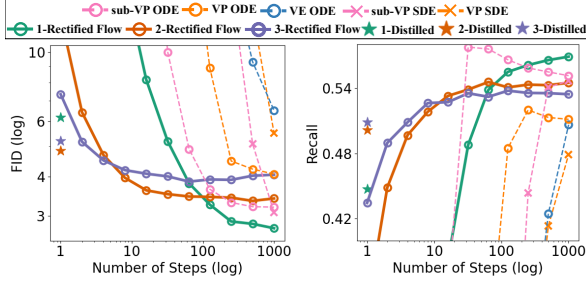
(a) 使用 DDPM++ 架构的结果。

方法	函数求值次数 (NFE) ($\downarrow$)	初始分数 (IS) ($\uparrow$)	FID ($\downarrow$)	召回 (Recall) ($\uparrow$)
生成对抗网络 (GAN) 单步生成				
SNGAN [52]	1	8.22	21.7	0.44
StyleGAN2 [28]	1	9.18	8.32	0.41
StyleGAN-XL [66]	1	-	1.85	0.47
StyleGAN2 + ADA [28]	1	9.40	2.92	0.49
StyleGAN2 + DiffAug [98]	1	9.40	5.79	0.42
TransGAN + DiffAug [26]	1	9.02	9.26	0.41
带 U-Net 的生成对抗网络 (GAN) 单步生成				
TDPm (T=1) [99]	1	8.65	8.91	0.46
去噪扩散生成对抗网络 (T=1) [91]	1	8.93	14.6	0.19
常微分方程 (ODE) 单步生成 (欧拉求解器, $N=1$)				
DDIM 蒸馏 [47]	1	8.36	9.36	0.51
NCSN++ (VE ODE) [73] (+ 蒸馏)	1	1.18 (8.57)	461 (25.4)	0.0 (0.0)
常微分方程 (ODE) 完整模拟 (龙格-库塔方法 (RK45), 自适应 N)				
NCSN++ (VE ODE) [73]	176	9.35	5.38	0.56
随机微分方程 (SDE) 完整模拟 (欧拉求解器)				
DDPM [24]	1000	9.46	3.21	0.57
NCSN++ (VE SDE) [73]	2000	9.83	2.38	0.59

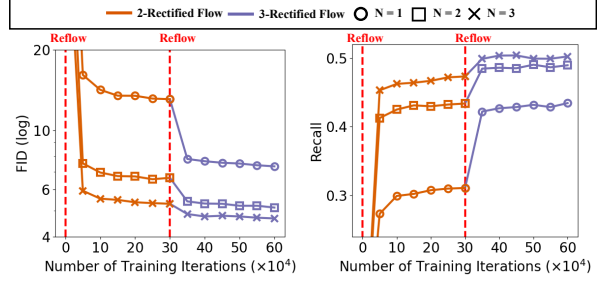
(b) 文献中报告的具有不同架构的最近结果。

表 1: CIFAR10 无条件图像生成的结果。Fréchet 初始距离 (FID) 和初始分数 (IS) 衡量生成图像的质量, 召回分数 [38] 衡量多样性。函数求值次数 (NFE) 表示推理期间调用主神经网络的次数。对于 ODE 和 SDE 模型, 它与离散化步数 N 一致。

结果 • 完整求解的 ODE 的结果。如表 1 (a) 所示, 在 DDPM++ 架构上训练的 1-整流流用 RK45 求解, 在所有基于 ODE 的方法中产生最低的 FID (2.58) 和最高的召回值 (0.57)。特别是, 0.57 的召回值相比现有的 ODE 和 GAN 方法产生了实质性的改进。使用相同的 RK45 ODE 求解



(a) FID 和召回值 vs. 欧拉离散化步数 N



(b) FID 和召回值 vs. 训练迭代

图 8: (a) Rectified Flow 和 (sub) -VP ODE 在 CIFAR10 上的结果, 具有不同数量 N 的欧拉离散化步数。(b) 不同 Reflow 和训练步数期间的 FID 和召回值。在 (a) 中, k -蒸馏 (k -Distilled) 指从 k -整流流 (k -Rectified Flow) 蒸馏的单步模型, 其中 $k = 1, 2, 3$ 。

器, Rectified Flow 相比 VE、VP、sub-VP ODE 需要更少的步数来生成图像。结果与完整模拟的 (sub) -VP SDE 相当, 后者产生的模拟代价相等。

- 少步和单步生成的结果。如图 8 所示, Reflow 过程在小步骤制度中 (例如, $N \lesssim 80$) 显著改进 FID 和召回值, 尽管它在大步骤制度中恶化了结果因为估计 v^x 的误差累积。图 8 (b) 显示每个 Reflow 在 FID 和召回值中都导致可见的改进。对于单步生成 ($N = 1$), 结果通过蒸馏进一步提升 (见图 8 (a) 中的星星)。总的来说, 蒸馏的 k -整流流 (k -Rectified Flow) 其中 $k = 1, 2, 3$ 产生单步生成式模型打败所有之前的带蒸馏的 ODE; 它们也打败了使用 GAN 训练的具有相似 U-Net 类型架构的单步模型的报告结果 (见表 1 (b) 中的 带 *U-Net* 的生成对抗网络 (*GAN with U-Net*))。

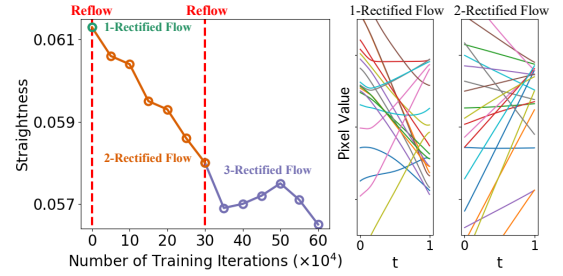


图 9: CIFAR10 上的平直化效应。左: 不同 Reflow 步数和训练迭代上的平直性度量。右: 随机采样的像素沿着 1-整流流和 2-整流流的轨迹。

特别地, 蒸馏的 2-整流流达到 FID 为 4.85, 打败了最好的已知的具有 U-Net 架构的单步生成式模型, 8.91 (TDPM, 表 1 (b))。2-整流流 (0.50) 和 3-整流流 (0.51) 的召回值都超过了 GAN 的最好已知结果 (0.49 来自 StyleGAN2+ADA) 展示了多样性的优势。应该注意的是, GAN 的报告结果已经用特殊技术进行了仔细优化, 如自适应鉴别器增强 (ADA) [28], 而我们的结果是基于 Rectified Flow 的范式实现。很可能通过适当的数据增强技术进一步改进 Rectified Flow, 或 GAN 的结合, 如 TDPM [99] 和去噪扩散生成对抗网络 [91] 所提出的。

- *Reflow* 平直化流。图 9 显示 Reflow 过程在 CIFAR10 上减少改进了流的平直性。在图 10 中可视化了 1-整流流和 2-整流流在 AFHQ 猫数据集上的轨迹: 在每个点 z_t , 我们通过 $\hat{z}_1^t = z_t + (1-t)v(z_t, t)$

