LooC：用于组合向量量化的高效低维码本

JieLi Kwan-YeeK.Wong KaiHan*香港大学jieli23@hku.hk, kykwong@cs.hku.hk, kaihanx@hku.hk

arXiv:2601.00222v1 [cs.CV] 1 Jan 2026

摘要

向量量化（*VQ*）是一种广泛应用且基础的技术，通过使用码本进行近似，将连续特征向量离散化。随着数据和模型的多样性与复杂性不断提升，业界迫切需要容量更大、同时更加紧凑的*VQ*方法。本文旨在调和这一矛盾，提出一种名为***LooC***的新方法，它利用有效的低维码本本实现组合式向量量化。首先，*LooC*重新构建码向量与特征向量之间的关系，引入一种参数高效码本，显著扩展其解空间。*LooC*不再将码向量与特征向量逐一匹配，而是将码向量视为特征向量中较低维的组合单元，并将它们组合起来，从而得到更紧凑、性能更优的码本。其次，*LooC*采用一种无参数的插值外推机制，在向量量化过程中增强并平滑特征，从而更好地保留细节，并提高特征近似的保真度。*LooC*的设计实现了码本的充分使用，有效利用紧凑码本，同时避免坍塌问题。第三，*LooC*可作为即插即用模块，用于基于*VQ*的不同下游任务中的现有方法。最后，在不同任务、数据集和架构上进行的大量评估表明，*LooC*优于现有*VQ*方法，以显著更小的码本实现了最先进的性能。

1. 引言

向量量化 [11, 32, 41] 是一种广泛使用的技术，可将连续的特征表示转换为有限个离散向量，这些向量称为码本，从而能够高效地分析和处理各种下游应用，例如表征学习 [41, 55, 60], 数据压缩 [33, 44, 58], 聚类 [17, 24, 38, 51], 模式识别 [14, 42, 45], 等。向量量化码本至关重要它可以最大限度地减少输入与匹配码向量之间的失真。这是通过在潜在空间中对特征进行聚类来存储领域信息，从而得到紧凑且信息丰富的表示。

随着数据和模型的多样性与复杂性不断增加，迫切需要改进向量量化方法，使其采用高效且有效、表示能力更强的码本。增大码本大小有望通过提供更丰富的代表性码向量集合，并提高数据表示的精度，来改善性能。然而，收益可能会逐渐趋于平缓，而计算和存储负担却仍在不断增加。此外，较大的码本可能需要更多训练数据才能确保充分表示，这在特定应用中可能成为限制因素。确定最佳码本大小和码向量大小时，需要在准确性、计算复杂度和存储需求之间找到恰当的平衡。通常需要通过实证评估和实验，针对特定应用或数据集确定理想的权衡方案。

已有许多方法被提出用于改进向量量化。例如，[28, 33, 54]将多个码本组合起来，以提高向量量化中码本的容量和表达能力。 [1, 29] 尝试通过后处理方法缩小学习得到的码本，同时避免缩减造成过多信息损失。 [49]引入了一个可学习模块，通过投影由高维码向量构成的学习所得大码本，对低维特征向量进行量化。近期研究 [40, 53] 指出，向量量化中普遍存在码本坍塌问题，即实际上只有一小部分码向量得到有效利用，从而限制了表达能力。对此，最先进的向量量化方法 CVQ-VAE [53], 通过将编码特征用作锚点来更新非活跃码向量，从而解决码本坍塌问题，使更大码本得以有效学习。

乘积量化（PQ） [19] 是一种广受欢迎的高维向量（例如 SIFT 描述符）压缩方法，最初用于向量相似性搜索。它能够以极低的

* 通讯作者。

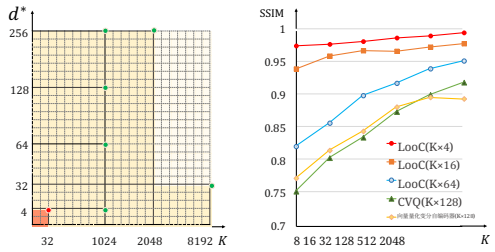


图1。 **码本大小和重建性能。左：**码向量数量 K 和维度 d^* 在码本中的典型配置（绿色圆点）。LooC（红色圆点）凭借显著更小的码本大小 32×4脱颖而出。 **右：**CIFAR10 [25]上的重建结果。与其他最先进方法相比，LooC 使用小得多的码本，却取得了显著更好的性能。

内存和时间成本生成指数级庞大的码本。PQ 提出了将空间分解为低维子空间的笛卡尔积，并分别量化每个子空间的思路。PQ 的最终码本由多个子码本的笛卡尔积构成，每个子码本对应一个特定子空间。然而，对于复杂或高维数据，这种分解可能会生成大量子码本，导致最终码本过于庞大。尽管整体码本大小很大，PQ 仍为不同的量化器使用彼此独立的子码本。这使每个子向量的量化空间受限于单个子码本，可能妨碍 PQ 准确捕捉各个子向量的细粒度细节。为了分别量化子向量，PQ 使用多个不同的量化器。然而，这种做法会显著增加量化器的数量，尤其是在分解粒度更细时。因此，量化器数量上的复杂性增加了网络模型及其实现的复杂程度，也可能显得不够优雅。本文重新审视了将特征空间分解为组合子空间的思路，并提出了一种极其简单且有效的方法 LooC，即组合向量量化的低维码本（**L**ow-dimensional codeb**o**ok for **C**ompositional vector quantization）。该方法引入了若干创新思路，使其不仅拥有极为紧凑的码本，还能取得出色的性能。

首先，我们提出了一种参数高效低维码本（LDC），通过将码向量视为特征向量中的组合元素，重新诠释码向量与特征向量之间的关系。这一视角将 LDC 中的码向量视为描绘完整视觉内容的一组“视觉字符”，而不是将每个对象特征分别与一个“视觉词”相对应。每个向量由共享码本中的多个码向量组成，这些码向量通过统一量化器获得。这种设计大幅提升了码本参数化效率，从而显著减小了码本大小，并明显改善了性能。

其次，为了提高码向量对特征的近似保真度，我们进一步引入一种无需参数的通过插值进行外推机制，在向量量化过程中增强并平滑特征，以更好地保留细节。值得注意的是，在将每个特征向量映射到多个码向量之前，我们首先沿空间维度以 β 倍插值扩展特征图。随后，使用我们的低维码本对插值后的特征进行量化，从而得到外推特征图。最后，通过池化外推后的特征，得到空间维度与原始特征图相同的平滑特征图。

第三，LooC 可作为即插即用模块，无缝集成到现有的基于 VQ 的方法中，以提升其在不同下游任务中的性能，例如图像重建和生成。我们在不同任务、数据集和架构上对 LooC 进行了广泛评估，结果表明，LooC 凭借更加紧凑的码本，在各项任务中均达到了最先进性能。值得注意的是，与最先进方法 CVQ 相比，LooC 使用显著更小的码本和维度更低的码向量，在多个数据集上实现了相当的性能，同时保持 100% 的码本利用率 [52], 。例如，在图像重建任务中，LooC 使用比 CVQ 小 $1024\times$ 的码本，却取得了更好的结果。凭借更加紧凑的码本，LooC 在图像生成方面大幅超越对比方法，生成细节丰富且逼真的图像。

2. 相关工作

向量量化（VQ） [11] 是一项基础研究领域，其起源可追溯至20世纪80年代，至今仍是众多下游应用持续关注的课题。LBG [32] 和 Classified-VQ [37] 等经典方法使用码本来表示输入数据的一组聚类中心。乘积量化（PQ） [19] 和 OPQ [9]提出了一种核心范式：将高维特征分解为低维子向量，并分别进行量化。加性量化（AQ） [2] 通过对多个码本中的码字求和来压缩高维向量，无需进行正交子空间分解，从而降低近似误差、提高搜索/分类准确率，同时保持类似 PQ 的效率。LOPQ [21] 将高维数据划分为多个单元，并针对每个单元局部优化旋转/空间分解以进行残差编码，其开销固定且与数据规模无关，能够降低失真，并在十亿级数据集上实现更快的搜索。随着深度神经网络的兴起，VQ-VAE [41] 将 VQ 引入表征学习，并采用无监督或自监督方法，通过码本学习先验知识。VQ-GAN [7] 将向量量化技术融入 GAN [10, 36],，利用其优势增强生成模型的表征能力，提高样本生成质量。VQ 的有效性使其得到广泛应用

，并应用于不同领域和下游任务中的各种场景。例如，它已用于视觉识别 [20, 24, 51]、图像压缩与重建 [33, 44, 58]、、图像检索 [59]、图像分割[23, 52]、、视觉与音频生成 [5, 8, 12]、、神经辐射场 [30, 43, 56]、、知识蒸馏 [15]、、跨模态检索与翻译 [3, 14, 26]、视觉语言模型 [6], 及其他诸多领域 [17, 26, 38, 42, 45, 47]。在这些方法中，码本充当离散潜在空间的先验分布，从而能够有效地建模和操控数据分布。

在优化向量量化中码本的各个方面，已取得显著进展。首先，为了提升码本的表达能力和表示容量，研究人员探索了不同的方法。AdaCode [33] 为每种图像类别学习一组基码本，并引入权重图来进行自适应图像复原。MoVQ [54] 在解码器中加入两个卷积层，从嵌入向量中学习调制参数，将离散表示转换为经过缩放和平移的值。此外，研究人员还致力于解决码本坍塌问题并提高其使用率。SQ-VAE [40]引入随机反量化和量化技术来应对这一问题。最先进方法 CVQ [53], 提出在线聚类码本学习，通过将编码特征作为锚点来更新不活跃的码向量，从而解决这一问题。此外，研究人员还提出了多种技术来缩小向量量化中码本的规模。 [1]中的一种方法是先对码向量排序，然后利用霍夫曼编码对相邻码向量之间的差值进行编码。

