# LaTtE-Flow：逐层时间步专家流变换器

**Ying Shen**[*1] **Zhiyang Xu**[*2] **Jiuhai Chen**[3] **Shizhe Diao**[4] **Jiaxin Zhang**[5]
**Yuguang Yao**[5] **Joy Rimchala**[5] **Ismini Lourentzou**[†1] **Lifu Huang**[†6]
[1]University of Illinois Urbana-Champaign [2]Virginia Tech
[3] University of Maryland [4]Nvidia [5]Intuit AI Research [6]UC Davis
ying22@illinois.edu, zhiyangx@vt.edu

## Abstract

最近在统一图像理解和生成的多模态基础模型方面的进展为在单一框架内解决广泛的视觉语言任务开辟了令人振奋的途径。尽管取得了一定的进展，现有的统一模型通常需要广泛的预训练，而且难以达到与专用于每个任务的模型相同的性能水平。此外，这些模型中的许多在图像生成速度上较慢，限制了其在实时或资源受限环境中的实际部署。在这项工作中，我们提出了 La 年基于 吨 次 时间 ep- xpert F l o w 的 Transformer（L 一个 吨 t E -F l o 终端电流），这是一种新颖且高效的架构，将图像理解和生成统一在一个多模态模型中。LaTtE-Flow 建立在强大的预训练视觉语言模型（VLMs）之上，以继承强大的多模态理解能力，并通过一种新颖的基于层次时间步专家的流架构来扩展，实现高效的图像生成。LaTtE-Flow 将流匹配过程分布在专门的 Transformer 层组中，每个组负责一组不同的时间步。这种设计通过在每个采样时间步中仅激活少部分层，大大提高了采样效率。为了进一步提高性能，我们提出了一种时间步调节的残差注意力机制，以便在层间高效地重用信息。实验表明，LaTtE-Flow 在多模态理解任务上表现强劲，同时与最近的统一多模态模型相比，以大约 6 倍的推断速度实现了具有竞争力的图像生成质量。[1]

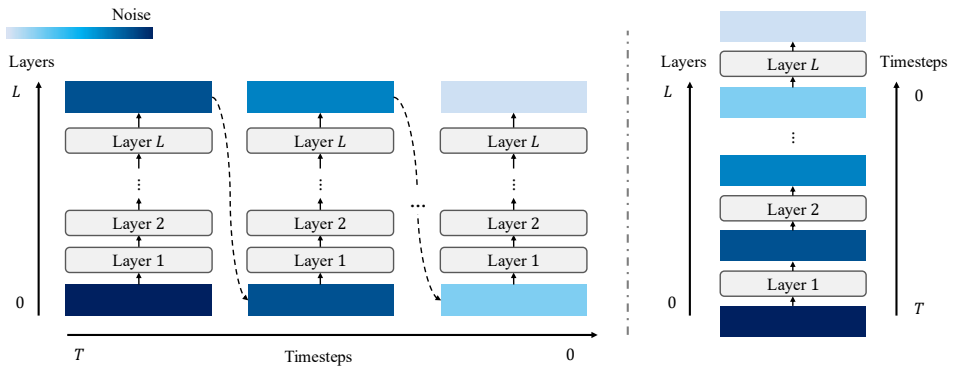
## 1 介绍

最近在多模态基础模型方面的进展，使得能够同时进行图像理解和生成的模型为构建执行广泛视觉语言任务的统一架构打开了有希望的途径 [? ? ? ? ? ? ? ]。这种统一的多模态模型在构建通用代理方面具有巨大潜力，这些代理可以根据用户指令解释、推理和生成多模态内容。目前对统一多模态建模的方法大体上可以分为两大类。第一类利用矢量量化自编码器 [? ? ? ] 将图像离散化为标记序列，然后将其纳入大型语言模型（LLMs）的词汇中 [? ? ? ? ? ? ? ]。这些模型随后被训练为自回归地生成下一个标记，无论是文本标记还是视觉标记，从而在单一框架内整合视觉和语言生成。第二类方法利用基于扩散的方法，或者通过与外部扩散模块耦合 LLMs，或者通过训练 LLMs 直接执行去噪步骤 [? ? ? ? ? ]。

尽管取得了显著进展，现有的统一多模态模型在多模态理解和图像生成方面往往难以同时取得高性能，因为提升一种模态的性能常常以牺牲另一种模态为代价。即使在两者都取得强劲表现时，通常也伴随着大量的计算开销。这些统一模型往往计算密集，推理速度慢，从而阻碍了实际的部署。例如，利用扩散或流匹配过程的统一模型通常需要在推理期间多次通过完整的骨干模型进行前向传递，导致推理速度慢且资源消耗高 [? ]。类似地，自回归方法也面临漫长的解码时间，尤其是对于需要顺序生成大量标记的高分辨率图像而言 [? ]。

[*]Ying Shen and Zhiyang Xu contributed equally to this work.
[†]Equal supervision.
[1] 代码和模型检查点可以在 https://github.com/yingShen-ys/LaTtE-Flow 找到。

Preprint. Under review.




**(a) Diffusion / Flow-Matching** **(b) LaTtE-Flow**

Figure 1: 标准扩散/流匹配模型与我们提出的 LaTtE-Flow 之间的流匹配过程比较。与基于扩散/流匹配的模型在每个采样时间步调用整个模型不同，LaTtE-Flow 在每一步仅激活部分层，提高了效率。

为了解决这些挑战，我们提出了 La yerwise 吨 imes t ep- E xpert F l o 的权重 -基于的 Transformer（L 一个 T t E - F l o w ），这是一种创新的架构，将高效的图像生成和多模态理解统一在一个模型中。特别地，LaTtE-Flow 引入了两个关键的架构创新，旨在实现高效且高质量的图像生成。首先，我们提出了一种新颖的 逐层时间步专家架构 ，通过在变换器层组之间分配流匹配过程，减少了采样时间复杂度。LaTtE-Flow 将变换器层划分为不相交的组，每个组在流匹配过程中分配到特定的时间步范围，如 1 图所示。在推理过程中，仅在每个时间步激活相关的专家组，这大幅减少了计算量，同时保持生成质量。其次，我们引入了 时间步条件残差注意力 ，这是一种轻量级机制，使后续层能够重用在之前层计算的自注意力图，并由当前时间步调制。这一设计鼓励模型在各层上逐步细化特征，从而在训练期间实现更快的收敛。

总结来说，我们的贡献有：（1）我们提出了 LaTtE-Flow ，一种高效且统一的多模态架构，它将基于流匹配的图像生成与预训练的视觉语言模型相结合。（2）我们引入了一种层次时间步专家，一种新颖设计，通过将变换器层分配给特定时间步的专家，显著降低推理复杂度。（3）我们设计了一个时间步条件残差注意模块，使得可以有效复用跨层的注意信息，提高训练效率和性能。（4）大量实验表明，LaTtE-Flow 在生成和理解任务上都实现了有竞争力的性能，而在推理速度方面比近期的统一模型快 6 × 。