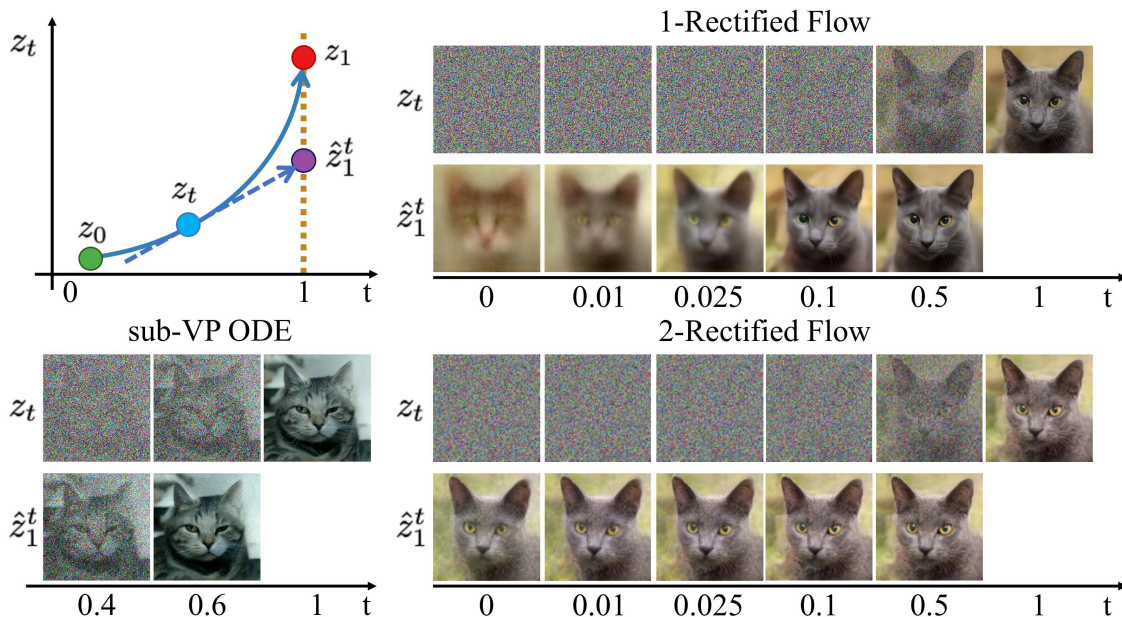


图 10: AFHQ 猫数据集上不同流的样本轨迹 z_t ，以及来自不同 z_t 的外推 $\hat{z}_1^t = z_t + (1-t)v(z_t, t)$ 。所有三个方法采用相同的随机种子。2-整流流的 $\hat{z}_1^t$ 几乎与 t 无关，表明其轨迹几乎是直的。

推断 $t=1$ 处的终端值；如果 ODE 的轨迹沿着直线，当沿着相同的路径跟随时，当我们改变 t 时， $\hat{z}_1^t$ 应该不改变。我们观察到 $\hat{z}_1^t$ 对于 2-整流流几乎与 t 无关，显示路径几乎是直的。此外，即使 1-整流流不是直的， $\hat{z}_1^t$ 随着时间的变化，它仍然在很早的时间 ($t \approx 0.1$) 产生可识别和清晰的图像。相比之下，从 sub-VP ODE 的外推需要 $t \approx 0.6$ 才能获得清晰的图像。

高分辨率图像生成 图 11 显示了 1-整流流在图像生成上的结果在高分辨率 (256×256) 数据集上，包括 LSUN Bedroom [93]、LSUN Church [93]、CelebA HQ [27] 到 AFHQ Cat [9]。我们可以看到它能在不同的数据集上生成高质量的结果。图 1 和 10 显示 1- (2-) 整流流在一个或几个欧拉步内产生好的结果。

图 12 显示了一个使用 1-整流流的简单图像编辑例子：我们首先通过拼接两个自然图像的上下部分获得非自然图像 z_1 ，然后反向运行 1-整流流获得潜在代码 z_0 。我们然后修改 z_0 来增加它在 π_0 下的似然（其为 $\mathcal{N}(0, I)$ ）以获得缝合图像更自然看起来的变体。

5.3 图像到图像翻译

假设给定两组不同风格的图像集合（分属不同领域），其分布分别记为 π_0, π_1 。我们的目标是在没有配对样本的情况下，将一个领域中的图像风格（或其他关键特征）迁移到另一个领域。经典的方法是循环一致对抗网络（CycleGAN）[100, 25]，它同时学习前向和反向映射 F, G ，通过最小化两个



图 11: 由 1-整流流生成的 256×256 图像的例子。

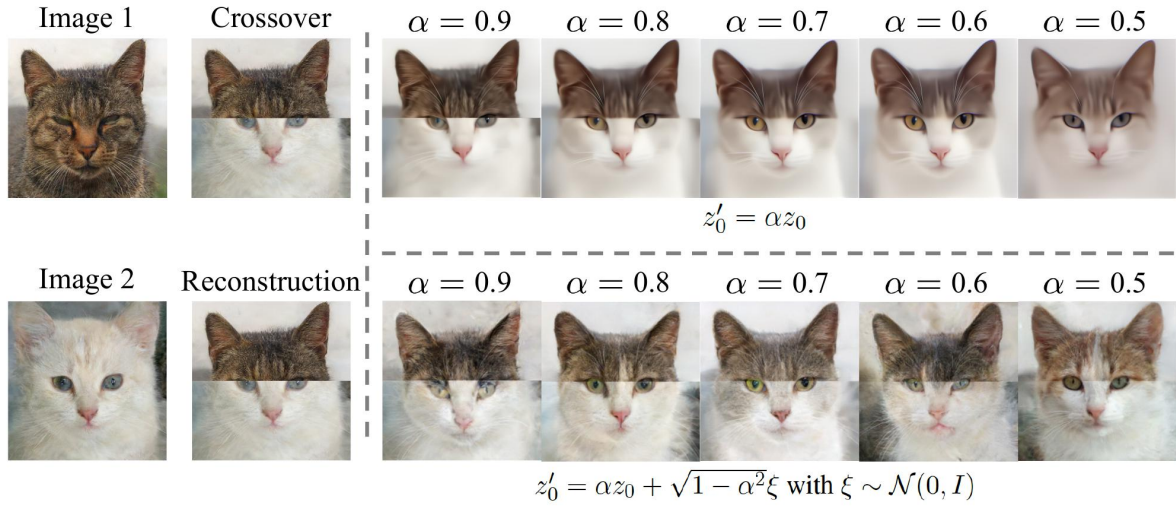


图 12: 一个使用 1-整流流的图像编辑例子。这里，我们将一个白色猫和一个黑色猫的图像拼接成一个非自然图像（记为 z_1 ）。我们从 z_1 反向模拟 ODE 获得潜在代码 z_0 。因为 z_1 不是自然图像， z_0 应该在 $\pi_0 = \mathcal{N}(0, I)$ 下有低似然。因此，我们移动 z_0 朝向 π_0 的高概率区域获得 z'_0 并向前求解 ODE 获得更现实看起来的图像 z'_1 。修改可以通过改进 π_0 -似然通过 $z'_0 = \alpha z_0$ 其中 $\alpha \in (0, 1)$ 确定性地完成，或通过 Langevin 动力学随机性地完成，其产生 $z'_0 = \alpha z_0 + \sqrt{1 - \alpha^2} \xi$ 其中 $\xi \sim \mathcal{N}(0, I)$ 的公式。