[34] 使用 K-means 聚类算法调整码本大小。HyperHill [29]利用双曲嵌入，通过共现信息增强码本向量，并使用希尔伯特曲线重新排列码本。另一种方法 [49] 通过引入从编码器输出到低维潜变量空间的线性投影，采用更小的码本，但码本数量有所增加。此外，人们还提出了多种复合量化方法。SQ [35]提出了一种迭代量化向量及其残差的方法，将向量表示为一叠代码，称为堆叠量化。RQ [28]以由粗到细的方式递归地使用残差来近似特征向量。TQR [31] 通过结合二值化主干和残差部分，增强了 RQ 在三值神经网络中的效果。

总的来说，这些技术将特征向量视为一个统一实体，并采用维度相同的码本对其进行处理。相比之下，我们将每个特征向量视为由多个更小的码向量拼接而成的组合。在量化过程中将特征向量划分为子向量的有效性，已在视觉检索领域的 PQ [19] 和 SPQ [18] 中得到证明。LooC 将这一设计扩展到图像重建，并进一步改进，使其能够生成表征能力更强的嵌入。与采用多个不同码本的 PQ 不同，LooC 使用一个供所有子向量共享的码本，以增加所有码向量组合的多样性，从而扩展码本的匹配空间。PQ 使用手工设计特征构建乘积量化码本，并施加正交约束。相比之下，LooC 通过图像重建训练，直接学习一个轻量且富有表达力的码本，能够忠实地重建图像。此外，与采用多个不同子量化器的 PQ 不同，LooC 为所有子向量采用统一量化器，从而能够优雅、统一地处理所有特征和码向量。

3. 方法

本节介绍我们提出的量化器 LooC，它为组合式向量量化提供了高效的低维码本。我们首先简要回顾向量量化变分自编码器 [41] 和乘积量化 [19]，作为预备知识。随后，我们介绍 LooC，并阐明 LooC 如何借助低维码本以及通过外推-插值实现的特征增强，取得出色性能。我们还将展示 LooC 如何解决坍塌问题，从而确保充分利用其表示空间。

3.1. 预备知识

用于视觉表示的向量量化变分自编码器。 考虑一个编码器$\Phi_{\text{ENC}}(\cdot)$和一个解码器 $\Phi_{\text{DEC}}(\cdot)$。z 令 $\mathcal{Q}(\cdot)$表示量化算子。于是，$z_{i,j} \in \mathbb{R}^d$ 在每个空间位置$(i,\ j)$处的特征向量会被量化，即用码本 $\mathcal{C}$中最相似的码向量替换它，即，

$$\mathcal{Q}(z_{i,j};\mathcal{C}) = \underset{k\in[K]}{\arg\min} \|z_{i,j} - c(k)\|, \quad \mathcal{C} = \{(k, c(k))\}_{k=1}^{K}, \tag{1}$$

其中，k 是码索引，$c(k) \in \mathbb{R}^{d^*}$ 是 $\mathcal{C}$中对应的码向量。为确保$z_{i,\ j}$ 可以被 $c(k)$替换，它们必须具有相同的维度，即，$d^* = d$。

乘积量化（PQ）。 在乘积量化中，输入向量 $\bar{z}$ 被拆分为 m 个不同的子向量 $\bar{z}_\lambda$、$1 \le \lambda \le m$，每个子向量的维度为$d^* = d/m$，其中 d 是 m的倍数。随后，使用 m 个不同的量化器分别对这些子向量进行量化。因此，给定向量 $\bar{z} = (a_1,\ a_2,\ \cdots,\ a_d)$会按如下方式映射:

$$\begin{aligned} &\underbrace{a_1, \cdots, a_{d^*}}_{\bar{z}_1(a)}, \cdots, \underbrace{a_{d-d^*+1}, \cdots, a_d}_{\bar{z}_m(a)} \\ &\rightarrow q_1(\bar{z}_1(a)), \cdots, q_m(\bar{z}_m(a)), \end{aligned} \tag{2}$$

其中，q_λ 是一种与第λ个子向量相关联的低复杂度量化器。码本 $\mathcal{C}_\lambda$ 可通过子量化器q_λ 与相应的重构值$c_{\lambda,k}$相关联。完整码本 $\mathcal{C}$ 定义为 m 个不同子码本 $\mathcal{C} = \mathcal{C}_1 \times \cdots \times \mathcal{C}_m$的笛卡尔积。

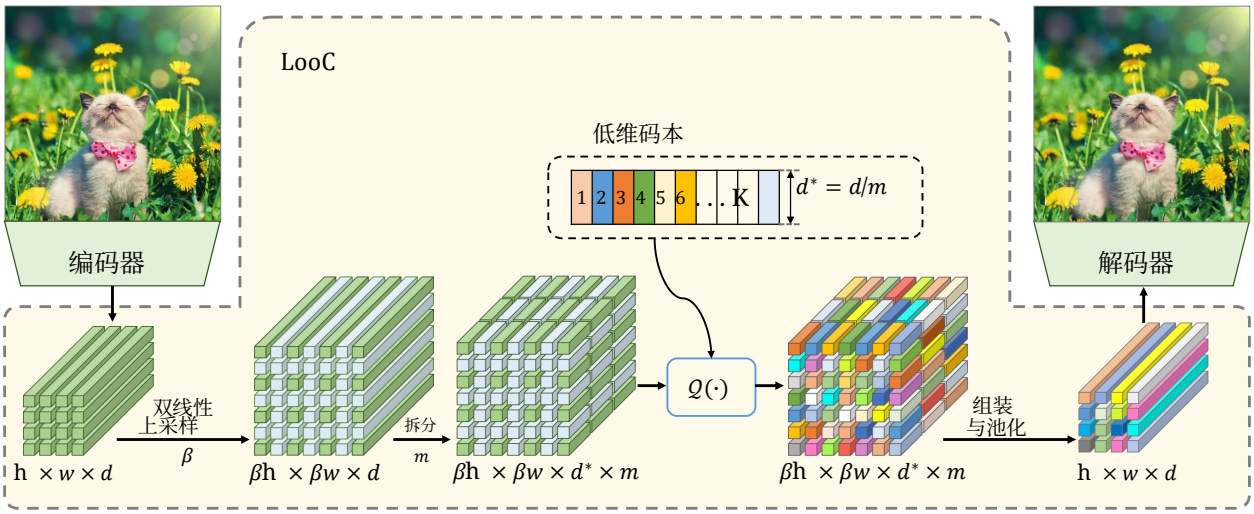


图2. **用于组合向量量化的低维码本（LooC）框架**。 编码器将输入图像转换为连续的潜在特征图 z。随后，使用缩放因子为 β 的双线性插值对 z 进行上采样。同时，将 z 中的每个特征向量划分为 m 个单元，并使用共享码本 $\mathcal{C}$ 进行量化，该码本包含 K 个维度为 $d^* = d/m$ 的码向量。随后，重新组装量化后的单元，并使用平均池化进行平滑，以恢复为 z 的形状。最后，解码器将特征图转换回图像。

3.2. LooC：学习用于组合式向量量化的低维码本

与乘积量化（PQ）采用多个独立子量化器不同，LooC采用一个统一量化器来处理所有子向量。此外，LooC为子向量使用共享码本，无需多个码本，并通过避免笛卡尔积，防止码本像在PQ中那样急剧膨胀。此外，LooC采用插值外推机制来增强并平滑特征，在保留细节的同时确保特征近似的准确性。

低维码本（LDC）。我们不再像乘积量化（PQ）那样将码向量视为原型，而是采用不同的视角，将码向量视为特征向量中的组合单元。类似地，我们不再将码向量视为“视觉词”，而是将其视为“视觉字符”。通过利用特征向量固有的组合特性，LooC 能够用更少的码向量有效捕捉数据的底层结构。具体而言，对于每个特征向量 $z_{i,j} \in \mathbb{R}^d$，我们使用统一量化器 $\mathcal{Q}(\cdot)$同时量化 m 个子向量，从而利用码本 $\mathcal{C} = \{(k, c(k))\}_{k=1}^{K}$中的 m 个码向量，对 $z_{i,j}$进行组合式量化；其中，k 是码索引，$c(k) \in \mathbb{R}^{d^*}$ 是对应的码向量，$d^* = d/m$是每个码向量的维度。m 个码索引可通过以下量化操作获得 $\mathcal{I}_{i,j} = \{\mathcal{Q}(z_{i,j}[(q-1)d^*{:}\ qd^*]; \mathcal{C})\}_{q=1}^{m}$；利用这些索引，$z_{i,j}$ 可以通过拼接 $\mathcal{C}$中与 $\mathcal{I}_{i,j}$ 对应的码向量进行量化。与乘积量化不同，我们为所有子向量使用共享量化器，而非采用多个不同的量化器。此外，我们消除了对多个子码本和笛卡尔积的需求，无需借此构建最终码本；相反，单个紧凑的共享码本便已足够。

我们消除了对多个子码本和笛卡尔积的需求，无需借此构建最终码本；相反，单个紧凑的共享码本便已足够。

特征增强与平滑。 除了低维码本设计之外，LooC 还采用了一种无参数的插值外推机制，在向量量化过程中增强并平滑特征，从而保留细节并确保准确的特征近似。从编码器 $\Phi_{\text{ENC}}(\cdot)$ 获得潜在特征图 z 后，我们首先采用双线性插值，按照缩放因子 β 沿空间维度对特征图进行插值。这一步会生成更大的特征图 $z^{it} \in \mathbb{R}^{\beta h \times \beta w \times (d^* \times m)}$，其中用于向量量化的特征向量数量也有所增加。接下来，对于每个原始特征向量及其插值邻域，我们基于 LDC $\mathcal{C}$ 对其进行量化，从而得到外推特征图 $z^{ex} \in \mathbb{R}^{\beta h \times \beta w \times (d^* \times m)}$。这样可以增强特征的表达能力。最后，我们对每个量化后的原始特征周围的特征进行平均池化，以平滑特征，得到特征图 $\tilde{z} \in \mathbb{R}^{h \times w \times (d^* \times m)}$，随后可由解码器 $\Phi_{\text{DEC}}(\cdot)$ 解码，以重建输入图像。