## 2 相关工作

**统一模型。** 统一的多模态架构将多模态理解和生成整合在一个模型中，使得通用智能体能够根据用户指令解释和生成多模态内容 [? ? ? ? ? ? ? ] 。现有统一建模的方法大致可分为两类：第一类模型依赖于向量量化自编码器 [? ? ? ] ，将图像转换为离散的符号序列，从而可以类似于文本进行处理。这些视觉符号被添加到 LLM 词汇表中，以支持语言和视觉的统一自回归训练 [? ? ? ? ? ? ? ] 。第二类则结合了连续生成过程，最著名的是扩散模型 [? ] 或流匹配模型 [? ] 。一些方法通过外部扩散模块连接 LLM，使用语言模型指导图像生成 [? ? ? ? ? ] ，而另一些方法直接训练 LLM 以共同执行去噪或流匹配步骤 [? ? ? ] 。尽管在这两类中都有进展，但许多模型在图像生成速度上较慢，限制了它们在实时或资源受限环境中的实际部署。

**扩散模型中的多专家** 近年来，扩散模型的进展越来越多地采用模块化或基于专家的架构来实现更好的图像生成 [? ? ] 。基于这一方向，最近的一些方法探讨了使用针对不同扩散时间步调整的专家模型 [? ? ? ] 。通过为特定时间间隔分配不同的专家，这些模型旨在更好地捕捉去噪过程的发展特性。这种设计部分受到了之前研究的启发，该研究表明来自不同时间步的优化梯度常常存在冲突，导致收敛速度减慢和模型性能下降 [? ? ] 。然而，这些模型通常在不同的时间步间隔内维持一个接近完整参数的专家网络，这在固定的采样步骤数量下几乎不会提高推理效率。与此相反，我们引入了一种分层时间步专家架构，将变换器层划分为不同的层组，每组负责特定范围的时间步。在推理时，只激活相应的层组，从而显著减少每步涉及的参数数量。此外，我们的设计允许所有专家组联合训练，并进一步将其整合在一个统一的模型架构中，提升效率和性能。

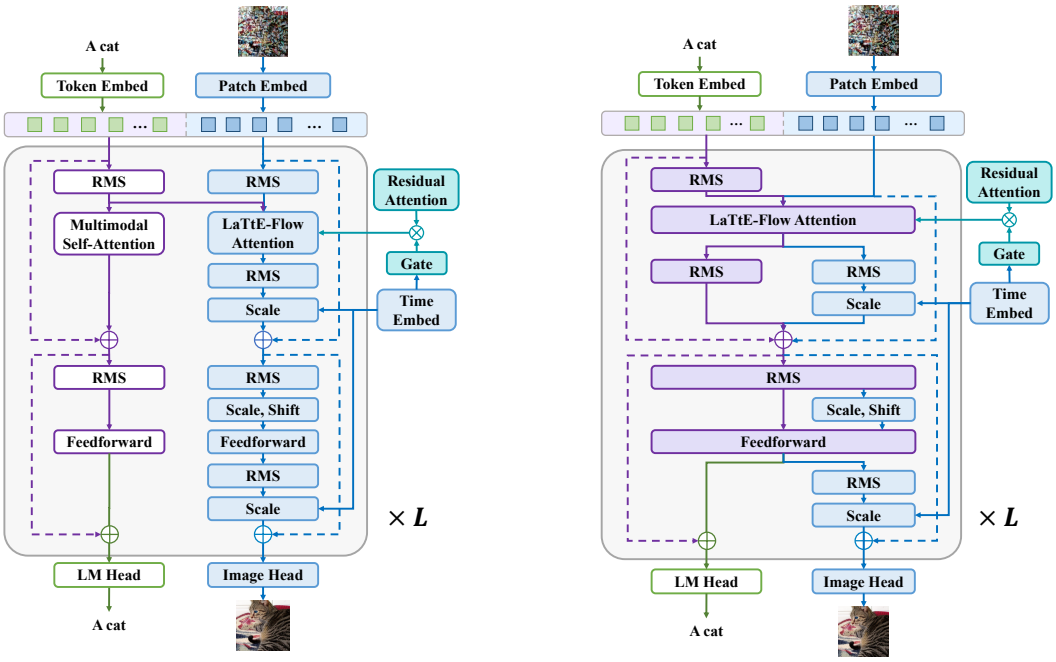



Figure 2: L 一个 吨 时间 E - F 小写字母 L o w 整体架构。

## 3 预备知识

**流动匹配。** 基于流的生成模型 [? ? ?] 旨在学习一个时间依赖的速度场 $\boldsymbol{v}_t$，通过常微分方程（ODE）将样本从简单的源分布 $p_0(\boldsymbol{x})$（例如，标准高斯分布）传输到复杂的目标分布 $p_1(\boldsymbol{x})$：最近，? ] 提出了一种简单的无仿真条件流匹配（CFM）目标，通过为每个样本 $\boldsymbol{x}_1$ 定义条件概率路径 $p_t(\boldsymbol{x}_t \mid \boldsymbol{x}_1)$ 及相应的条件向量场 $\boldsymbol{u}_t(\boldsymbol{x}_t \mid \boldsymbol{x}_1)$。该模型直接在条件向量场 $\boldsymbol{u}_t(\cdot \mid \boldsymbol{x}_1)$ 上回归速度 $\boldsymbol{v}_t$：其中 $\boldsymbol{u}_t(\cdot \mid \boldsymbol{x}_1)$ 独特地确定一个朝向目标数据样本 $\boldsymbol{x}_1$ 的条件概率路径 $p_t(\cdot \mid \boldsymbol{x}_1)$。条件概率路径广泛采用的选择是在源数据和目标数据之间进行线性插值 [? ]：$\boldsymbol{x}_t = t\boldsymbol{x}_1 + (1-t)\boldsymbol{x}_0$。假设源分布 $p_0$ 是一个标准高斯分布，这产生 $\boldsymbol{x}_t \sim \mathcal{N}(t\boldsymbol{x}_1, (1-t)^2\boldsymbol{I})$。从学习到的模型中采样可以通过首先采样 $\boldsymbol{x}_0 \sim \mathcal{N}(\boldsymbol{x} \mid 0, 1)$，然后在方程（?? ）中数值求解 ODE 获得。