图 13: 由 1-整流流在不同领域之间模拟 $N = 100$ 欧拉步的样本。

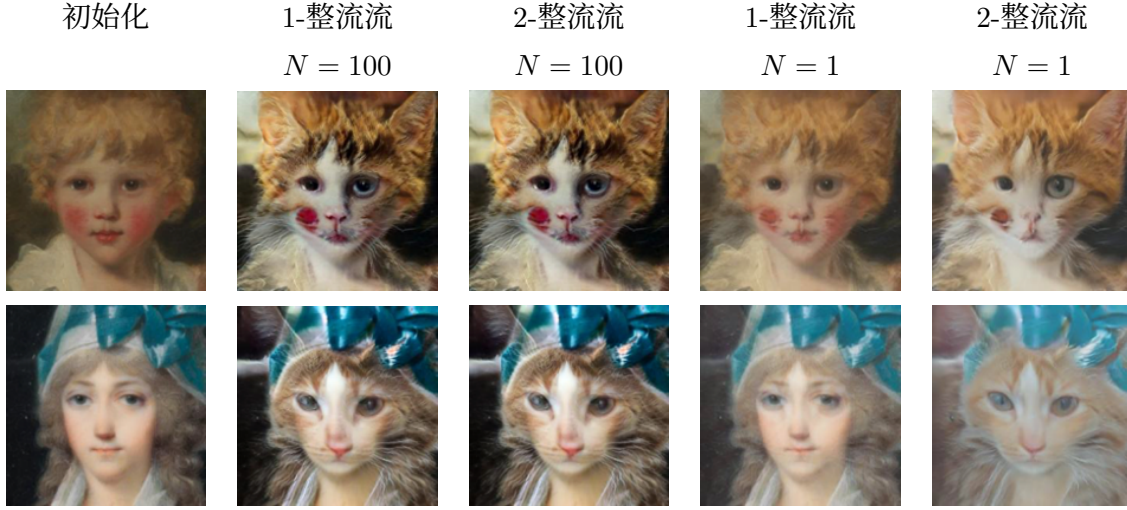


图 14: 1-整流流和 2-整流流用 $N = 1$ 和 $N = 100$ 欧拉步模拟的结果样本。

领域上的对抗损失之和，并用循环一致性损失进行正则化，以确保所有图像 x 满足 $F(G(x)) \approx x$ 。

通过构造 π_0 和 π_1 的 Rectified Flow，我们得到一个简单的图像翻译方法，无需进行对抗优化和循环一致性约束。训练 Rectified Flow 仅需简单的优化过程，而由于 ODE 的可逆性，循环一致性在流模型中得以自动满足。

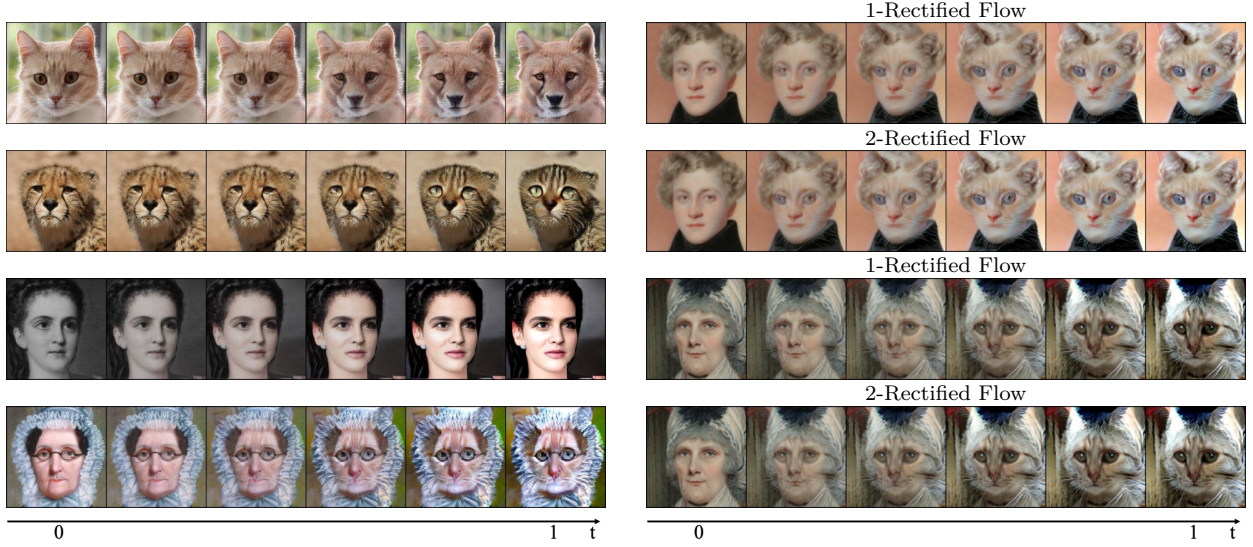
我们的主要目标是获得好的视觉效果。我们不关心将 $X_0 \sim \pi_0$ 完全精确地迁移到严格遵循 π_1 的 X_1 。相反，我们关注保留图像中主要对象的身份特征，同时迁移图像风格。例如，将人脸图像迁移到猫脸时，我们期望得到一个不真实的人猫混合脸，但仍能“看起来像”原始人脸。

为实现这一点，设 $h(x)$ 为图像 x 的特征映射，代表我们想要迁移的风格。令 $X_t = tX_1 + (1-t)X_0$ 。则 $H_t = h(X_t)$ 满足常微分方程 $dH_t = \nabla h(X_t)^\top (X_1 - X_0)dt$ 。为确保风格正确迁移，我们提议学习 v 使得 $H'_t = h(Z_t)$ ，其中 $dZ_t = v(Z_t, t)dt$ ，尽可能近似 H_t 。由于 $dH'_t = \nabla h(Z_t)^\top v(Z_t, t)dt$ ，我

们提议最小化以下损失：

$$\min_v \int_0^1 \mathbb{E} \left[\left\| \nabla h(X_t)^\top (X_1 - X_0 - v(X_t, t)) \right\|_2^2 \right] dt, \quad X_t = tX_1 + (1-t)X_0. \quad (20)$$