3.3. 码本紧凑性与指数级表征容量

LDC 的一项关键优势在于，它可以通过组合少量“视觉字符”生成大量“视觉词”。长度为 d 的特征向量被划分为 m 个子向量，每个子向量的长度为 $d^* = d/m$。因此，d^* 的取值决定了 LDC 中码向量的粒度。较小的 d^* 表示对

视觉特征中各个组成部分进行更细粒度的表示。反之，较大的 d^* 则意味着组合单元的粒度更粗。使用包含 K 个码向量的共享码本，在考虑 m 个子向量时，可以实现 K^m 种可能的组合。相比之下，PQ 需要多个独立码本才能实现此类组合。请注意，当 $d^* = d$ 时，LDC 会退化为基础码本。在这种情况下， $m = 1$，这意味着每个特征向量在码本中只有 K 种可能的选择。

.....的增加 m 会逐步提高向量量化重建准确率 [57]。当码向量数量 K 较少时，这一效果尤为显著。如图 1-右所示，我们可以观察到，随着 m 增加，$d^* = d/m$ 减少，而使用 LooC 能有效缓解减少 K所带来的不利影响。详细分析见第 4.2 节。请注意，在极端情况下，当 $d^* = 1$ 时，向量 $z_{i\text{中的}j}$ 每个值都会被单独量化。

通过插值进行外推操作中的参数 β 控制特征增强和平滑程度。在外推过程中，该操作通过整合相邻向量的信息，提高其自身表示向量的容量。我们的实验表明，将 β 设为 2 始终能取得出色的性能。

3.4. LooC 的训练

输入图像 x 经由编码器 $\Phi_{\text{ENC}}(\cdot)$ 转换为特征图 z，再通过 $z = \Phi_{\text{ENC}}(x)$。令 $\hat{z}$ 表示 z 的量化特征图。随后，解码器可利用 $\hat{x} = \Phi_{\text{DEC}}(\hat{z})$ 重建图像。

编码器 $\Phi_{\text{ENC}}(\cdot)$、解码器 $\Phi_{\text{DEC}}(\cdot)$和码本 $\mathcal{C}$通过最小化以下损失进行联合优化：

$$\mathcal{L} = \|x - \hat{x}\|_2^2 + \|\text{sg}[z] - \hat{z}\|_2^2 + \mu \|z - \text{sg}[\hat{z}]\|_2^2, \quad (3)$$

其中，sg 是停止梯度算子，第一项称为重建损失，第二项称为码本损失，第三项称为承诺损失。我们基于向量量化变分自编码器 [41]开发方法。此外，我们遵循 [53]，使用特征作为锚点来更新码向量，以避免码本坍塌。

4. 实验

应用。 为了验证 LooC 的有效性，我们在两项下游任务中开展实验：图像重建 和图像生成。在重建任务中，我们将 LooC 与多种 VQ方法进行比较。为确保公平比较，我们遵循最先进方法 CVQ [53]，将 LooC 集成到 VQ-VAE 的训练框架中，以替换 VQ 模块。我们还重新实现了乘积量化 [19] 中的量化器，并与其他方法一样，采用向量量化变分自编码器网络架构进行训练。随后，我们采用 VQ-GAN [7]架构，在更大的数据集上评估 LooC 的泛化能力。我们的实验使用

方法		$K \times d^* \downarrow$	LPIPS ↓	rFID ↓	结构相似性指数 ↑	峰值信噪比 ↑
向量量化变分自编码器 [41]	MNIST	1024×128	0.0282	3.43	0.9777	26.48
分层向量量化变分自编码器 [44]	MNIST	1024×128	0.0270	3.17	0.9790	26.90
SQ-VAE [40]	MNIST	1024×128	0.0256	3.05	0.9819	27.49
CVQ-VAE [53]	MNIST	1024×128	0.0222	1.80	0.9833	27.87
乘积量化 [19]	MNIST	256×4×#32	0.0120	1.76	0.9933	32.32
LooC	MNIST	32×4	**0.0083**	**1.70**	**0.9961**	**35.15**
LooC	MNIST	256×4	**0.0058**	**1.31**	**0.9976**	**37.58**
向量量化变分自编码器 [41]	CIFAR10	1024×128	0.2504	39.67	0.8595	23.32
分层向量量化变分自编码器 [44]	CIFAR10	1024×128	0.2553	41.08	0.8553	23.22
SQ-VAE [40]	CIFAR10	1024×128	0.2333	37.92	0.8779	24.07
CVQ-VAE [53]	CIFAR10	1024×128	0.1883	24.73	0.8978	24.72
乘积量化 [19]	CIFAR10	256×4×#32	0.0953	27.15	0.9527	28.27
LooC	CIFAR10	32×4	**0.0435**	**24.53**	**0.9805**	**32.22**
LooC	CIFAR10	256×4	**0.0285**	**19.22**	**0.9880**	**34.51**

表 1. **低分辨率数据集上的图像重建结果**，包括 MNIST [27] 和 CIFAR 10 [25]。LooC 以显著减小的码本大小 32×4，取得了优于其他最先进方法的表现；该大小比 1024×128 大多数最先进方法所用的码本小 $1024\times$ 。

方法		$K \times d^* \downarrow$	使用率↑	rFID↓	结构相似性指数↑	峰值信噪比↑
VQGAN [7]	FFHQ	1024×256	42%	4.42	0.6641	22.24
ViT-VQGAN [49]	FFHQ	8192×32	–	3.13	–	–
RQ-VAE [28]	FFHQ	2048×256	–	3.88	0.6700	22.99
MoVQ [54]	FFHQ	1024×64	56%	2.26	0.8212	26.72
SeQ-GAN [13]	FFHQ	1024×256	100%	3.12	–	–
CVQ-VAE [53]	FFHQ	1024×256	100%	2.03	0.8398	26.87
LooC-VAE	FFHQ	256×4	100%	**1.97**	**0.8499**	**27.73**
LooC-VAE	FFHQ	1024×4	100%	**1.37**	**0.9276**	**32.44**
VQGAN [7]	ImageNet	1024×256	44%	7.94	0.5183	19.07
ViT-VQGAN [49]	ImageNet	8192×32	96%	1.28	–	–
RQ-VAE [28]	ImageNet	2048×256	–	1.83	–	–
MoVQ [54]	ImageNet	1024×64	63%	1.12	0.6731	22.42
SeQ-GAN [13]	ImageNet	1024×256	100%	1.99	–	–
CVQ-VAE [53]	ImageNet	1024×256	100%	1.57	0.7115	23.37
LooC-VAE	ImageNet	256×4	100%	1.68	**0.7233**	**23.64**
LooC-VAE	ImageNet	1024×4	100%	**1.01**	**0.7160**	**29.15**

表2. **高分辨率数据集上的图像重建**，包括 FFHQ [22] 和 ImageNet [4]。LooC 的码本规模较小，为 256×4，比大多数最先进方法的码本规模 $256\times$ 小。1024×256

与 CVQ-VAE [53]相同的骨干网络和码本更新方法。在图像生成任务中，我们使用 LDM 框架 [39]，并以 LooC 和其他 VQ方法替换 VQ 模块。

数据集。 我们在各种数据集上检验并验证了我们的方法：MNIST [27]、、CIFAR10 [25], 和 FASHION-MNIST [46]。随后，我们在 ImageNet [4]、、FFHQ [22],和 LSUN [48]等更大型数据集上评估我们的方法。

评价指标。遵循先前工作的做法 [41, 53], ，我们使用多种指标，将重建图像与其原始图像进行比较，包括块级结构相似性指数（SSIM）、特征级学习感知图像块相似度（LPIPS） [50]、、图像级峰值信噪比（PSNR）以及数据集级 Fr´echet Inception 距离（FID） [16]。

4.1. 与既有工作的比较

定量结果。 我们在 MNIST [27] 和 CIFAR10 [25], 等小型数据集，以及 ImageNet[4]和 FFHQ [22]等大型数据集上进行了实验。在



图3. **定性结果**。 使用 VQGAN 重建的图像 [7], CVQ [53], 和 LooC。LooC 通过保留图像细节并还原纹理结构，显著提升了重建质量，如红框所示（建议在 PDF 中放大查看）。

表 1中，我们将 LooC 与多种 VQ 方法进行了比较，包括 VQ-VAE [41],HVQ [44],SQVAE [40],和 CVQ-VAE [53], 中提出的方法，以及重新实现的乘积量化方法 [19]（使用 32 个独立码本）。如表 1 所示，显然，我们的方法在多项评价指标上均优于其他技术。我们方法的一项主要优势是，LooC 的码本大小显著小于先前最先进方法 CVQ-VAE 的码本大小。码本大小仅为 32×4 的 LooC，其性能优于码本大小为 1024×128的 CVQ-VAE。在这种情况下，LooC 的码本大小比 CVQ-VAE $1024\times$ 小。尽管码本大小更小，我们的方法仍显著优于 CVQ-VAE，具体表现为在 MNIST 上的 rFID 分数显著降低至 1.31 ，在 CIFAR10 上降低至 19.22 。尽管使用的码本小得多，我们的方法在各方面也明显优于乘积量化方法。此外，共享码本使我们能够取得优于乘积量化方法的结果。PSNR指标下的结果进一步凸显了我们方法的优势。

接下来，在表2中，我们在具有高分辨率的更具挑战性的大规模数据集 FFHQ [22]和 ImageNet [4]上验证我们方法的有效性。我们的 LooC-VAE 与当前最先进方法进行了比较，包括 CVQ-VAE [53], VQ-GAN [7],ViT-VQGAN [49], RQ-VAE [28],SeQ-GAN [13], 和 MoVQ [54], 在重建任务上的表现。如表2所示，我们的方法在这两个数据集上始终优于以往的最先进方法。与 CVQ-VAE [53], 相比，我们的 LooC-VAE（256×4）在使用更小码本的同时，取得了相当或更好的结果。 $256\times$更小的码本。

同时，如表2所示，我们的方法实现了 100% 的码本利用率。这使得所有码向量都能得到优化，并确保了码本的参数效率。这些实验结果表明，我们的码本在维度和码向量数量方面都具有轻量化的显著优势。