## 4 L 一个 T 时间 E - F l o w

我们提出了 LaTtE-Flow （Layerwise Timestep-Expert Flow-based Transformer ），这是一种新颖的架构，旨在实现高效且高质量的图像生成和多模态理解，并在单一模型中实现统一。该架构构建在预训练视觉-语言模型（VLMs）之上，LaTtE-Flow 利用了它们强大的理解能力，同时引入了额外的基于流匹配的生成组件，从而实现可扩展且有效的图像合成。为了有效地统一生成和理解，我们探索了两种架构设计：LaTtE-Flow Couple 和 LaTtE-Flow Blend ，这两种设计在图 2 中进行了说明。这些变体主要区别在于生成和理解组件如何在 Transformer 层中组合（第 4.1 节）。

此外，我们引入了两个适用于这两种变体的核心架构创新，以提高图像生成的效率和质量：（1）层次时间步专家（第 4.2 节），将模型划分为时间步特定的模块，以降低采样复杂度；（2）时间步调节残差注意力（第 4.3 节），通过由学习到的时间步嵌入调制的门控机制将时间步感知的残差注意力注入每个注意力层，通过在层间有效重复利用信息提高训练效率。

### 4.1 LaTtE-Flow 层设计

LaTtE-Flow Couple 完全保留了预训练的 VLM，即保持其参数冻结（在图 2 中的 紫色 所示），以在不进行微调的情况下保留强大的多模态理解能力。为了实现图像生成，它在冻结的骨干网络旁边引入了一个可训练的生成路径。具体来说，每个 Transformer 层都通过可训练的

原始 VLM 层副本进行增强，并配备用于基于流匹配的生成的附加组件（在图 2 中的 蓝色 所示）。因此，LaTtE-Flow Couple 允许模型在利用预训练 VLM 的强大理解能力的同时进行图像合成。

LaTtE-Flow Blend 通过部分共享的 transformer 层统一了图像生成和理解组件。其中，每一层由具有生成和理解所需的单独参数的任务特定子模块和两者均使用的一组共享子模块组成。此设计支持生成与理解信号之间更加紧密的融合，在保持为每种模式专业化的灵活性的同时，促进更有效的信息交换。

如图 2 所示，两种 LaTtE-Flow 变体都引入了一个 LaTtE-Flow 注意模块，以实现生成图像潜变量与多模态上下文之间的有效交互。具体来说，在基于流的生成过程中使用的噪声图像潜变量会关注文本和视觉上下文标记，详细信息见附录 A 。该注意模块采用了一种混合位置编码方案，结合了从预训练的 VLM 继承的原始 3D 旋转位置嵌入（RoPE） [**?** ]，用于编码多模态上下文中的空间和时间结构，以及应用于生成图像标记的新引入的 2D 位置编码。

### 4.2 层级时间步专家

在扩散模型 [**? ?** ] 或流匹配模型 [**? ? ?** ] 中，典型的采样过程需要在大量时间步上反复调用整个网络，导致推理时间速度缓慢。例如，考虑一个具有 $L$ 层的标准扩散变压器（DiT）模型 [**?** ] 。如图 1 （a）所示，$T$ 采样步骤的有效计算成本是 $\mathcal{O}(L \times T)$ 。为了解决这种低效问题，我们引入了一种新颖的分层时间步专家架构，通过在变压器层组之间分配流匹配过程，降低了有效采样时间复杂度。

具体来说，我们不是在每个时间步上执行整个模型，而是将 $L$ 变压器层划分为 $K$ 个不重叠的组，每个组专门负责在特定的时间步区间内对样本进行去噪，如图 1 (b) 所示。这种设计有效地实现了高效采样，因为在每个时间步只需要执行网络的一个子集。

令每个专家组表示为 $\mathcal{G}_k^{l,l+M} = \{l, l+1, \ldots, l+M\}$ ，由 $M = L/K$ 个连续层组成（从层 $l$ 到层 $l+M$ ）。在训练期间，每个层组学习预测其分配的时间步区间 $[t_k, t_{k+1}]$ 上的速度场，使用逐层流匹配损失。具体而言，每个层组 $\mathcal{G}_k^{l,l+M}$ 接收先前层 $l-1$ 中获得的噪声潜在图像 $\boldsymbol{x}_t \in \mathbb{R}^{N_x \times d}$ 以及多模态上下文 $\boldsymbol{m}^l$ ，并预测速度场 $\boldsymbol{s}_\theta(\boldsymbol{x}_t, \boldsymbol{m}^l, t)$ 。正式地，对于时间步 $t \in [t_k, t_{k+1}]$ ，逐层流匹配损失定义为：

$$\mathcal{L}_t = \mathbb{E}_{t, p_1(\boldsymbol{x}_1), p_t(\boldsymbol{x}_t|\boldsymbol{x}_1)} \left\| \mathcal{G}_k^{l,l+M}(\boldsymbol{x}_t, \boldsymbol{m}^l, t) - \boldsymbol{u}_t(\boldsymbol{x}_t \mid \boldsymbol{x}_1) \right\|^2, \quad \text{for } t \in [t_k, t_{k+1}], \tag{1}$$

，其中 $\mathcal{G}_k^{l,l+M}(\cdot)$ 表示由专家组产生的预测，$\boldsymbol{u}_t(\boldsymbol{x}_t \mid \boldsymbol{x}_1)$ 是时间步 $t$ 的真实速度。通过仅在其各自的时间步区间上训练每个组，LaTtE-Flow 促进了时间步专门化，使模型能够学习跨流匹配过程的时间步特定表示。

推理。在推理时使用 $T'$ 采样步骤，我们首先预计算每个 Transformer 层所需的用于条件的多模态隐状态。这些多模态表示在推理开始时计算一次并缓存，以便在所有时步中重用。然后，对于每个时步 $t \in [t_k, t_{k+1}]$ ，只有相关的专家层组 $\mathcal{G}_k^{l,l+M}$ 被激活以执行从层 $l$ 到层 $M$ 的前向传递。这一过程在所有 $T'$ 时步中重复，每步仅评估 $M = L/K$ 层。与标准扩散模型或流匹配模型在每步执行所有 $L$ 层相比，这一设计显著降低了推理时的复杂性，从 $\mathcal{O}(L \times T')$ 减少到 $\mathcal{O}(M \times T')$ 。这在不牺牲生成质量的情况下显著降低了生成过程中的计算成本和延迟。

### 4.3 时间步条件残差注意力

为了促进信息在 transformer 层之间的重用，并提高训练效率和生成性能，我们提出了时间步条件残差注意力，这是一种新机制，它基于当前时间步在连续图像注意力层之间引入自适应残差连接。其目标是使后续层能够重用和完善在前面层中计算的注意力模式，同时通过当前流匹配时间步动态控制过去注意力的影响。