在实践中，我们将 $h(x)$ 设为一个分类器的潜在表示，该分类器经过从预训练 ImageNet 模型 [77] 微调，用于区分两个领域 π_0, π_1 中的图像。直观地， $\nabla_x h(x)$ 充当显著性分数，重新加权坐标，使式 (20) 中的损失集中于惩罚对 h 产生显著变化的误差。



(a) 不同领域间迁移的 1 次和 2 次整流流轨迹样例 (b) MetFace $\rightarrow$ 猫的 1 次和 2 次整流流轨迹样例。

图 15: 1 次和 2 次整流流在不同领域间迁移时的轨迹 z_t 样例。(a) 展示不同领域对间的迁移；(b) 展示 MetFace $\rightarrow$ 猫的迁移。

实验设置 我们将领域 π_0, π_1 设为 AFHQ [9]、MetFace [28] 和 CelebA-HQ [27] 数据集的配对。对于每个数据集，我们随机选择 80% 作为训练数据，其余作为测试数据；结果通过从测试数据初始化训练的流来展示。我们将图像缩放到 512×512 。训练和网络配置大体遵循 5.2 节的实验设置。详细描述见附录。

结果 图 1、13、14、15 展示了用欧拉方法以不同步数 N 模拟 1 次和 2 次整流流的结果示例。我们可以看到整流流能够成功迁移风格并生成高质量图像。例如，将猫迁移到野生动物时，我们可以生成具有不同动物脸多样图像，如狐狸、狮子、老虎和猎豹。此外，经过一次 Reflow（重整流），2 次整流流在单个欧拉步 ($N = 1$) 下仍能获得良好结果。更多示例见附录。

Method	ERM	IRM	ARM	Mixup	MLDG	CORAL	Ours
OfficeHome	66.5 ± 0.3	64.3 ± 2.2	64.8 ± 0.3	68.1 ± 0.3	66.8 ± 0.6	68.7 ± 0.3	69.2 ± 0.5
DomainNet	40.9 ± 0.1	33.9 ± 2.8	35.5 ± 0.2	39.2 ± 0.1	41.2 ± 0.1	41.5 ± 0.2	41.4 ± 0.1

表 2: 用不同方法在 OfficeHome 和 DomainNet 数据集上的迁移测试数据的精度。更高的精度表示更好的性能。

5.4 领域自适应

将机器学习应用于现实问题的关键挑战是训练和测试数据集之间的领域偏移：当在与训练集不同的新领域上测试时，机器学习模型的性能可能显著下降。整流流可用于将新领域 (π_0) 迁移到训练领域 (π_1) 以缓解领域偏移的影响。

实验设置 我们在多个数据集上测试整流流进行领域自适应。DomainNet [58] 是来自 DomainBed [20] 的六个不同领域的常见对象数据集。DomainNet 的所有领域包含 345 个对象类别（类），如手镯、飞机、鸟和大提琴。Office-Home [83] 是领域自适应的基准数据集，包含 4 个领域，每个领域有 65 个类别。为应用我们的方法，首先我们将训练和测试数据映射到预训练模型最后隐藏层的潜在表示，并在潜在表示上构造整流流。我们使用相同的 DDPM++ 模型架构进行训练。为推理，我们用均匀离散化将流模型的步数设为 100。方法通过在训练数据上训练的分类模型对迁移测试数据的预测精度进行评估。

结果 如表 2 所示，1 次整流流在 DomainNet 和 OfficeHome 上都展示了最先进的性能。它优于或与之前最佳方法（Deep CORAL [76]）相当，同时对所有其他方法有持续的改进。

参考文献

- [1] Luigi Ambrosio and Gianluca Crippa. Existence, uniqueness, stability and differentiability properties of the flow associated to weakly differentiable vector fields. In *Transport equations and multi-D hyperbolic conservation laws*, pages 3–57. Springer, 2008.
- [2] Luigi Ambrosio, Elia Brué, and Daniele Semola. *Lectures on optimal transport*. Springer, 2021.
- [3] Brian DO Anderson. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12(3):313–326, 1982.
- [4] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017.
- [5] Fan Bao, Chongxuan Li, Jun Zhu, and Bo Zhang. Analytic-DPM: an analytic estimate of the optimal reverse variance in diffusion probabilistic models. *arXiv preprint arXiv:2201.06503*, 2022.
- [6] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018.
- [7] Tianrong Chen, Guan-Horng Liu, and Evangelos A Theodorou. Likelihood training of Schrödinger bridge using forward-backward sdes theory. *arXiv preprint arXiv:2110.11291*, 2021.
- [8] Jooyoung Choi, Sungwon Kim, Yonghyun Jeong, Youngjune Gwon, and Sungroh Yoon. Ilvr: Conditioning method for denoising diffusion probabilistic models. *arXiv preprint arXiv:2108.02938*, 2021.
- [9] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. StarGAN v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8188–8197, 2020.
- [10] Max Daniels, Tyler Maunu, and Paul Hand. Score-based generative neural networks for large-scale optimal transport. *Advances in neural information processing systems*, 34:12955–12965, 2021.

- [11] Valentin De Bortoli, James Thornton, Jeremy Heng, and Arnaud Doucet. Diffusion Schrödinger bridge with applications to score-based generative modeling. *Advances in Neural Information Processing Systems*, 34, 2021.
- [12] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat GANs on image synthesis. *Advances in Neural Information Processing Systems*, 34, 2021.
- [13] Laurent Dinh, David Krueger, and Yoshua Bengio. Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*, 2014.
- [14] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.
- [15] Alessio Figalli and Federico Glaudo. *An Invitation to Optimal Transport, Wasserstein Distances, and Gradient Flows*. 2021.
- [16] R Flamary, N Courty, D Tuia, and A Rakotomamonjy. Optimal transport for domain adaptation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 1, 2016.
- [17] Hans Föllmer. An entropy approach to the time reversal of diffusion processes. In *Stochastic Differential Systems Filtering and Control*, pages 156–163. Springer, 1985.
- [18] Giulio Franzese, Simone Rossi, Lixuan Yang, Alessandro Finamore, Dario Rossi, Maurizio Filippone, and Pietro Michiardi. How much is enough? a study on diffusion times in score-based generative models. *arXiv preprint arXiv:2206.05173*, 2022.
- [19] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.
- [20] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. *arXiv preprint arXiv:2007.01434*, 2020.
- [21] William Harvey, Saeid Naderiparizi, Vaden Masrani, Christian Weilbach, and Frank Wood. Flexible diffusion modeling of long videos. *arXiv preprint arXiv:2205.11495*, 2022.
- [22] Ulrich G Haussmann and Etienne Pardoux. Time reversal of diffusions. *The Annals of Probability*, pages 1188–1205, 1986.
- [23] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

- [24] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *arXiv preprint arXiv:2204.03458*, 2022.
- [25] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.
- [26] Yifan Jiang, Shiyu Chang, and Zhangyang Wang. TransGAN: Two pure transformers can make one strong GAN, and that can scale up. *Advances in Neural Information Processing Systems*, 34, 2021.
- [27] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018.
- [28] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. *Advances in Neural Information Processing Systems*, 33:12104–12114, 2020.
- [29] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *arXiv preprint arXiv:2206.00364*, 2022.
- [30] Valentin Khrulkov and Ivan Oseledets. Understanding DDPM latent codes through optimal transport. *arXiv preprint arXiv:2202.07477*, 2022.
- [31] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [32] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.
- [33] Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. Diffwave: A versatile diffusion model for audio synthesis. In *International Conference on Learning Representations*, 2020.
- [34] Alexander Korotin, Lingxiao Li, Aude Genevay, Justin M Solomon, Alexander Filippov, and Evgeny Burnaev. Do neural optimal transport solvers work? a continuous wasserstein-2 benchmark. *Advances in Neural Information Processing Systems*, 34:14593–14605, 2021.