定性结果。 在图3中，我们比较了我们的方法 LooC-VAE 与包括 VQ-GAN [7]和 CVQ-VAE [53]在内的最先进技术的可视化结果。我们的方法在保留图像细节和还原纹理结构方面表现出色，相较于其他方法具有显著优势。查看图3中红框标出的区域即可清楚地看到这一点。值得注意的是，第二列中桌上的水果、第四列中鬣狗的爪子以及最后一列中的建筑物都展现了我们方法的优越性。我们的方法能够高保真度地还原原始图像的外观，而其他方法则可能造成信息丢失或失真。这对于下游任务至关重要。

4.2. LooC 的表达能力

在本节中，我们将探究 LooC 卓越性能和紧凑设计背后的原因。我们主要关注组合式向量量化和通过插值进行外推这两项操作，它们是 LooC 最关键的两个组成部分。

通过细粒度组合增强容量。分析图1-右图显示，在对 LooC 进行初步性能分析时，我们观察到了两个值得注意的现象。

方法	$K \times d^* \downarrow$	MNIST（28×28）/ CIFAR10（32×32）/ FASHION-MNIST（28×28）				
		l_1 损失↓	LPIPS↓	rFID↓	结构相似性指数↑	峰值信噪比↑
向量量化变分自编码器	1024×128	0.0207 / 0.0527 / 0.0377	0.0282 / 0.2504 / 0.0801	3.43 / 39.67 / 12.73	0.9777 / 0.8595 / 0.9140	26.48 / 23.32 / 23.93
CVQ-VAE	1024×128	0.0180 / 0.0448 / 0.0344	0.0222 / 0.1883 / 0.0693	1.80 / 24.73 / 8.85	0.9833 /0.8978 /0.9233	27.87 / 24.72 / 24.66
LooC-VAE	256×4	0.0062 / 0.0144 /0.0103	0.0058 / 0.0285 / 0.0098	1.31 / 19.22 / 6.24	0.9976 /0.9880 /0.9924	37.58 / 34.51 / 35.34
LooC-VAE	256×8	0.0068 / 0.0188 /0.0115	0.0064 / 0.0430 / 0.0114	1.40 / 24.17 / 6.93	0.9972 /0.9809 /0.9907	36.79 / 32.30 / 34.42
LooC-VAE	256×16	0.0097 / 0.0246 / 0.0180	0.0100 / 0.0681 / 0.0211	1.88 / 31.55 / 10.93	0.9949 /0.9679 /0.9796	33.73 / 29.97 / 30.62
LooC-VAE	256×32	0.0118 / 0.0295 / 0.0236	0.0125 / 0.0934 / 0.0325	2.19 / 37.58 / 13.95	0.9928 /0.9551 /0.9660	32.05 / 28.43 / 28.20
LooC-VAE	256×64	0.0138 / 0.0367 / 0.0288	0.0157 / 0.1435 / 0.0476	2.51 / 51.06 / 17.08	0.9903 /0.9293 /0.9497	30.58 / 26.46 / 26.42

表 3. **采用含有 个码向量的码本，在不同组合粒度下的结果** 。随着 d^* 从 64 降至 4（通过将组合粒度 m 从 2 提高至 32 实现），我们的方法在全部三个数据集上均持续提升性能。

方法	$K \times d^* \downarrow$	MNIST(28×28) / CIFAR10(32×32) / FASHION-MNIST(28×28)				
		l_1 损失↓	LPIPS↓	rFID↓	结构相似性指数↑	峰值信噪比↑
向量量化变分自编码器	1024×128	0.0207 / 0.0527 / 0.0377	0.0282 / 0.2504 / 0.0801	3.43 / 39.67 / 12.73	0.9777 / 0.8595 / 0.9140	26.48 / 23.32 / 23.93
CVQ-VAE	1024×128	0.0180 / 0.0448 / 0.0344	0.0222 / 0.1883 / 0.0693	1.80 / 24.73 / 8.85	0.9833 / 0.8978 / 0.9233	27.87 / 24.72 / 24.66
LooC-VAE	256×128	0.0166 / 0.0464 / 0.0344	0.0213 / 0.2144 / 0.0680	3.17 / 64.71 / 21.76	0.9853 / 0.8923 / 0.9256	28.63 / 24.46 / 24.71
LooC-VAE	512×64	0.0122 / 0.0343 / 0.0266	0.0133 / 0.1249 / 0.0410	2.28 / 45.23 / 16.02	0.9923 /0.9392 /0.9566	31.77 / 27.09 / 27.16
LooC-VAE	1024×32	0.0094 / 0.0261 / 0.0197	0.0095 / 0.0735 / 0.0243	1.84 / 33.51 / 11.59	0.9952 /0.9643 /0.9760	34.10 / 29.45 / 29.83
LooC-VAE	2048×16	0.0073 / 0.0205 / 0.0138	0.0070 / 0.0490 / 0.0144	1.51 / 26.29 / 8.29	0.9968 /0.9773 /0.9873	36.14 / 31.55 / 32.84
LooC-VAE	4096×8	0.0056 / 0.0146 / 0.0098	0.0049 / 0.0293 / 0.0091	1.17 / 19.40 / 5.91	0.9980 /0.9878 /0.9929	38.39 / 34.40 / 35.71
LooC-VAE	8192×4	0.0047 / 0.0099 / 0.0083	0.0040 / 0.0154 / 0.0070	0.99 / 12.65 / 4.86	0.9984 /0.9937 /0.9947	39.71 / 37.49 / 37.05

表4. **在固定码本大小 $s = K \times d^*$ 且 K 和 d^*变化的条件下的结果**。组合粒度参数 m 随 d^* 成比例变化，其中 $m = d/d^*$。

首先，随着码向量数量 K 增加，所有方法的性能都得到提升。其次，我们的方法优于其他最先进方法，尤其是在 K 较小时（例如., $K = 32$），优势更加明显。此外，随着 m 增加，我们的方法性能也有所提升，从而使 LooC 中的维度d^* 更低。

这一分析得出了两个关键发现。首先，采用更大的 K 会增大码本大小，显著影响向量量化的准确率。这与现有方法利用更大的 K 来提升向量量化效果这一事实一致，也凸显了缩减码本大小的挑战性。其次，当 K 较小时，我们的 LooC 表现出更强的适应性，同时保持出色的准确率。LooC 通过控制组合式向量量化中的组合粒度参数 m 实现这一点；较大的 m值在 K 较小时表现出更强的适应性。

我们通过实验进一步验证组合式向量量化的重要性，结果如表3所示。在实验中，我们设置 $K = 256$ 并改变 m的取值，使其从2 逐步增加到32。每个潜在特征向量的维度 d 为128，从而使 d^* 从64 降至4。我们的方法在三个不同数据集上均展现出持续的性能提升。例如，在 CIFAR10 上，峰值信噪比从26。46 逐步提升至34。51，而 rFID 则从51。06 降至19。22。同样，其他评价指标也呈现出逐步且稳定的改善。这凸显了我们方法的有效性：该方法利用细粒度组合，提升了码本的容量，从而带来更好的整体性能。

低维紧凑码本。在表 4 中，我们保持码本总大小 $s = K \times d^*$ 不变，同时调整 K 和 d^*。随着 d^* 减小，$m = d/d^*$ 增加，K 也随之增加。这表明，较小的单个码向量能够在保持码本总体大小不变的同时，添加更多码向量。实验结果清楚地表明，随着 d^* 减小（或 m 增大），我们方法的有效性逐步提升，其相较于现有最先进方法的优势也进一步扩大。例如，在 MNIST 数据集上，通过将 LooC($K = 256$, $d^* = 128/m$) 中 m 的值从 1 增加到 32，我们观察到峰值信噪比显著提升，从 28.6 增加到 39.7。三个数据集在各项评价指标上均表现出更好的性能。基于这一事实，我们进一步研究了减少码本中码向量数量的可能性。探索结果如表 5 所示。我们通过保持 $d^* = 4$ 不变，将 m 固定为 32，并仅在实验中改变 K 的值。即使在 K 较小时，我们的方法仍能保持较高的性能。此外，随着 K 增大，我们的方法性能也逐步提升。例如，就 rFID 而言，在 CIFAR10 数据集上，我们的方法在码本 ($K = 32, d^* = 4$) 下取得了与 CVQ ($K = 1024, d^* = 128$) 相近的结果。这相当于将码本大小缩减了 $1024\times$ 倍。在 MNIST 和 FASHION-MNIST 数据集上也观察到了类似现象。

无需参数的插值外推。表。6从两个方面评估插值外推操作的有效性。首先，我们在高分辨率数据集 FFHQ 上将我们的方法与最先进方法 CVQ 进行比较 [53]

方法	$K \times d^* \downarrow$	MNIST（28×28）/ CIFAR10（32×32）/ FASHION-MNIST（28×28）					
		l_1 损失↓		LPIPS↓		rFID↓	结构相似性指数↑	峰值信噪比↑
向量量化变分自编码器	1024×128	0.0207 / 0.0527 / 0.0377	0.0282 / 0.2504 / 0.0801	3.43 / 39.67 / 12.73	0.9777 / 0.8595 / 0.9140	26.48 / 23.32 / 23.93	
CVQ-VAE	1024×128	0.0180 / 0.0448 / 0.0344	0.0222 / 0.1883 / 0.0693	1.80 / 24.73 / 8.85	0.9833 / 0.8978 / 0.9233	27.87 / 24.72 / 24.66	
LooC-VAE	8×4	0.0095 / 0.0220 / 0.0168	0.0097 / 0.0559 / 0.0197	1.86 / 28.35 / 9.92	0.9950 / 0.9740 / 0.9819	33.90 / 30.96 / 31.26	
LooC-VAE	16×4	0.0079 / 0.0211 / 0.0162	0.0080 / 0.0521 / 0.0184	1.61 / 27.13 / 9.89	0.9964 / 0.9763 / 0.9833	35.53 / 31.32 / 31.53	
LooC-VAE	32×4	0.0082 / 0.0189 / 0.0145	0.0083 / 0.0435 / 0.0158	1.70 / 24.53 / 8.73	0.9961 / 0.9805 / 0.9864	35.15 / 32.22 / 32.49	
LooC-VAE	64×4	0.0075 / 0.0178 / 0.0133	0.0072 / 0.0407 / 0.0140	1.54 / 23.70 / 8.05	0.9967 / 0.9825 / 0.9883	36.02 / 32.73 / 33.26	
LooC-VAE	128×4	0.0068 / 0.0158 / 0.0114	0.0065 / 0.0332 / 0.0112	1.43 / 21.15 / 6.90	0.9972 / 0.9859 / 0.9910	36.77 / 33.70 / 34.59	
LooC-VAE	256×4	0.0062 / 0.0144 / 0.0103	0.0058 / 0.0285 / 0.0098	1.31 / 19.22 / 6.24	0.9976 / 0.9880 / 0.9924	37.58 / 34.51 / 35.34	
LooC-VAE	512×4	0.0055 / 0.0137 / 0.0092	0.0051 / 0.0258 / 0.0083	1.18 / 18.14 / 5.50	0.9980 / 0.9891 / 0.9937	38.40 / 34.92 / 36.24	
LooC-VAE	1024×4	0.0051 / 0.0116 / 0.0083	0.0045 / 0.0192 / 0.0073	1.10 / 14.10 / 5.07	0.9982 / 0.9918 / 0.9946	38.97 / 36.38 / 37.10	