令 $\boldsymbol{A}^l \in \mathbb{R}^{N_x \times N_x}$ 为第 $l$ 层的图像自注意力矩阵，其中 $N_x$ 是图像标记的数量。在标准自注意力层中，注意力矩阵计算为：

$$\boldsymbol{A} = \text{Softmax}\left( \frac{(\boldsymbol{h}\boldsymbol{W}^Q)(\boldsymbol{h}\boldsymbol{W}^K)^T}{\sqrt{d}} \right), \tag{2}$$

，其中 $\boldsymbol{h} \in \mathbb{R}^{N_x \times d}$ 表示噪声图像潜变量的隐藏状态，$\boldsymbol{W}^Q, \boldsymbol{W}^K \in \mathbb{R}^{d \times d}$ 是可学习的查询和键投影矩阵。

为了结合来自前一层的残差注意力，我们将第 $l+1$ 层的增强自注意力矩阵定义为：

$$\tilde{\boldsymbol{A}}^{l+1} = \boldsymbol{A}^{l+1} + g(t) \odot \boldsymbol{A}^{l}, \quad g(t) = \tanh(\boldsymbol{h}_t \boldsymbol{W}_t), \tag{3}$$

，其中 $\boldsymbol{h}_t \in \mathbb{R}^d$ 是当前流匹配时间步 $t$ 的嵌入，$\boldsymbol{W}_t \in \mathbb{R}^{d \times H}$ 是一个可训练的投影矩阵，$d$ 表示隐藏维度，$H$ 为注意力头的数量。头部级门控向量 $g(t) \in (-1, 1)^H$ 由 $\tanh(\cdot)$ 激活产生，动态控制每个注意力头结合前一层残差注意力信息的程度。运算符 $\odot$ 表示逐元素乘法，广播到所有注意力头。值得注意的是，虽然 LaTtE-Flow 注意模块联合处理有噪声的图像状态和多模态隐藏状态，但残差注意力机制仅应用于有噪声图像隐藏状态上的自注意力图，如图 **??** 所示。

时间步长条件的残差注意力机制使模型能够动态控制每个头部在当前层中合入多少来自上一层的残差注意力，并以时间步长为条件。根据实验结果，这种设计加速了训练过程中的收敛性并提高了生成图像的质量。

## 5 实验设置

骨干模型和图像编码器。LaTtE-Flow 基于 Qwen2-VL-2B-Instruct [**?** ] 构建，该模型是由 $L=28$ 个 Transformer 层组成的预训练视觉语言模型。在 LaTtE-Flow Couple 变体中，我们从原始的 Qwen2-VL-2B-Instruct 中创建每个 Transformer 层的可训练副本，并将其与为流匹配的图像生成量身定制的附加组件集成。这些重复的组件使用原始视觉语言模型中对应的预训练权重进行初始化。对于图像编码，我们采用了最近提出的深度压缩自动编码器（DC-AE）[**?** ]，该编码器使用 32× 下采样率将原始图像像素压缩到紧凑的潜在空间中。

时间步分布。为了实现逐层时间步专家，LaTtE-Flow 将模型划分为 $K=4$ 个不重叠的层组，每个组包含 $M=7$ 个连续的层用于最终结果。设计这些组是为了在流匹配时间步的不同区间上操作。训练期间，我们使用 $T=1000$ 个流匹配步骤，最初将其均匀地划分为四个区间。为了增强在区间边界附近的稳健性并促进跨组平滑过渡，我们在训练过程中在相邻的时间步区间之间引入了 100 步的重叠。这种重叠允许边界时间步被多个层组看到，从而提高泛化能力。在推理时，我们禁用重叠以保持时间步区间的严格分区。因此，在每个去噪步骤中，只有相应的专家层组被激活，每个推理步骤仅需要 $M=7$ 层。这与标准扩散或流匹配模型在每一步激活所有 $L=28$ 层形成了鲜明对比，大大提高了生成效率。更多细节见附录 B 。

基线架构。我们构造了两个基线模型：Vanilla Couple 和 Vanilla Blend ，它们分别匹配 LaTtE-Flow Couple 和 LaTtE-Flow Blend 的架构，但去除了层次时间步专家和时间步条件残差注意机制，使我们可以直接评估这些提出的机制的有效性。Vanilla Couple 基线保留了一个与原始 VLM 模块并行的生成路径。从概念上讲，它类似于之前的模型，例如 LMFusion [**?** ] ，它通过一个单独的分支增强语言模型以处理图像生成。相比之下，Vanilla Blend 在共享层内统一了生成和理解计算，类似于 Transfusion [**?** ] 的设计。

训练和评估细节。所有 LaTtE-Flow 变体（Blend 和 Couple ）在从 ImageNet [**?** ] 训练集中获取的 120 万张图像上进行训练，分辨率为 256 × 256 ，全球批处理大小为 2048，恒定学习率为 $5e$-4 ，共进行 240K 步。对于 Vanilla Blend 和 LaTtE-Flow Blend ，我们进行全参数微调，而对于 Vanilla Couple 和 LaTtE-Flow Couple ，我们仅微调专门用于图像生成的参数，而保持图像理解的参数不变。评估时，我们报告 ImageNet 上的 FID、Inception Score、Precision 和 Recall，遵循以前的惯例 [**?** ] 。更多细节可以在附录 **??** 中找到。

## 6 结果与讨论

### 6.1 图像生成和理解结果

我们在图像生成（表格 1 ）和多模态理解（表格 2 ）任务中评估了 LaTtE-Flow 。表格 1 报告了 LaTtE-Flow 、最近的统一模型和领先的图像生成模型的定量对比。我们从生成质量、每个推理步骤激活的参数数量以及推理效率方面评估了每个模型的表现。所有的推理时间均在一台 NVIDIA L40 GPU 上以批量大小为 50 测量。LaTtE-Flow 在与最新的统一模型相比时获得了更好的 FID 分数 [**? ? ?** ] ，这些模型是在 ImageNet 和其他大规模图像-文字数据集的混合上预训练的，同时实现了更快的推理速度，即比 Show-o [**?** ] 快 48 × ，比 Janus Pro [**?** ] 快 6 × 。此外，LaTtE-Flow 的两个变体均优于各自的基准模型 Vanilla Blend 和 Vanilla Couple ，这些基准模型在概念上类似于 Transfusion [**?** ] 和 LMFusion [**?** ] ，且每个流匹配步骤所需激活的参数更少，推理速度快 3 到 4 × 。此外，相较于专门用于图像生成的扩散模型 [**?**