- [35] Alexander Korotin, Daniil Selikhanovych, and Evgeny Burnaev. Neural optimal transport. *arXiv preprint arXiv:2201.12220*, 2022.
- [36] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.
- [37] Thomas G Kurtz. Equivalence of stochastic equations and martingale problems. In *Stochastic analysis 2010*, pages 113–130. Springer, 2011.
- [38] Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. *Advances in Neural Information Processing Systems*, 32, 2019.
- [39] Hugo Lavenant and Filippo Santambrogio. The flow map of the fokker–planck equation does not provide optimal transport. *Applied Mathematics Letters*, page 108225, 2022.
- [40] Junhyeok Lee and Seungu Han. Nu-wave: A diffusion probabilistic model for neural audio upsampling. *arXiv preprint arXiv:2104.02321*, 2021.
- [41] Xiang Lisa Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori B Hashimoto. Diffusion-lm improves controllable text generation. *arXiv preprint arXiv:2205.14217*, 2022.
- [42] Qiang Liu. On rectified flow and optimal coupling. *preprint*, 2022.
- [43] Xingchao Liu, Chengyue Gong, Lemeng Wu, Shujian Zhang, Hao Su, and Qiang Liu. Fuse-dream: Training-free text-to-image generation with improved clip+ gan space optimization. *arXiv preprint arXiv:2112.01573*, 2021.
- [44] Xingchao Liu, Lemeng Wu, Mao Ye, and Qiang Liu. Let us build bridges: Understanding and extending diffusion generative models. *arXiv preprint arXiv:2208.14699*, 2022.
- [45] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.
- [46] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. DPM-solver: A fast ODE solver for diffusion probabilistic model sampling in around 10 steps. *arXiv preprint arXiv:2206.00927*, 2022.
- [47] Eric Luhman and Troy Luhman. Knowledge distillation in iterative generative models for improved sampling speed. *arXiv preprint arXiv:2101.02388*, 2021.

- [48] Zhaoyang Lyu, Xudong Xu, Ceyuan Yang, Dahua Lin, and Bo Dai. Accelerating diffusion models via early stop of the diffusion process. *arXiv preprint arXiv:2205.12524*, 2022.
- [49] Ashok Makkuva, Amirhossein Taghvaei, Sewoong Oh, and Jason Lee. Optimal transport mapping via input convex neural networks. In *International Conference on Machine Learning*, pages 6672–6681. PMLR, 2020.
- [50] Chenlin Meng, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021.
- [51] Gautam Mittal, Jesse Engel, Curtis Hawthorne, and Ian Simon. Symbolic music generation with diffusion models. *arXiv preprint arXiv:2103.16091*, 2021.
- [52] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations*, 2018.
- [53] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021.
- [54] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021.
- [55] Derek Onken, Samy Wu Fung, Xingjian Li, and Lars Ruthotto. Ot-flow: Fast and accurate continuous normalizing flows via optimal transport. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 9223–9232, 2021.
- [56] George Papamakarios, Eric T Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *J. Mach. Learn. Res.*, 22(57):1–64, 2021.
- [57] Stefano Peluchetti. Non-denoising forward-time diffusions. 2021.
- [58] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1406–1415, 2019.

- [59] Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019.
- [60] Vadim Popov, Ivan Vovk, Vladimir Gogoryan, Tasnima Sadekova, and Mikhail Kudinov. Grad-tts: A diffusion probabilistic model for text-to-speech. In *International Conference on Machine Learning*, pages 8599–8608. PMLR, 2021.
- [61] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.
- [62] Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *International conference on machine learning*, pages 1530–1538. PMLR, 2015.
- [63] Litu Rout, Alexander Korotin, and Evgeny Burnaev. Generative modeling with optimal transport maps. *arXiv preprint arXiv:2110.02999*, 2021.
- [64] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022.
- [65] Filippo Santambrogio. Optimal transport for applied mathematicians. *Birkäuser, NY*, 55 (58-63):94, 2015.
- [66] Axel Sauer, Katja Schwarz, and Andreas Geiger. StyleGAN-XL: Scaling StyleGAN to large diverse datasets. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Proceedings*, pages 1–10, 2022.
- [67] Vivien Seguy, Bharath Bhushan Damodaran, Rémi Flamary, Nicolas Courty, Antoine Rolet, and Mathieu Blondel. Large-scale optimal transport and mapping estimation. *arXiv preprint arXiv:1711.02283*, 2017.
- [68] Abhishek Sinha, Jiaming Song, Chenlin Meng, and Stefano Ermon. D2C: Diffusion-decoding models for few-shot conditional generation. *Advances in Neural Information Processing Systems*, 34:12533–12548, 2021.
- [69] Leslie N Smith and Nicholay Topin. Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial intelligence and machine learning for multi-domain operations applications*, volume 11006, pages 369–386. SPIE, 2019.

- [70] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2020.
- [71] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019.
- [72] Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. *Advances in neural information processing systems*, 33:12438–12448, 2020.
- [73] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2020.
- [74] Yang Song, Conor Durkan, Iain Murray, and Stefano Ermon. Maximum likelihood training of score-based diffusion models. *Advances in Neural Information Processing Systems*, 34, 2021.
- [75] Xuan Su, Jiaming Song, Chenlin Meng, and Stefano Ermon. Dual diffusion implicit bridges for image-to-image translation. *arXiv preprint arXiv:2203.08382*, 2022.
- [76] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European conference on computer vision*, pages 443–450. Springer, 2016.
- [77] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.
- [78] Anastasiya Tanana. Comparison of transport map generated by heat flow interpolation and the optimal transport brenier map. *Communications in Contemporary Mathematics*, 23(06): 2050025, 2021.
- [79] Giulio Trigila and Esteban G Tabak. Data-driven optimal transport. *Communications on Pure and Applied Mathematics*, 69(4):613–648, 2016.
- [80] Belinda Tzen and Maxim Raginsky. Theoretical guarantees for sampling and inference in generative models with latent diffusions. In *Conference on Learning Theory*, pages 3084–3114. PMLR, 2019.
- [81] Arash Vahdat, Karsten Kreis, and Jan Kautz. Score-based generative modeling in latent space. *Advances in Neural Information Processing Systems*, 34:11287–11302, 2021.
- [82] Francisco Vargas, Pierre Thodoroff, Austen Lamacraft, and Neil Lawrence. Solving Schrödinger bridges via maximum likelihood. *Entropy*, 23(9):1134, 2021.