表5。**固定 时，不同** $K \in \{8, 16, \cdots, 1024\}$ 码向量的结果。



(a) LSUN卧室数据集 (b) LSUN教堂数据集 (c) ImageNet

图4。**U LSUN 上的无条件图像生成** [48] 以及 **ImageNet 上的类别条件图像生成** [4].

方法	β	$K \times d^* \downarrow$ LPIPS↓ rFID↓ 结构相似性指数↑ 峰值信噪比↑
向量量化变分自编码器	-	1024×256 0.1175 4.42 0.6641 22.24
CVQ-VAE	-	1024×256 0.0533 2.03 0.8398 26.87
LooC-VAE	2	1024×256 0.0532 1.83 0.8627 27.02
LooC-VAE 1		$256 \times$ 4 0.0528 2.27 0.8571 27.19
LooC-VAE 2		$256 \times$ 4 0.0501 1.97 0.8499 27.73
LooC-VAE 3		$256 \times$ 4 0.0478 1.89 0.8523 27.00
LooC-VAE 4		$256 \times$ 4 0.0490 2.01 0.8510 26.88

表 6. **FFHQ** 上各种设置下的结果 [22]。

。在此次比较中，我们设置 $m = 1$，使我们的 LDC 等同于标准码本。其他参数，如 $K = 1024$和 $d^* = d = 256$，均与 CVQ 保持一致。随后，我们使用 $\beta = 2$应用所提出的插值外推机制。结果表明，我们提出的机制显著提高了向量量化的准确性。例如，与 CVQ 相比，我们的方法将 rFID 分数从 2.03 降至 1.83，同时将峰值信噪比从 26.87 提高至 27.02。

随后，我们使用 LooC，并结合 $K = 256$ 和 $d^* = 4$，设置不同的 $\beta \in \{1, 2, 3, 4\}$。相应的实验结果见表6。我们发现，采用带有 $\beta = 2$ 的双线性差分可以带来显著改进。随着 β 的增加，向量的空间相关性减弱，改进效果逐渐下降。因此，我们选择 $\beta = 2$，从而可以将这一机制与其他 VQ 技术相结合，以进一步提高准确率。

4.3. 用于图像生成的即插即用模块

我们采用量化器 LooC，按照先进的 CVQ [53] 方法评估其在图像生成中的有效性。我们将 VQGAN 中的 VQ 模块 [7] 替换为 LooC，再将其应用于 LDM [39] 方法，并在 LSUN [48]和 ImageNet [4]上进行训练。LSUN 上的无条件图像生成结果以及 ImageNet 上的类别条件图像生成结果如图 4所示。我们的方法能够生成细节丰富、逼真的图像，证明该方法是一个实用的即插即用模块，适用于各种下游任务。

5. 结论

我们提出了 LooC，这是一种用于组合向量量化的高效量化器，采用低维码本。LooC 不仅性能出色，而且码本极为紧凑。通过将码向量视为特征向量中的组合单元，LooC 在不牺牲性能的前提下实现了更精简的码本。此外，LooC 还采用插值外推机制来增强并平滑特征，从而确保特征近似准确，同时保留细节。我们的量化器方案简单而有效，可无缝集成到现有的表征学习架构中。

致谢

本研究获中国国家自然科学基金（项目编号：62306251）、香港研究资助局优配研究金（项目编号：17211024）以及香港大学基础研究种子基金资助。

参考文献

[1] Sahar K Ahmed. 图像向量量化中缩减码本规模的新方法。AL-Rafidain计算机科学与数学期刊，2 1 53 62, 2005 1, 3 ( ): –。[2] Artem Babenko 和 Victor Lempitsky。用于极限向量压缩的加性量化。载于 IEEE 计算机视觉与模式识别会议论文集，第931–938 页，2014年。2[3] Yue Cao、Mingsheng Long、Jianmin Wang 和 Shichen Liu。用于高效跨模态检索的集体深度量化。载于 AAAI人工智能会议论文集，2017年。3[4] Jia Deng、Wei Dong、Richard Socher、Li-Jia Li、Kai Li 和 Li Fei-Fei。ImageNet：大型层次化图像数据库。载于 2009 年 IEEE计算机视觉与模式识别会议论文集，第 248–255 页。IEEE，2009年。5、6、8、3、4[5] Prafulla Dhariwal、Heewoo Jun、Christine Payne、Jong Wook Kim、Alec Radford 和 Ilya Sutskever。Jukebox：一种音乐生成模型。arXiv预印本 arXiv:2005.00341，2020年。3[6] Xiaoyi Dong、Jianmin Bao、Ting Zhang、Dongdong Chen、Weiming Zhang、Lu Yuan、Dong Chen、Fang Wen、Nenghai Yu 和 Baining Guo。PECO：用于视觉Transformer的 BERT 预训练感知码本。载于 AAAI人工智能会议论文集，第 552 560 页，2023年 3 – 。[7] Patrick Esser、Robin Rombach 和 Bjorn Ommer。驾驭 Transformer 以进行高分辨率图像合成。载于 IEEE/CVF计算机视觉与模式识别会议论文集，第 12873–12883 页，2021年。2、5、6、8、3、4[8] Rinon Gal、Yuval Alaluf、Yuval Atzmon、Or Patashnik、Amit H Bermano、Gal Chechik 和 Daniel Cohen-Or。一张图像胜过千言万语：使用文本反演实现文本到图像生成的个性化。载于国际学习表征会议，2023 3 。[9] Tiezheng Ge、Kaiming He、Qifa Ke 和 Jian Sun。优化乘积量化。IEEE 模式分析与机器智能汇刊，36(4):744–755，2013年。2[10] Ian Goodfellow、Jean Pouget-Abadie、Mehdi Mirza、Bing Xu、David Warde-Farley、Sherjil Ozair、Aaron Courville 和 Yoshua Bengio。生成对抗网络。神经信息处理系统进展，27, 2014 2 。[11] Robert Gray。向量量化。IEEE ASSP杂志，1 (2):4–29，1984年。1、2[12] Shuyang Gu、Dong Chen、Jianmin Bao、Fang Wen、Bo Zhang、Dongdong Chen、Lu Yuan 和 Baining Guo。用于文本到图像合成的向量量化扩散模型。载于 IEEE/CVF计算机视觉与模式识别会议论文集，第 10696–10706 页，2022年。3[13] Yuchao Gu、Xintao Wang、Yixiao Ge、Ying Shan、Xiaohu Qie 和 Mike Zheng Shou。重新思考用于图像合成的向量量化标记器的目标。arXiv 预印本 arXiv:2212.03185，2022年。5, 6, 3[14] Haohan Guo、Fenglong Xie、Xixin Wu、Frank K Soong 和 Helen MengFellow。Msmc-tts：基于多阶段多码本 VQ-VAE 的神经文本转语音。IEEE/ACM 音频、语音与语言处理汇刊，2023年。1, 3[15] Liyong Guo、Xiaoyu Yang、Quandong Wang、Yuxiang Kong、Zengwei Yao、Fan Cui、Fangjun Kuang、Wei Kang、Long Lin、Mingshuang Luo 等。预测用于知识蒸馏的多码本向量量化索引。载于 ICASSP 2023-2023 IEEE声学、语音与信号处理国际会议论文集，第1–5页。IEEE，2023年。3[16] Martin Heusel、Hubert Ramsauer、Thomas Unterthiner、Bernhard Nessler 和 Sepp Hochreiter。采用双时间尺度更新规则训练的生成对抗网络收敛至局部纳什均衡。神经信息处理系统进展，第30卷，2017年。5[17] Qiyu Hu、Guangyi Zhang、Zhijin Qin、Yunlong Cai、Guanding Yu 和 Geoffrey Ye Li。利用掩码 VQ-VAE 码本实现稳健语义通信。IEEE 无线通信汇刊，2023年。1, 3[18] Young Kyun Jang 和 Nam Ik Cho。用于深度无监督图像检索的自监督乘积量化。载于 IEEE/CVF 国际计算机视觉会议论文集，第12085–12094页，2021年。3[19] Herve Jegou、Matthijs Douze 和 Cordelia Schmid。用于最近邻搜索的乘积量化。IEEE 模式分析与机器智能汇刊，33(1)：117–128，2010年。1, 2, 3, 5, 6, 4 ( ): – .[20] Frederic Jurie 和 Bill Triggs。为视觉识别创建高效码本。载于第十届 IEEE 国际计算机视觉会议论文集，第604–610页。IEEE，2005年。3[21] Yannis Kalantidis 和 Yannis Avrithis。用于近似最近邻搜索的局部优化乘积量化。载于 IEEE 计算机视觉与模式识别会议论文集，第2321–2328页，2014年。2[22] Tero Karras、Samuli Laine 和 Timo Aila。用于生成对抗网络的基于风格的生成器架构。载于 IEEE/CVF计算机视觉与模式识别会议论文集，第4401–4410页，2019年。5, 6, 8, 1, 3, 4[23] HB Kekre、Tanuja K Sarode 和 Bhakti Raul。基于能量排序概念的向量量化技术用于彩色图像分割。International Journal of Computing Science and Communication Technologies，1:2，2009年。3[24] Muhammad Hassan Khan、Muhammad Shahid Farid 和 Marcin Grzegorzek。基于码本的步态识别特征融合综合研究。信息融合，92：216–230，2023年。1, 3[25] Alex Krizhevsky、Geoffrey Hinton 等。学习微小图像的多层特征。2009年。2, 5, 3[26] Zhibin Lan、Jiawei Yu、Xiang Li、Wen Zhang、Jian Luan、Bin Wang、Degen Huang 和 Jinsong Su。探索更好的