| | Model | FID | IS | Pre | Rec | # Params | # Step | Time (s / img) | Rel. Time |
|---|---|---|---|---|---|---|---|---|---|
| Diffusion Models | ADM [? ] | 10.94 | 101.0 | 0.69 | 0.63 | 554M | 250 | 9.677 | 168 |
| | CDM [? ] | 4.88 | 158.7 | – | – | – | 8100 | – | |
| | LDM-4-G [? ] | 3.60 | 247.7 | – | – | 400M | 250 | – | |
| | DiT-L/2 [? ] | 5.02 | 167.2 | 0.75 | 0.57 | 458M | 250 | 1.786 | 31 |
| | DiT-XL/2 [? ] | 2.27 | 278.2 | 0.83 | 0.57 | 675M | 250 | 2.592 | 45 |
| Masked Models | MaskGIT [? ] | 6.18 | 182.1 | 0.80 | 0.51 | 227M | 8 | 0.029 | 0.5 |
| | MAGE [? ] | 6.93 | 195.8 | – | – | 230M | – | – | |
| AR Models | VQVAE-2 † [? ] | 31.11 | ~ 45 | 0.36 | 0.57 | 13.5B | 5120 | – | |
| | VQGAN † [? ] | 18.65 | 80.4 | 0.78 | 0.26 | 227M | 256 | 1.094 | 19 |
| | VQGAN [? ] | 15.78 | 74.3 | – | – | 1.4B | 256 | 1.382 | 24 |
| | ViT-VQGAN [? ] | 4.17 | 175.1 | – | – | 1.7B | 1024 | 1.382 | 24 |
| | RQTran. [? ] | 7.55 | 134.0 | – | – | 3.8B | 68 | 1.210 | 21 |
| Unified Models | Show-o [? ] | 31.26 | 98.7 | 0.55 | 0.69 | 1.3B | 50 | 2.493 | 48 |
| | Janus Pro [? ] | 23.68 | 105.2 | 0.58 | 0.49 | 1.5B | 576 | 0.311 | 6 |
| | Vanilla Blend (Ours) | 6.12 | 193.7 | 0.78 | 0.69 | 2.0B | 40 | 0.185 | 4 |
| | LaTtE-Flow Blend (Ours) | 6.03 | 193.9 | 0.77 | 0.68 | 500M | 40 | 0.061 | 1 |
| | Vanilla Couple (Ours) | 6.33 | 192.4 | 0.80 | 0.67 | 2.0B | 40 | 0.158 | 3 |
| | LaTtE-Flow Couple (Ours) | 5.79 | 213.1 | 0.78 | 0.69 | 500M | 40 | 0.052 | 1 |

Table 1: 在 ImageNet-50K 上比较生成模型的 FID、IS、精度、召回率、参数、步骤和推理时间。对于 LaTtE-Flow ，我们报告了每个时间步激活的参数数量，因为它具有时间步专家架构，在每一步仅使用部分层。我们还报告了相对于 LaTtE-Flow Couple 的推理时间。†： 取自 MaskGIT [? ]

| Model | MMBench | SEED | POPE | MM-Vet | MME-P | MMMU | RWQA | TEXTVQA |
|---|---|---|---|---|---|---|---|---|
| EMU2 Chat 34B [? ] | - | 62.8 | - | 48.5 | - | 34.1 | - | 66.6 |
| Chameleon 7B [? ] | 19.8 | 27.2 | 19.4 | 8.3 | 202.7 | 22.4 | 39.0 | 0.0 |
| Chameleon 34B [? ] | 32.7 | - | 59.8 | 9.7 | 604.5 | 38.8 | 39.2 | 0.0 |
| Seed-X [? ] 17B | 70.1 | 66.5 | 84.2 | 43.0 | 1457.0 | 35.6 | - | - |
| VILA-U 7B [? ] | 66.6 | 57.1 | 85.8 | 33.5 | 1401.8 | 32.2 | 46.6 | 48.3 |
| EMU3 8B [? ] | 58.5 | 68.2 | 85.2 | 37.2 | 1243.8 | 31.6 | 57.4 | 64.7 |
| MetaMorph 8B [? ] | 75.2 | 71.8 | - | - | - | 41.8 | 58.3 | 60.5 |
| Show-o 1.3B [? ] | - | - | 80.0 | - | 1097.2 | 27.4 | - | - |
| Janus 1.5B [? ] | 69.4 | 63.7 | 87.0 | 34.3 | 1338.0 | 30.5 | - | - |
| Janus Pro 1.5B [? ] | 75.5 | 68.3 | 86.2 | 39.8 | 1444.0 | 36.3 | - | - |
| LaTtE-Flow Couple 2B | 74.9 | 72.4 | 87.3 | 51.5 | 1501.4 | 41.1 | 60.7 | 79.7 |

Table 2: 在综合图像理解基准上的结果。最佳分数以粗体显示。由于我们的 LaTtE-Flow Couple 是一个专家架构，我们报告了用于图像理解的激活参数数量。

**? ? ?** ]、掩码模型 [**? ?** ] 和自回归（AR）模型 [**? ? ? ?** ]，LaTtE-Flow 表现出具有竞争力的性能，实现了更好的参数和推理时间效率。这些结果表明，LaTtE-Flow 是一个有前途的、高效而有效的图像生成架构。在 ImageNet 上的定性结果见附录 C 。

表格 2 展示了在多模态理解基准测试上的结果 [**? ? ? ? ? ? ?** ]。LaTtE-Flow Couple 相较于最近的统一模型实现了具有竞争力或更优的性能，展示了其无需对理解任务进行额外微调的情况下，通过继承其强大能力来有效利用冻结的视觉-语言骨干网络的能力。

### 6.2 消融研究

**LaTtE-Flow 的收敛速度更快。** 图 3 展示了 LaTtE-Flow Blend 和 LaTtE-Flow Couple 与 Vanilla Blend 和 Vanilla Couple 的训练动态对比。

我们观察到，LaTtE-Flow Blend 和 LaTtE-Flow Couple 在训练过程中表现出显著更快的收敛速度，在更少的训练步骤中达到有竞争力的图像生成性能（更低的 FID）。我们将这个有利的特性归因于 LaTtE-Flow 的层级时间步专家架构。如先前研究所述 [**? ?** ]，扩散模型的缓慢收敛部分是由于不同时间步之间优化方向的冲突。优化接近的