- [83] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5018–5027, 2017.
- [84] Cédric Villani. *Optimal transport: old and new*, volume 338. Springer, 2009.
- [85] Cédric Villani. *Topics in optimal transportation*, volume 58. American Mathematical Soc., 2021.
- [86] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.
- [87] Gefei Wang, Yuling Jiao, Qian Xu, Yang Wang, and Can Yang. Deep generative learning via Schrödinger bridge. In *International Conference on Machine Learning*, pages 10794–10804. PMLR, 2021.
- [88] Rose E Wang, Esin Durmus, Noah Goodman, and Tatsunori Hashimoto. Language modeling via stochastic processes. *arXiv preprint arXiv:2203.11370*, 2022.
- [89] Antoine Wehenkel and Gilles Louppe. Diffusion priors in variational autoencoders. *arXiv preprint arXiv:2106.15671*, 2021.
- [90] Lemeng Wu, Chengyue Gong, Xingchao Liu, Mao Ye, and Qiang Liu. Diffusion-based molecule generation with informative prior bridges. *arXiv preprint*, 2022.
- [91] Zhisheng Xiao, Karsten Kreis, and Arash Vahdat. Tackling the generative learning trilemma with denoising diffusion GANs. *arXiv preprint arXiv:2112.07804*, 2021.
- [92] Ruihan Yang, Prakhar Srivastava, and Stephan Mandt. Diffusion probabilistic modeling for video generation. *arXiv preprint arXiv:2203.09481*, 2022.

- [93] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015.
- [94] Qinsheng Zhang and Yongxin Chen. Fast sampling of diffusion models with exponential integrator. *arXiv preprint arXiv:2204.13902*, 2022.
- [95] Qinsheng Zhang, Molei Tao, and Yongxin Chen. gDDIM: Generalized denoising diffusion implicit models. *arXiv preprint arXiv:2206.05564*, 2022.
- [96] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.
- [97] Min Zhao, Fan Bao, Chongxuan Li, and Jun Zhu. EGSDE: Unpaired image-to-image translation via energy-guided stochastic differential equations. *arXiv preprint arXiv:2207.06635*, 2022.
- [98] Shengyu Zhao, Zhijian Liu, Ji Lin, Jun-Yan Zhu, and Song Han. Differentiable augmentation for data-efficient GAN training. *Advances in Neural Information Processing Systems*, 33: 7559–7570, 2020.
- [99] Huangjie Zheng, Pengcheng He, Weizhu Chen, and Mingyuan Zhou. Truncated diffusion probabilistic models. *arXiv preprint arXiv:2202.09671*, 2022.
- [100] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.

算法 2 训练 (数据)

输入: Data={x0, x1}

输出: 整流流的模型 $v(x, t)$

初始化模型

```
for x0, x1 in Data: # x0, x1: 采样来自  $\pi_0, \pi_1$ 
    Optimizer.zero_grad()
    t = torch.rand(batchsize) # 随机采样  $t \in [0, 1]$ 
    Loss = ( Model(t*x1+(1-t)*x0, t) - (x1-x0) ).pow(2).mean()
    Loss.backward()
    Optimizer.step()
```

返回模型

算法 3 采样 (模型, 数据)

输入: 整流流的模型 $v(x, t)$

输出: 整流耦合的采样 (Z_0, Z_1)

coupling = []

```
for x0, _ in Data: # x0: 采样来自  $\pi_0$  (batchsize $\times$ dim)
    x1 = model.ODE_solver(x0)
    coupling.append((x0, x1))
```

返回 coupling

A 额外实验细节

CIFAR-10 上的实验配置 我们使用 CIFAR-10 数据集进行无条件图像生成 [36]。图像分辨率设置为 32×32 。对于 Rectified Flow, 我们采用与 DDPM++ 相同的网络结构 [73]。网络训练使用指数移动平均 (EMA) 平滑 [73], 系数为 0.999999。我们使用 Adam 优化器 [31], 学习率为 $2e - 4$, Dropout 比例为 0.15。

对于 Reflow, 我们首先生成 400 万对 (z_0, z_1) 得到新数据集 D , 然后对第 i 代 Rectified Flow 模型微调 300,000 步得到第 $(i + 1)$ 代 Rectified Flow 模型。进一步对这些 Rectified Flow 模型进行蒸馏以实现少步生成。为从第 i 代 Rectified Flow 获得 k 步图像生成器, 微调时随机采样 $t \in \{0, 1/k, \dots, (k - 1)/k\}$, 而不是随机采样 $t \in [0, 1]$ 。特别地, 对于 $k = 1$, 我们将 L2 损失函数替换为 LPIPS 相似性 [96], 实验表明这能获得更好的性能。

算法 4 Reflow(数据)

输入: $\text{Data}=\{x_0, x_1\}$ # 输出: 第 K 代整流耦合的采样

Coupling = Data

for $k = 1, \dots, K$:

Model = Train(Coupling)

Coupling = sample(Model, Data)

返回 Coupling

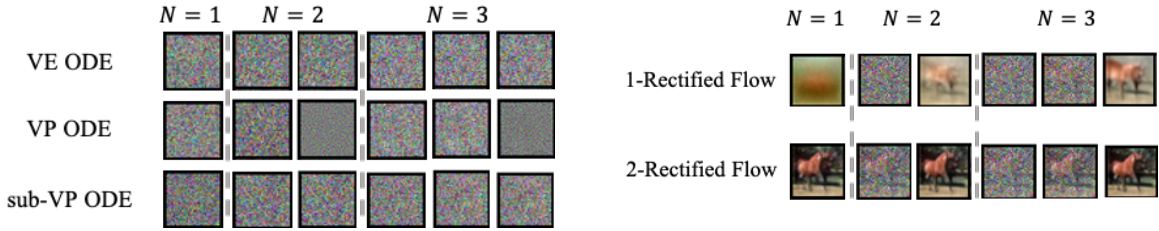


图 16: 不同 ODE 的少步生成对比。与 VE、VP、sub-VP ODE 相比, 1-代 Rectified Flow 仅用 1、2、3 步就能生成模糊图像。经过一次整流后, 2-代 Rectified Flow 能用 1、2、3 步生成清晰图像。

图像到图像翻译的实验配置 本实验同样采用 DDPM++ 的 U-Net 结构 [73] 来表示漂移场 v^X 。我们遵循算法 1 的流程。为进行生成式建模, 设 π_0 为一个领域数据集, π_1 为另一领域数据集。优化时使用 AdamW 优化器 [45], β 为 (0.9, 0.999), 权重衰减为 0.1, Dropout 比例为 0.1。训练模型的批大小为 4, 共 1,000 个 epoch。进一步应用系数为 0.9999 的指数移动平均 (EMA) 优化器。我们对学习率进行网格搜索, 范围为 $\{5 \times 10^{-4}, 2 \times 10^{-4}, 5 \times 10^{-5}, 2 \times 10^{-5}, 5 \times 10^{-6}\}$, 选择训练损失最低的模型。