多模态码本文本图像翻译。载于计算语言学协会第61届年会论文集，第3479–3491页，2023年。3[27] Yann LeCun、L´eon Bottou、Yoshua Bengio 和 Patrick Haffner。应用于文档识别的基于梯度的学习。IEEE论文集，86(11)：2278–2324，1998年。5, 3[28] Doyup Lee、Chiheon Kim、Saehoon Kim、Minsu Cho 和 Wook-Shin Han。利用残差量化进行自回归图像生成。载于IEEE/CVF计算机视觉与模式识别会议论文集，第11523–11532页，2022年。1, 3, 5, 6[29] Lei Li、Tingting Liu、Chengyu Wang、Minghui Qiu、Cen Chen、Ming Gao 和 Aoying Zhou。无需重新训练即可调整向量量化码本大小。多媒体系统，第1–14页，2023年。1, 3[30] Lingzhi Li、Zhongshu Wang、Zhen Shen、Li Shen 和 Ping Tan。利用神经码本构建紧凑实时辐射场。arXiv 预印本 arXiv:2305.18163，2023年。3[31] Yue Li、Wenrui Ding、Chunlei Liu、Baochang Zhang 和 Guodong Guo。TRQ：采用残差量化的三值神经网络。载于 AAAI人工智能会议论文集，第8538–8546页，2021年。3[32] Yoseph Linde、Andres Buzo 和 Robert Gray。一种矢量量化器设计算法。IEEE通信汇刊，28(1)：84–95，1980年。1, 2[33] Kechun Liu、Yitong Jiang、Inchang Choi 和 Jinwei Gu。学习图像自适应码本，以实现类别无关图像复原。arXiv 预印本 arXiv:2306.06513，2023年。1, 3[34] Qi Mao、Tinghan Yang、Yinuo Zhang、Shuyin Pan、Meng Wang、Shiqi Wang 和 Siwei Ma。使用微调后的 VQGAN 模型进行极致图像压缩。arXiv 预印本 arXiv:2307.08265，2023年。3[35] Julieta Martinez、Holger H Hoos 和 James J Little。用于组合式向量压缩的堆叠量化器。arXiv 预印本 arXiv:1411.2173，2014年。3[36] Alec Radford、Luke Metz 和 Soumith Chintala。利用深度卷积生成对抗网络进行无监督表征学习。载于国际学习表征会议，2016年。2[37] B. Ramamurthi 和 A. Gersho。图像的分类矢量量化。IEEE通信汇刊，34(11)：1105–1115，1986年。2[38] Arturo Ribes、Senshan Ji、Arnau Ramisa 和 Ramon Lopez de Mantaras。用于码本构建的自监督聚类：一种应用于目标定位的方法。载于《人工智能研究与发展——加泰罗尼亚人工智能协会第14届国际会议论文集》，第208–217页。IOS Press，2011年。1, 3[39] Robin Rombach、Andreas Blattmann、Dominik Lorenz、Patrick Esser 和 Bj¨orn Ommer。利用潜在扩散模型进行高分辨率图像合成。载于 IEEE/CVF计算机视觉与模式识别会议论文集，第10684–10695页，2022年。5, 8, 2[40] Yuhta Takida、Takashi Shibuya、Weihsiang Liao、Chieh-Hsin Lai、Junki Ohmura、Toshimitsu Uesaka、Naoki Murata、Shusuke Takahashi、Toshiyuki Kumakura 和 Yuki Mitsufuji。SQ-VAE：通过自退火随机量化对离散表示进行变分贝叶斯推断。载于《国际机器学习会议》，第 20987–21012 页。PMLR，2022年。1, 3, 5, 6[41] Aaron Van Den Oord、Oriol Vinyals 等。神经离散表示学习。载于《神经信息处理系统进展》，第 30 卷，2017年。1, 2, 3, 5, 6[42] Vivek Venugopal、Surbhi Pillai 和 Suresh Sundaram。用于在线作者识别的层次化码本描述符方法。载于《2018年第16届国际手写识别前沿会议（ICFHR）》，第 398–403 页。IEEE，2018年。1, 3[43] Matthew Wallingford、Aditya Kusupati、Alex Fang、Vivek Ramanujan、Aniruddha Kembhavi、Roozbeh Mottaghi 和 Ali Farhadi。神经辐射场码本。载于《国际学习表征会议》，2023年。3[44] Will Williams、Sam Ringer、Tom Ash、David MacLeod、Jamie Dougherty 和 John Hughes。层次化量化自编码器。载于《神经信息处理系统进展》，第 33 卷：4524–4535，2020年。1, 3, 5, 6[45] Rui-Qi Wu、Zheng-Peng Duan、Chun-Le Guo、Zhi Chai 和 Chongyi Li。Ridcp：通过高质量码本先验重振真实图像去雾。载于《IEEE/CVF计算机视觉与模式识别会议论文集》，第 22282–22291 页，2023年。1, 3[46] Han Xiao、Kashif Rasul 和 Roland Vollgraf。Fashion-MNIST：一个用于机器学习算法基准测试的新型图像数据集，2017年。5, 3[47] Peng Ye 和 David Doermann。使用视觉码本进行无参考图像质量评价。IEEE图像处理汇刊，21(7)：3129–3138，2012年。3[48] Fisher Yu、Yinda Zhang、Shuran Song、Ari Seff 和 Jianxiong Xiao。LSUN：利用人在回路中的深度学习构建大规模图像数据集。arXiv 预印本 arXiv:1506.03365，2015年。5, 8, 4[49] Jiahui Yu、Xin Li、Jing Yu Koh、Han Zhang、Ruoming Pang、James Qin、Alexander Ku、Yuanzhong Xu、Jason Baldridge 和 Yonghui Wu。改进 VQGAN 的向量量化图像建模。载于《国际学习表征会议》，2022年。1, 3, 5, 6[50] Richard Zhang、Phillip Isola、Alexei A Efros、Eli Shechtman 和 Oliver Wang。深层特征作为感知度量的惊人效力。载于《IEEE计算机视觉与模式识别会议论文集》，第 586–595 页，2018年。5[51] Wei Zhang、Akshat Surve、Xiaoli Fern 和 Thomas Dietterich。学习非冗余码本以分类复杂对象。载于《第26届国际机器学习年会论文集》，第 1241–1248 页，2009年。1, 3 – .[52] Tianchen Zhao、Niansong Zhang、Xuefei Ning、He Wang、Li Yi 和 Yu Wang。Code DVTR：具有几何引导的基于码本的稀疏体素变换器。载于《IEEE/CVF计算机视觉与模式识别会议论文集》，第 1435–1444 页，2022年。2, 3

[53] Chuanxia Zheng 和 Andrea Vedaldi。在线聚类码本。载于《IEEE/CVF国际计算机视觉会议论文集》，第 22798–22807 页，2023年。1, 3, 5, 6, 7, 8, 2[54] Chuanxia Zheng、Tung-Long Vuong、Jianfei Cai 和 Dinh Phung。MoVQ：通过调制量化向量实现高保真图像生成。《神经信息处理系统进展》，第 35 卷：23412–23425，2022年。1, 3, 5, 6[55] Sipeng Zheng、Bohan Zhou、Yicheng Feng、Ye Wang 和 Zongqing Lu。Unicode：为多模态大语言模型学习统一码本。载于《欧洲计算机视觉会议》，第 426–443 页。施普林格，2025年。1[56] Hongliang Zhong、Jingbo Zhang 和 Jing Liao。Vq-nerf：使用向量量化进行神经反射分解与编辑。arXiv 预印本 arXiv:2310.11864，2023年。3[57] Lei Zhu、Fangyun Wei、Yanye Lu 和 Dong Chen。将 VQ-GAN 的码本大小扩展至 100,000，同时实现 99% 的利用率。载于《神经信息处理系统进展》，第 12612–12635 页。Curran Associates, Inc.，2024年。5[58] Xiaosu Zhu、Jingkuan Song、Lianli Gao、Feng Zheng 和 Heng Tao Shen。用于高效神经图像压缩的统一多元高斯混合。载于《IEEE/CVF计算机视觉与模式识别会议论文集》，第 17612–17621 页，2022年。1, 3[59] Xiaosu Zhu、Jingkuan Song、Lianli Gao、Xiaoyan Gu 和 Heng Tao Shen。重新审视多码本量化。IEEE图像处理汇刊，2023年。3[60] Wenbin Zou、Hongxia Gao、Tian Ye、Liang Chen、Weipeng Yang、Shasha Huang、Hongsheng Chen 和 Sixiang Chen。Vqcnir：利用向量量化码本实现更清晰的夜间图像复原。载于《AAAI人工智能会议论文集》，第 7873–7881 页，2024年。1

LooC：用于组合向量量化的高效低维码本

— 补充材料 —

1. 存储与计算效率

第 3.3 节表明，与容量相同的普通向量量化相比，我们的码本理论上只需 $K' = K^{\frac{1}{m}}$ 个码向量，而普通向量量化需要 K 个码向量。特别是，当 $m > 1$ 时，K' 显著小于 K。在此，我们深入分析 LooC 的存储和计算成本。结果表明，与传统向量量化相比，LooC 能以更少的空间和计算开销提供更大的容量，因此是效率更高的替代方案。