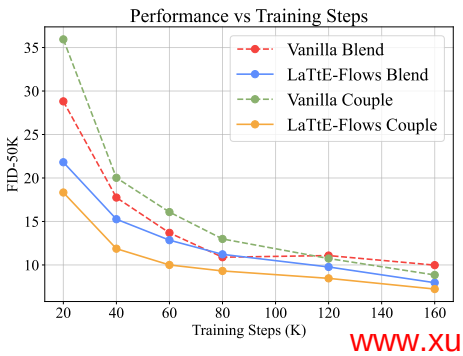


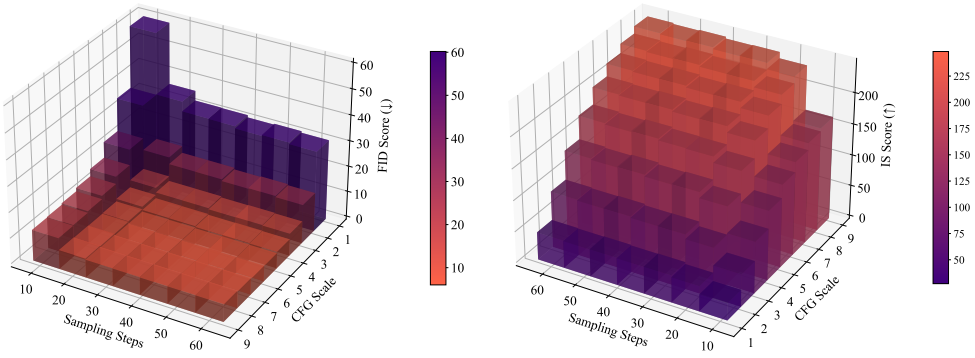

Figure 3: LaTtE-Flow 与基线的训练动态。在 ImageNet 50K 上的 FID

时间步可以相互受益，而优化相隔较远的时间步则可能相互干扰。我们的层级时间步专家架构通过将时间步分布在不同的 Transformer 层中，缓解了这一挑战。

我们还研究了时间步专家组大小 $M$ 如何影响生成质量和推理效率之间的权衡。具体而言，我们使用组大小为 $M \in \{4, 7, 14\}$ 的情况训练 LaTtE-Flow Couple，对应于将变压器层分别分成 7、4 和 2 个专家组。图 **??** 报告了在 120K 个训练步骤上的结果。我们观察到较大的组大小由于增加了建模能力，一致性地提高了生成质量（由 FID 衡量）。然而，这也带来了推理速度的下降，因为每个时间步执行更多层。与基线 Vanilla Couple （Vanilla）相比，$M = 7$ 和 $M = 14$ 都在生成质量和效率上达到了更好的效果，基线方法在每个步骤上应用所有 28 层。因此，考虑到性能和效率之间的权衡，我们在表 1 的主要结果中选择 $M = 7$ 作为默认的组大小，以提供强大的生成质量和显著的采样加速。

为了量化时间步条件残差注意力的效果，我们将 LaTtE-Flow Couple 与一个移除了时间步条件残差注意力的变体进行比较。如表 **??** 所示，移除残差注意力导致了在多个指标上的显著退化，突出了跨层时间条件注意力的有效性。添加时间步条件残差注意力不会引入额外的推理时间成本。

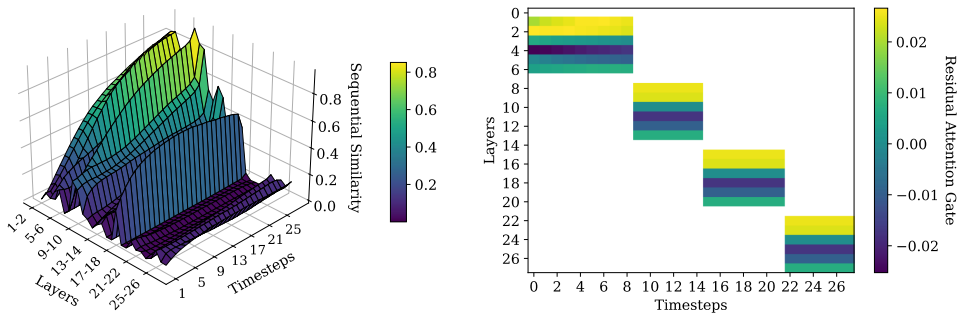
**采样步骤和 CFG 的影响。** 图 4 展示了改变采样步骤数量和无分类器指导缩放（CFG）对图像生成质量的影响。我们观察到增加步骤数量通常可以提高图像生成质量，从而降低 FID 并提高 Inception 分数。然而，当采样步骤数量超过 40 时，性能提升变得微不足道。总体而言，较高的 CFG 通常导致更好的 Inception 分数，但对于 FID，当 CFG 超过 5 时，性能开始略微下降。




(a) FID Score VS. Sampling Steps and CFG

(b) Inception Score vs Sampling Steps and CFG

Figure 4: # 采样步骤和 CFG 强度对 Inception Score 和 FID 的影响。




(a) Timestep-varying sequential similarity across adjacent transformer layers

(b) Timestep-conditioned residual attention gate for head 5 across transformer layers

Figure 5: 时间步条件化残差注意力分析。(a) 在 Vanilla Couple 中注意力行为的可视化，(b) 在 LaTtE-Flow Couple 中学习到的残差门控模式。

**残差注意中的时间步条件。** 为了更好地理解时间步长条件在残差注意力中的作用，我们对 LaTtE-Flow Couple 和 LaTtE-Flow Blend 进行深入分析。具体来说，我们首先调查了基线模型中注意力模式如何随着变压器层和采样时间步长演变。我们使用基于总变差的度量来量化每个时间步长相邻层之间的顺序相似性：

$$S(\boldsymbol{A}^l, \boldsymbol{A}^{l+1}) = 1 - 0.5 \sum_i \left| \text{Softmax} \left( \boldsymbol{A}_i^l \right) - \text{Softmax} \left( \boldsymbol{A}_i^{l+1} \right) \right|, \tag{4}$$

，其中 $\text{Softmax}\left(\boldsymbol{A}_i^l\right)$ 是注意力图 $\boldsymbol{A}^l$ 第 $i$ 行的 softmax 归一化结果。更高的 $S$ 值反映了连续层之间图像注意力图的更大相似性。

图 5 (a) 显示了在采样过程中 Vanilla Couple 的顺序相似性是如何演变的，计算平均值基于 100 个随机选择的样本。我们观察到，在采样初期，跨层的注意力图显示出较低的相似性，但随着生成的进展，特别是在后期的时间步中，相似性增加，有时在早期层中接近 1.0。这激励我们使用残差注意力进行有效再利用，同时需要动态门控以适应不同时间步的相似性模式。图 5 (b) 展示了 LaTtE-Flow Couple 中的时间步长条件残差注意力门控，调节了过去层注意力的再利用程度。如图 9 所示，通过所有头部的对比，门控在单个头部内的时间步长上保持稳定，而在不同头部之间有所变化，表明了其专业化。这些结果突出显示了在流匹配生成中，动态、头部特定的残差注意力的有效性。LaTtE-Flow Blend 的结果在附录 D 中。

## 7 结论

在这项工作中，我们提出了 Layerwise Timestep-Expert Flow-based Transformer ( LaTtE-Flow )，这是一种新颖高效的架构，可以在一个单一的多模态模型中统一图像理解和生成。LaTtE-Flow 引入了两个关键的新颖架构创新：分层时间步专家，通过将 transformer 层专用于不同的时间步间隔，从而降低采样复杂度，以及时间步条件残差注意力，它促进了注意力结构在各层间的自适应重用和精炼。广泛的实验评估表明，LaTtE-Flow 不仅在多模态理解和图像生成性能上表现强劲，而且与现有的统一模型相比，推理速度提高了最多 48 倍。