我们使用 AFHQ [9]、MetFace [28] 和 CelebA-HQ [27] 数据集。AFHQ (Animal Faces HQ) 是一个动物脸部数据集, 包含 15,000 张 512×512 分辨率的高质量图像, 分为猫、狗和野生动物三个领域, 每个领域 5,000 张。MetFace 收集了 1,336 张 1024×1024 分辨率的高质量 PNG 人脸图像, 提取自艺术作品。CelebA-HQ 包含 30,000 张 1024×1024 分辨率的人脸图像。我们随机选取 80% 作为训练集, 余下部分作为测试集, 并将所有图像缩放至 512×512 。

领域自适应的实验配置 模型训练采用 AdamW 优化器 [45], 批大小为 16, 迭代次数 50 k, 学习率为 10^{-4} , 权重衰减为 0.1, 使用 OneCycle [69] 学习率调度。



图 17: 为可视化潜在空间, 我们从 $\mathcal{N}(0, I)$ 中随机采样 z_0 和 z_1 , 并显示 $\alpha \in [0, 1]$ 时 $\sqrt{\alpha}z_0 + \sqrt{1-\alpha}z_1$ 的生成图像。

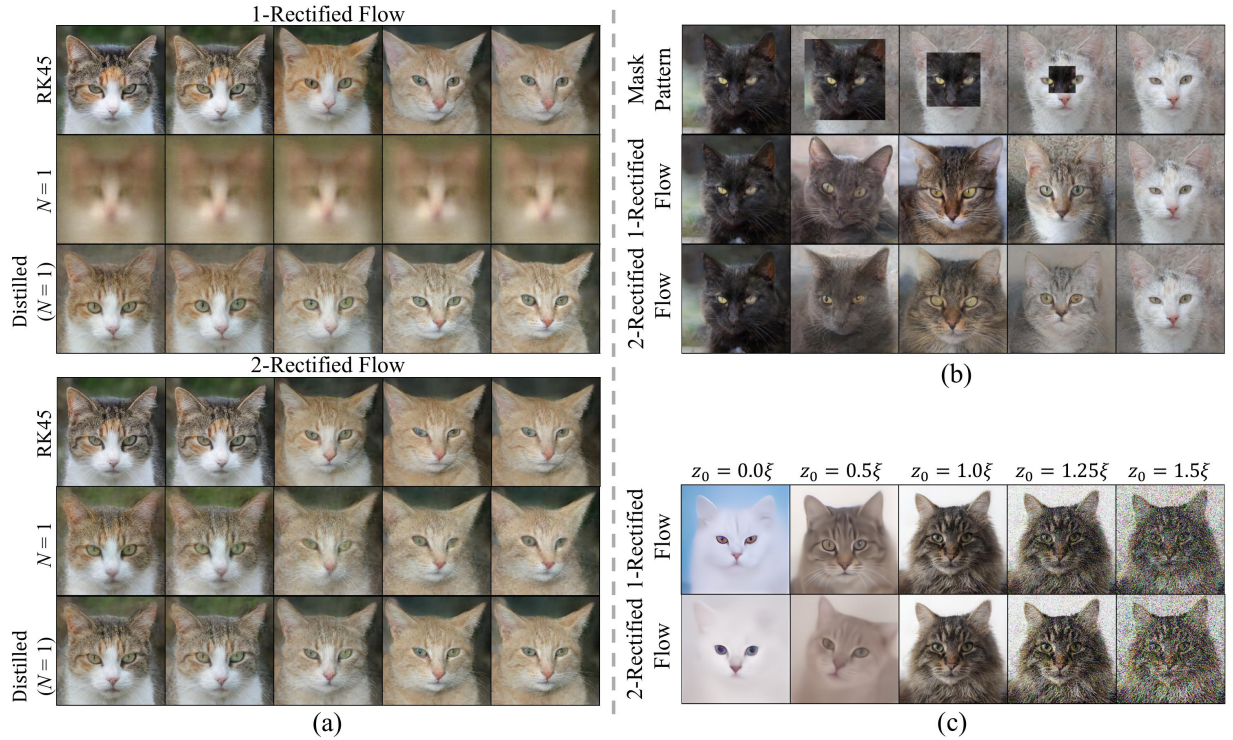


图 18: (a) 我们使用相同的随机种子, 用不同采样策略比较 0-代和 1-代 Rectified Flow 的潜在空间。观察发现: (i) 1-代和 2-代 Rectified Flow 都能提供平滑的潜在插值, 潜在空间相似; (ii) 使用单步采样 ($N = 1$) 时, 2-代 Rectified Flow 仍能提供视觉上可识别的插值, 而 1-代不行; (iii) 蒸馏的单步模型也能在图像间连续插值, 潜在空间与原始流差异很小。(b) 我们通过用白猫替换黑猫的边界来组合两张图像的潜在码, 然后可视化沿轨迹的变化。黑猫首先变成灰猫, 然后是混合颜色的猫, 最后变成白猫。(c) 我们随机采样 $\xi \sim \mathcal{N}(0, I)$, 生成 $\alpha\xi$ 对应的图像, 检验 α 对生成效果的影响。结果表明 $\alpha < 1$ 导致图像过度平滑, 而 $\alpha > 1$ 则产生噪声。

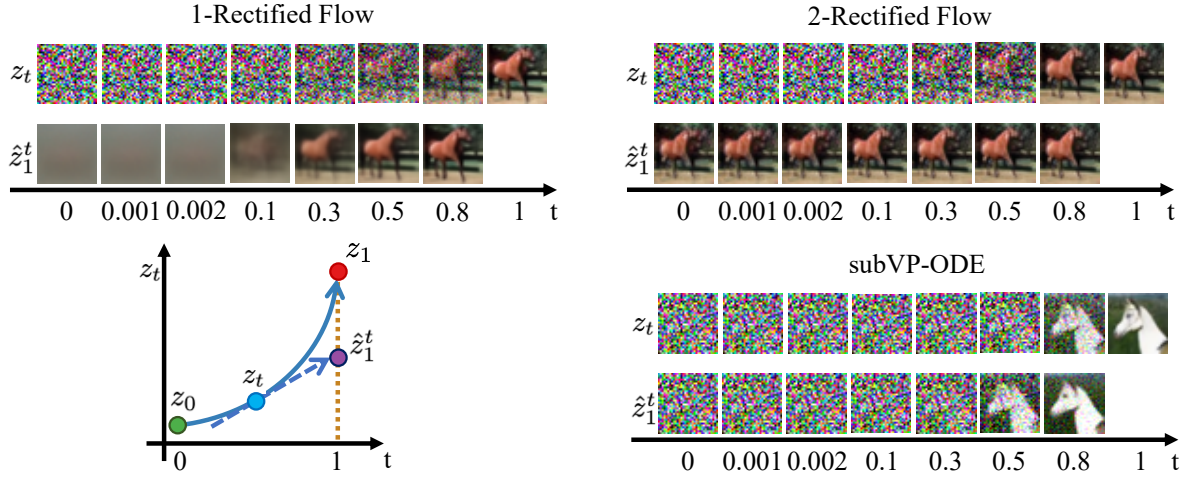


图 19: CIFAR-10 数据集上不同流的采样轨迹 z_t ，以及从不同 z_t 的外推 $\hat{z}_1^t = z_t + (1 - t)v(z_t, t)$ 。三种方法都使用相同的随机种子。2-代 Rectified Flow 的 $\hat{z}_1^t$ 几乎与 t 无关，表明其轨迹几乎是直线。