1.1. 存储效率

码本存储成本。 与传统向量量化相比，LooC 所需的码向量数量 (K') 更少，维度 (d) 更低，因此存储成本更低。使用包含 K 个维度为 d 的码向量的码本对特征图 $z \in \mathbb{R}^{h\times w\times d}$执行向量量化时，码本需要存储 $K \times d$ 个值，通常采用 32 位浮点格式。因此，总共需要 $S_{\text{codebook}} = 32 \times K \times d$ 位存储空间。而在我们的 LooC 中，需要存储$K' \times d^* = K' \times d/m$ 个值。理论上，当 $K' = K^{\frac{1}{m}}$时，只需 $S'_{\text{codebook}} = 32 \times K^{\frac{1}{m}} \times d/m$ 位存储空间。随着 K' 值增大，我们的方法能够在消耗更少存储空间的同时，实现比传统方法更大的容量。

索引存储成本。 对包含 K 个码向量的码本执行向量量化操作时，需要使用 $h\times w$ 个索引来存储量化匹配的对应关系。每个索引需要 $_2 K$ 位存储空间。因此，总共需要 $S_{\text{index}} = h \times w \times \log_2 K$ 的存储空间。在我们的 LooC 方法中，由于每个特征向量被划分为 m 个分量单元，因此需要 $h \times w \times m$ 个索引，占用 $S'_{\text{index}} = h \times w \times m \times \log_2 K'$ 位存储空间。值得注意的是，$K' < K$。当 $K' = K^{\frac{1}{m}}$ 时，S'_{index} 等价于 S_{index}。

1.2. 计算效率

为了确定码本中的 K 个码向量与特征表示$z \in \mathbb{R}^{h\times w\times d}$之间最相似的匹配项，总共需要进行 $h\times w\times K$ 次相似度计算。这里，我们采用广泛使用的余弦相似度进行匹配。因此，在传统的向量量化中，需要进行 $h \times w \times K \times d$ 次乘法运算，而加法运算则省略不计。然而，在我们的 LooC 方法中，对于包含 K' 个码向量的情况，特征表示 z 被分解为 m 个分段，因此需要进行 $h\times w\times m\times K'\times d^*$ 次乘法运算。这里，$d^* = d/m$。因此，它

需要 $h \times w \times m \times K' \times (d/m) = h \times w \times K' \times d$ 乘法运算。由于 K' 远小于 K，因此我们方法的计算成本也远低于传统的向量量化。

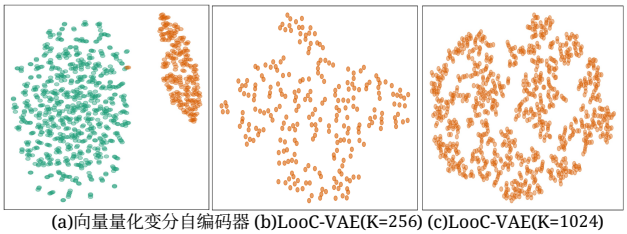
2. 码本使用率

2.1. 码向量的使用率

总体使用率。 在表2中，我们的方法展现出卓越的表现，码本总体使用率达到 100%。本节中，我们使用 t-SNE 进行可视化，以深入了解向量量化变分自编码器和我们提出的 LooC-VAE 学习到的码本。图 5(a) 展示了向量量化变分自编码器的码本可视化结果，可以明显看出，许多码向量未被使用。然而，对于我们的 LooC-VAE，如图 5(b) 和 (c) 所示，当码本大小分别为 1024 和 256 时，使用率均达到 100%。这证明了我们的方法能够有效地充分利用可用的码本容量。

不使用 CVQ 更新方法时的码本使用率 为了进一步研究这一点，我们通过移除 LooC 中的 CVQ 更新策略并统计码本使用率来进行实验。我们在大型数据集 FFHQ [22]上计算使用率。结果如表 7所示。我们的研究结果表明，在 100% 时，码本使用率在 $K = 256$ 和 $K = 1024$均为 100%。这清楚地表明，LooC 在码本使用率方面的优势并非仅仅源自 CVQ 更新策略。

分段使用率。 在组合式向量量化中，我们将特征图划分为m 个分段，本研究分析了每个分段的码本使用率。我们在分析中采用 CIFAR10 作为数据集，并考虑以下情况：



(a)向量量化变分自编码器 (b)LooC-VAE(K=256) (c)LooC-VAE(K=1024)

图 5. **使用 t-SNE 可视化码本**针对训练的模型
在 CIFAR10 上训练并在验证集上评估。向量量化变分自编码器有未使用的码向量（绿色点），使用率仅为 24.12%。LooC 在 $K = 256$ 和 $K = 1024$ 处均实现了 100% 的使用率。

方法	数据集	$K \times d^* \downarrow$	使用率
LooC-VAE	FFHQ	256×4	100%
LooC-VAE		1024×4	100%

表 7. **码向量的使用率**我们的 LooC（未采用 CVQ 的码本更新策略）。

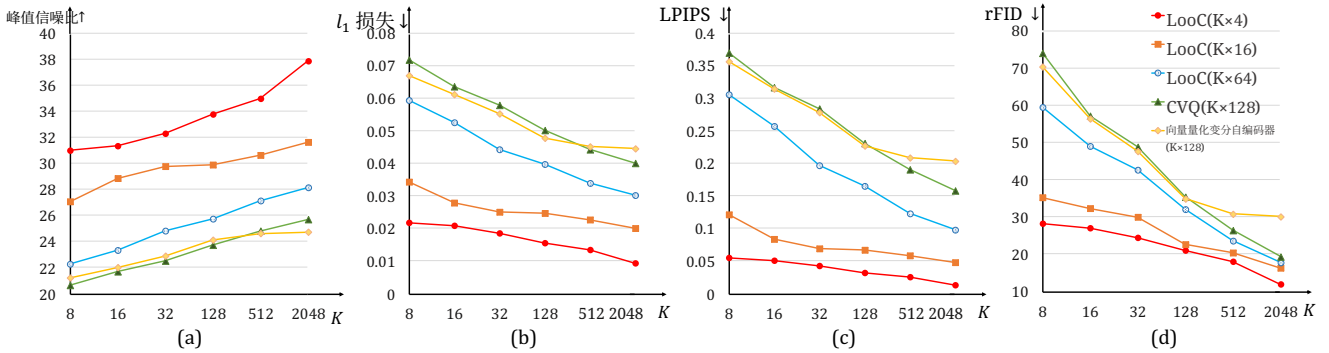


图6。 **LooC 与其他最先进方法在 CIFAR**10 [25] 上采用不同数量码向量时的重建结果表明，LooC 使用更小的码本便能取得更优性能，展现出其灵活性和效率。降低码向量维度 d^*，即，增大 $m = d/d^*$ 的值，会使 LooC 实现更精细的组合量化并提升性能，尤其是在 K 较小时。

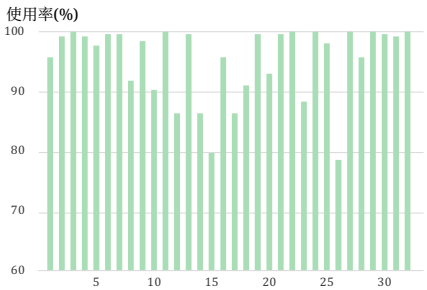


图7. **使用率**各分段的码向量使用情况。该实验在 CIFAR10 数据集上进行，采用 $K = 256$ 和$m = 32$。较高的码向量分段使用率也表明，不同分段之间对码向量共享有相当大的需求。

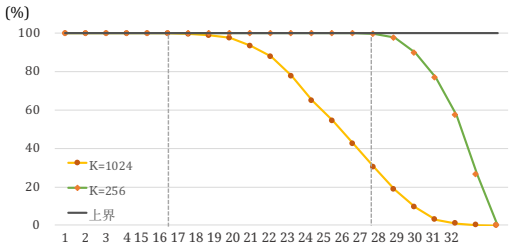


图8. **(a)** 当 $K = 256$ 时，至少 26 个分段中的码向量会被共享。**(b)** 当 $K = 1024$ 时，至少 16 个分段中的码向量会被共享。较小的 K 值会带来更高的码向量共享率。较大的 K 值则降低了对高共享率的需求。

$K = 256$ 和 $m = 32$，如表 3所示。当综合考虑全部 32 个分段的码本使用率时，总体使用率仍为 100%。图 7 展示了各分段的码向量使用率。横轴表示分段索引，纵轴表示各分段的码本使用率。进一步考察各个分段后，我们发现，除 6 个分段外，其余分段的使用率均超过 90%。

2.2. 码向量共享

在先前的使用率分析中，每个分段较高的码本使用率表明，不同分段之间存在显著的码向量共享。较高的使用率与较高的共享程度直接相关。为进一步探究不同 K 值对码向量共享的影响，我们使用$K = 256$和 $K = 1024$，在 CI-FAR10 数据集上分析 $m = 32$ 个分段。我们分析至少被 $n \in \{1, 2, \cdots, 32\}$ 个分段共享的码向量所占的比例。图8(a) 显示，当 $K = 256$时，所有码向量都至少被 32 个分段中的 26 个共享，其中大多数至少被 30 个分段共享。图 8(b) 表明，当 $K = 1024$时，所有码向量都至少被 16 个分段共享，其中大多数至少被 20 个分段共享。较小的 K 值会提高不同分段之间的码向量共享率。这一发现解释了为什么我们的方法在 K较小时表现更好。相反，较大的 K 值则降低了追求高共享率的必要性。

3. 实现细节

在图像重建任务中，我们将 LooC 与各种 VQ 模块进行比较。为确保公平比较，我们参照 CVQ [53]，通过替换 VQ 模块将 LooC 集成到 VQ-VAE 的网络结构中。我们还重新实现了乘积量化 [19]中的量化器，并与其他方法一样，采用 VQ-VAE 结构进行训练。之后，我们使用 VQ-GAN [7] 架构，在更大型的数据集上评估 LooC 方法的泛化能力。实验采用与 CVQ-VAE [53],相同的编码器和解码器，两者均基于卷积神经网络。我们的模型使用单张 RTX3090 GPU 进行训练，优化器、批量大小、学习率和训练轮数的设置均与 CVQ 一致