## A LaTtE-Flow 注意模块

图 6 展示了 LaTtE-Flow 注意力模块的架构。我们的框架将从预训练的 VLM 中提取的 3D 旋转位置嵌入 (RoPE) [**?** ] 应用于多模态隐藏状态，并对生成的图像令牌使用一个新的 2D 旋转位置嵌入。我们在生成的图像令牌上采用双向注意力，并允许所有生成的图像令牌关注之前的多模态令牌。

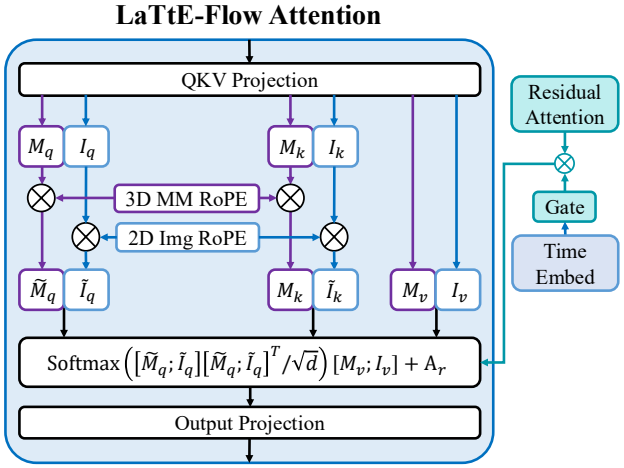



Figure 6: LaTtE-Flow 注意

## B 实现细节

**时间步长分布** 为了启用层次时间步专家，LaTtE-Flow 将模型划分为 $K=4$ 个不重叠的层组，每组包含 $M=7$ 个连续层以获得最终结果。这些组旨在跨越流匹配时间步的不同区间进行操作。在训练期间，我们使用 $T=1000$ 个流匹配步，这些步最初被均匀地划分为四个区间：$[1000.0, 750.25]$ 、$[750.25, 500.50]$ 、$[500.50, 250.75]$ 和 $[250.75, 0]$ 。为了提高区间边界附近的鲁棒性并促进组之间的平滑过渡，我们在训练期间引入了相邻时间步区间之间的 100 步重叠。这种重叠允许边界时间步被多个层组看到，从而提高泛化能力。具体来说，层 1 到 7 被分配到时间步区间 $[1000, 700]$ ，层 8 到 14 覆盖 $[700, 450]$ ，层 15 到 21 在 $[450, 200]$ 上操作，层 22 到 28 处理最后的区间 $[200, 0]$ 。每个组根据方程（1）专门训练其分配的范围，使其能够专注于该流匹配时间步区间特定段的速度预测。

在推理时，我们禁用重叠以保持时间步长区间的严格分区。因此，在每个去噪步骤中，只有对应的专家层组被激活，每个推理步骤只需使用 $M=7$ 层。这与标准的扩散或流匹配模型形成对比，这些模型在每一步都会激活所有的 $L=28$ 层，从而显著提高了生成效率。

我们在八个 H200 上训练所有模型变体大约四天。在训练中，遵循以前的方法，我们使用无分类器指导 [**?** ] ，通过放大有条件生成和无条件生成之间的差异，以指导采样过程，从而提高采样质量，指导尺度为 $> 1$ 。在训练过程中，我们以 10% 的概率随机去掉多模态条件，以促进无条件预测。

为了评估，每个模型基于我们在第 6.2 节的消融研究，对 ImageNet 的 1,000 个类别中的每个类别生成 50 张图像，使用 40 个采样步骤和 5 的无分类器引导（CFG）。我们报告了与 ImageNet 验证集中的 50K 真实图像相比，生成的 50K 图像的 FID 和 Inception Score。遵循之前的惯例 [**?** ] ，我们使用 1,000 张生成图像计算精准率和召回率。所有评分均使用 torch-fidelity [2] 中的标准实现计算。

## C 定性结果

图 7 显示了通过 LaTtE-Flow Couple 采样的 $256 \times 256$ 图像的定性结果。

[2] `https://github.com/toshas/torch-fidelity`

Figure 7: 由在 ImageNet 上训练的 LaTtE-Flow Couple 生成 256 × 256 样本。

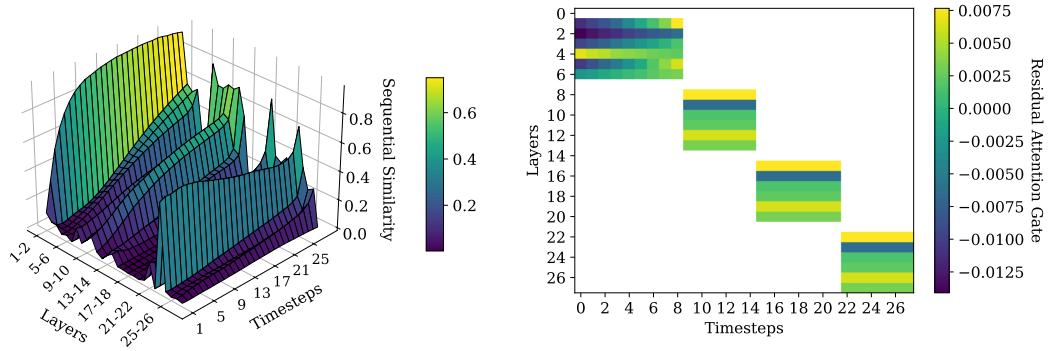
## D 时步条件残差注意力

根据第 6.2 节中的实验设置，我们还对 LaTtE-Flow Blend 变体进行了深入分析。图 8 (a) 显示了相邻层之间的这种顺序相似性如何随着采样时间步的演变而变化。该图显示了在 100 个随机采样的示例中计算的平均相似性。我们观察到，对于大多数相邻层，在早期时间步时顺序相似性相对较低，并随着时间步的推进而逐渐增加，特别是在早期层中，相似性上升并接近 1.0。然而，观测到的相似性模式随着时间步和层而显著变化，激发了对残差注意力流进行时间步条件化门控策略的需求。

在图 8 (b) 中，我们可视化了 LaTtE-Flow Blend 中第 11 号头学习到的残差注意力门控值。这些门由时间步嵌入动态调整，并控制将上一层的残差注意力融入当前层计算的程度。为了进一步理解残差注意力在不同头之间的作用，图 10 展示了 LaTtE-Flow Blend 中所有 12 个头的门控值。我们观察到，特定头的门控在时间步之间相对稳定，但在不同头之间出现了不同的模式。在 LaTtE-Flow Couple 变体中也观察到类似趋势（图 9），其中头特定的门控模式反映出不同的行为。总而言之，这些结果验证了时间步条件化、头特定残差注意力的设计。门控机制实现了对先前注意力的自适应重复利用。