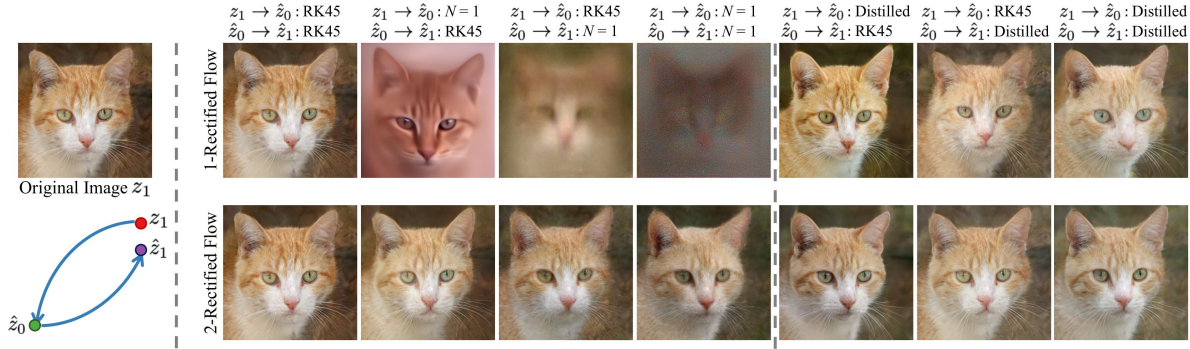


图 20: 我们在此进行潜在空间嵌入和图像重构。给定图像 z_1 ，使用反向 ODE 求解器得到潜在码 $\hat{z}_0$ ，再用前向 ODE 求解器得到图像的重构 $\hat{z}_1$ 。图中各列为反向 ODE 求解器（前向 ODE 求解器）。(i) 得益于平滑化效应，2-代 Rectified Flow 仅用一个反向步骤就能得到有意义的潜在码，也能用一个前向步骤生成可识别的图像。(ii) 借助蒸馏模型，单步嵌入和重构显著改进。

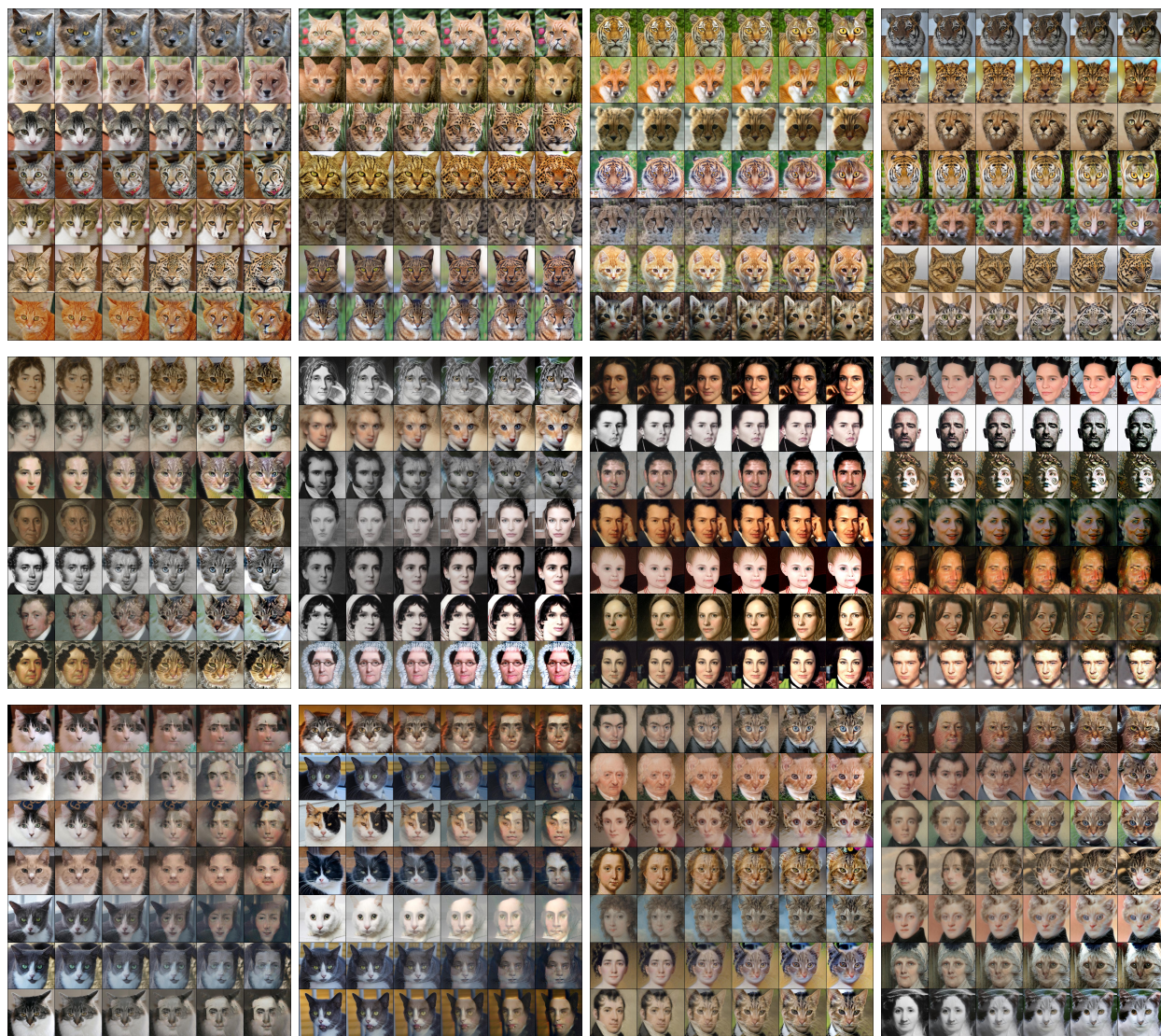


图 21: 不同领域间图像到图像翻译的更多结果。每行的图像是从用常数步长求解 $N = 100$ 欧拉步的 1-代 Rectified Flow 轨迹中均匀时间采样得到的。