对于图像生成任务，我们采用 LDM 框架 [39] ，并用 LooC 和其他对比方法替换向量量化模块。与基线 LDM 类似，我们

方法		$K \times d^* \downarrow$	l_1 损失 ↓	LPIPS ↓	rFID ↓	结构相似性指数 ↑	峰值信噪比 ↑
向量量化变分自编码器 [41]	MNIST	1024×128	0.0207	0.0282	3.43	0.9777	26.48
分层向量量化变分自编码器 [44]	MNIST	1024×128	0.0202	0.0270	3.17	0.9790	26.90
SQ-VAE [40]	MNIST	1024×128	0.0197	0.0256	3.05	0.9819	27.49
CVQ-VAE [53]	MNIST	1024×128	0.0180	0.0222	1.80	0.9833	27.87
LooC(32×4)	MNIST	32×4	0.0082	**0.0083**	**1.70**	**0.9961**	**35.15**
LooC(256×4)	MNIST	256×4	0.0062	**0.0058**	**1.31**	**0.9976**	**37.58**
向量量化变分自编码器 [41]	CIFAR10	1024×128	0.0527	0.2504	39.67	0.8595	23.32
分层向量量化变分自编码器 [44]	CIFAR10	1024×128	0.0533	0.2553	41.08	0.8553	23.22
SQ-VAE [40]	CIFAR10	1024×128	0.0482	0.2333	37.92	0.8779	24.07
CVQ-VAE [53]	CIFAR10	1024×128	0.0448	0.1883	24.73	0.8978	24.72
LooC-VAE	CIFAR10	32×4	0.0189	**0.0435**	**24.53**	**0.9805**	**32.22**
LooC-VAE	CIFAR10	256×4	0.0144	**0.0285**	**19.22**	**0.9880**	**34.51**
向量量化变分自编码器 [41]	FASHION-MNIST	1024×128	0.0377	-	12.73	-	23.93
CVQ-VAE [53]	FASHION-MNIST	1024×128	0.0344	-	8.85	-	24.66
LooC-VAE	FASHION-MNIST	32×4	0.0145	0.0158	8.7310	0.9864	32.4850
LooC-VAE	FASHION-MNIST	256×4	0.0103	0.0098	6.2368	0.9924	35.3393

表 8。 **重建结果** 在 MNIST [27], CIFAR10 [25], 和 FASHION-MNIST 的验证集上 [46]。我们的方法优于其他最先进方法，即使显著减小码本大小，仍能保持具有可比性的结果。

方法		$K \times d^* \downarrow$	使用率↑	LPIPS↓	rFID↓	结构相似性指数↑	峰值信噪比↑
VQGAN [7]	FFHQ	1024×256	42%	0.1175	4.42	0.6641	22.24
ViT-VQGAN [49]	FFHQ	8192×32	–	–	3.13	–	–
RQ-VAE [28]	FFHQ	2048×256	–	0.1302	3.88	0.6700	22.99
MoVQ [54]	FFHQ	1024×64	56%	0.0585	2.26	0.8212	26.72
SeQ-GAN [13]	FFHQ	1024×256	100%	–	3.12	–	–
CVQ-VAE [53]	FFHQ	1024×256	100%	0.0533	2.03	0.8398	26.87
LooC-VAE	FFHQ	256×4	100%	0.0501	**1.97**	**0.8499**	**27.73**
LooC-VAE	FFHQ	1024×4	100%	0.0346	**1.37**	**0.9276**	**32.44**
VQGAN [7]	ImageNet	1024×256	44%	0.2011	7.94	0.5183	19.07
ViT-VQGAN [49]	ImageNet	8192×32	96%	–	1.28	–	–
RQ-VAE [28]	ImageNet	16384×256	–	–	1.83	–	–
MoVQ [54]	ImageNet	1024×64	63%	0.1132	1.12	0.6731	22.42
SeQ-GAN [13]	ImageNet	1024×256	100%	–	1.99	–	–
CVQ-VAE [53]	ImageNet	1024×256	100%	0.1099	1.57	0.7115	23.37
LooC-VAE	ImageNet	256×4	100%	0.0916	**1.68**	**0.7233**	**23.64**
LooC-VAE	ImageNet	1024×4	100%	0.7160	**1.01**	**0.7160**	**29.15**

表9。**FFHQ 和 ImageNet 上的重建结果**。即使大幅缩小码本大小，我们的方法仍优于其他最先进方法，并取得了相当的结果。

以 256×256 的分辨率生成结果，并采用相同的训练参数。我们还使用与 LDM 相同的 $16\times$ 潜在表示下采样尺度，只是采用了不同的量化器。

4. 更多实验结果

4.1. 重建的定量结果

在图6中，我们将结果与 CIFAR 上各种最新的量化器进行比较10 [25]。我们改变码向量的数量（记为 K），并在图1右侧的基础上，加入四项额外评价指标的分数： l_1 损失、LPIPS、rFID 和峰值信噪比。这项实验表明，我们的方法能够有效地利用最少数量的码向量，充分发挥组合式向量量化的优势，从而提升向量量化的效果。相比之下，其他最先进的方法严重依赖大型码本。

表 8 全面比较了我们的方法与三个数据集上的最先进方法：MNIST [27], CIFAR10 [25], 和 FASHION-MNIST [46]。该表是表1的扩展。与需要更大码本（大小为 1024×128）的最先进方法相比，我们的方法使用更小的码本（大小为 32×4），却能取得相近的 rFID。值得注意的是，我们的方法在 l_1 损失、LPIPS、结构相似性指数和峰值信噪比方面均优于最先进方法，表现更出色。此外，当我们的方法使用大小为 1024×4的码本时，在各项指标上都取得了更加令人瞩目的结果。

方法	数据集	$K \times d^* \downarrow$	rFID↓	结构相似性指数↑	峰值信噪比↑
乘积量化 [19]	FFHQ	$16 \times 4 \times$ #64	2.43	0.8054	25.55
LooC-VAE		1024×4	1.37	0.9276	32.44
乘积量化 [19]	ImageNet	$16 \times 4 \times$ #64	1.96	0.6701	21.73
LooC-VAE		1024×4	1.01	0.7160	29.15

表10。 **PQ** [19] 和 LooC 在 FFHQ [22] 与 ImageNet 高分辨率数据集上的图像重建 [4]结果。

根据所提供的信息，表 9 是表 2的扩展，新增了 FFHQ 和 ImageNet 数据集上的 LPIPS 指标结果。结果表明，我们的方法在所有指标上均优于先前的最先进方法。此外，我们的方法使用的码本更小，具体而言，其大小仅为 CVQ-VAE 的 1/256。

在表 10 中，我们提供了 PQ [19] 在 FFHQ 和 ImageNet 数据集上的定量评估结果。处理高分辨率图像时，LooC 明显优于 PQ。从表 10以及表9中其他对比方法的结果可以看出，PQ 优于许多其他方法，而 LooC 的表现则明显优于 PQ。此外，我们认为统一码本是一种更简洁且合理的解决方案。

方法	数据集	$K \times d^* \downarrow$	FID↓
LooC-VAE	ImageNet类别平均	1024×4	46.78
LooC-VAE	LSUN教堂数据集	1024×4	15.17
LooC-VAE	LSUN卧室	1024×4	17.52

表11. **类别条件合成的 FID 分数比较** 在 ImageNet [4] 和 LSUN [48]上。ImageNet 中每个类别的 FID 分数均单独计算，然后对所有类别取平均。

4.2 生成定量结果

除了图 4中的可视化结果外，我们还在表 11中提供了 ImageNet 和 LSUN 上类别条件合成的 FID 指标比较。ImageNet 上的 FID 为 46.78，LSUN教堂数据集和 LSUN卧室上的 FID 分别为 15.17 和 17.52。请注意，ImageNet 中每个类别的 FID 分数均单独计算，然后对所有类别取平均。

5. 泛化能力

为进一步研究我们方法的泛化能力，我们在 FFHQ 上训练插入 LooC 的重建模型，并在 CelebA 数据集上对其进行测试。表12 和图9 分别展示了定量和定性结果。结果证明了我们方法的有效性和强大泛化能力。尽管仅在 FFHQ 数据集上进行训练，我们的方法在 CelebA 验证集上取得了 5.66 的 rFID$K = 256$ 。这相比 VQGAN [7]，（其 $K = 400$的 rFID 为 10.2）有显著提升。此外，当 LooC 使用 $K = 1024$时，rFID 进一步提升至 3.86。图中的可视化结果

方法	$K \times d^* \downarrow$	rFID↓	结构相似性指数↑	峰值信噪比↑
LooC-VAE	256×4	5.66	0.8141	26.36
LooC-VAE		3.86	0.8234	26.73

表 12. **泛化能力**。在 FFHQ [22] 上训练并在 CelebA 上测试的 LooC 的图像重建结果。

图 9 表明，我们的方法能够生成细致且高质量的重建结果。

6. 进一步讨论

研究展望 当前向量量化中的码本都是针对特定数据集的，LooC 也不例外。一种更高效的解决方案是构建可跨不同数据集和不同类型数据通用的码本。探索跨数据集通用码本是一个很有前景且有价值的未来研究方向。随着跨数据集数据多样性的增加，对码本容量的要求也随之提高。因此，如何设计更紧凑且容量更大的码本仍是一个值得研究的课题，而 LooC 为这一方向的未来探索奠定了基础。

其他影响 本文旨在研究一种更高效、更紧凑的视觉数据表示方法。我们的方法有助于降低存储和传输成本，促进日常生活中的数据交换与共享。同时，它还能加速科学研究并推动技术创新。此外，我们的方法将特征划分为子片段，而不是将其视为彼此独立且完整的特征。这种组合式特性不仅增强了数据安全和隐私保护，也降低了数据泄露风险，最终保护个人和组织的数据资产。



图 9. **LooC 的图像重建可视化**，在 FFHQ 上训练并在 CelebA 上测试