## E 影响声明

这项工作通过引入 LaTtE-Flow 推进了统一多模态建模领域，该架构在单一高效的框架内有效结合了图像理解和生成。通过利用预训练的视觉-语言模型并引入新的架构机制——逐层时间步专家和时间步条件残差注意力，LaTtE-Flow 在显著提高推理速度的同时实现了强大的性能。所提出的模型在学术和实际环境中具有潜在影响，是构建高效、统一的多模态基础模型的可扩展解决方案。它使得在资源受限的环境中，如移动设备或实时应用，可以更高效地部署多模态系统，同时保持高性能。尽管 LaTtE-Flow 改善了性能和效率，但它继承了其预训练视觉-语言基础的偏见，如果没有适当限制，可能会生成误导性或不当的输出。对这些风险的仔细评估和缓解对后续部署至关重要。




(a) Timestep-varying sequential similarity across adjacent transformer layers

(b) Timestep-conditioned residual attention gate for head 11 across transformer layers

Figure 8: 基线混合和 LaTtE-Flow 混合中注意力的可视化。(a) 邻近层之间的顺序相似性随着时间步长的增加而增加，尤其是在早期层。(b) LaTtE-Flow 混合（第 11 个头）中的残差注意力门控显示相同头内跨时间步长的门控值相对一致。

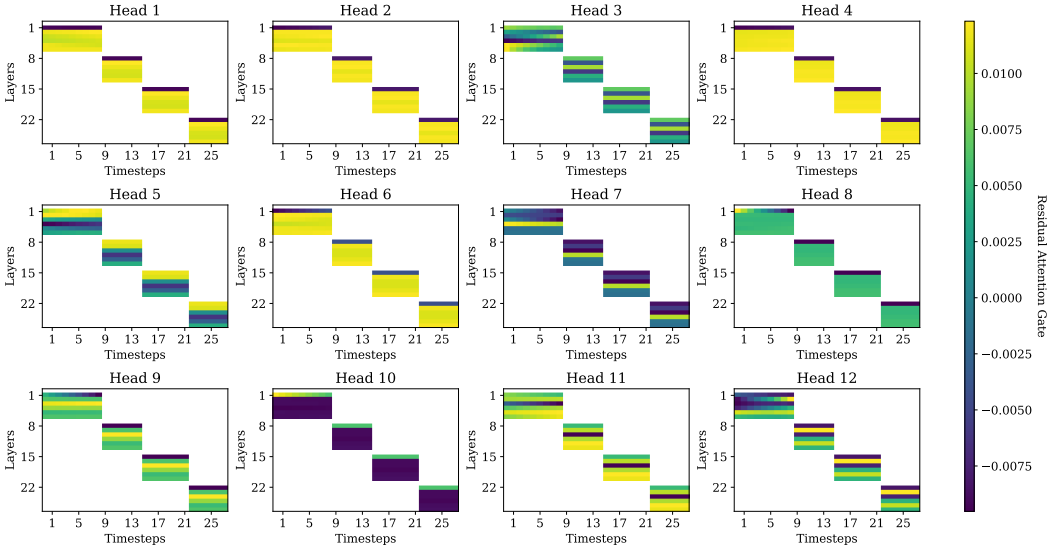



Figure 9: LaTtE-Flow Couple 中跨越 transformer 层的时间步长条件残差注意门控。白色区域表示没有门控值的位置，因为残差注意仅在预定义的层组内应用。值得注意的是，不同的头部表现出不同的门控动态，有些在较早的时间步长上更强调，而另一些则在较后的层中更强烈地调节，这表明头部在残差注意方面具有特定的专门化。

## F 局限性

尽管 LaTtE-Flow 在多模态理解和生成任务中实现了采样效率的显著提高，并取得了良好结果，但仍然存在一些限制。首先，我们的实验仅涉及对 LaTtE-Flow 进行 24 万次优化步骤的训练，这显著少于现有的统一多模态模型。延长训练时间可能会进一步提高模型的表现。第二，尽管我们的重叠区间的均匀时间步分布被证明是有效的，最佳的时间步分布或层划分策略仍然是一个开放的问题。未来的研究应该系统地探索和优化这些时间步划分策略。

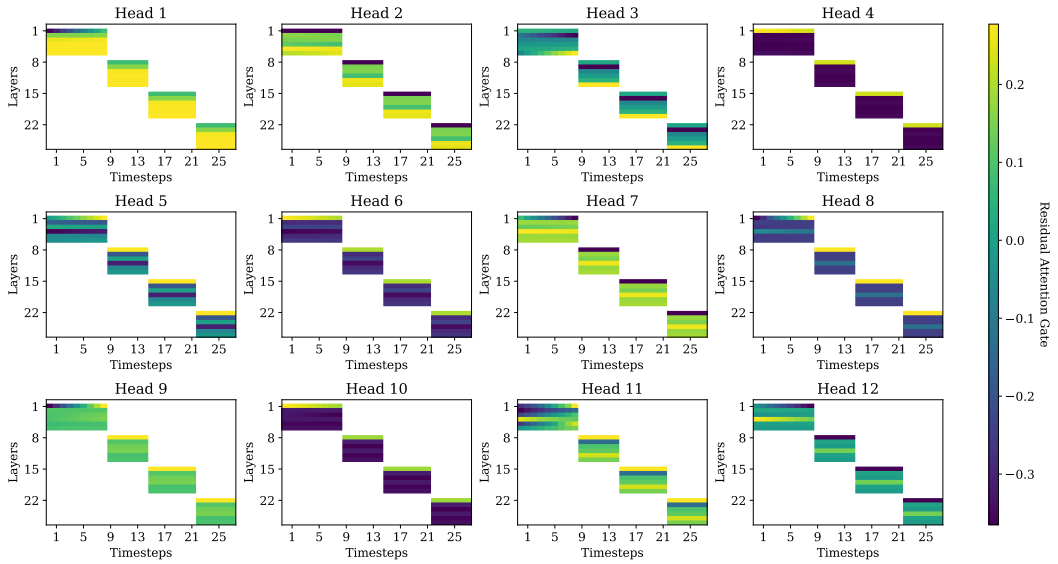



Figure 10: 在 LaTtE-Flow Blend 中，时间步长条件化的残差注意力门控分布于变换器层。白色区域表示没有门控值的位置，因为残差注意力仅在预定义的层组中应用。值得注意的是，不同的头展现出不同的门控动态，有些头重视较早的时间步长，而另一些头在后面的层中更强烈地调节，这表明头在残差注意力中具有特定的专业化。